

**TO ERR IS HUMAN?  
A FUNCTIONAL COMPARISON OF  
HUMAN AND MACHINE DECISION-MAKING**

**DISSERTATION**

DER MATHEMATISCH-NATURWISSENSCHAFTLICHEN FAKULTÄT  
DER EBERHARD KARLS UNIVERSITÄT TÜBINGEN  
ZUR ERLANGUNG DES GRADES EINES  
DOKTORS DER NATURWISSENSCHAFTEN  
(DR. RER. NAT.)

VORGELEGT VON

ROBERT GEIRHOS  
AUS RAVENSBURG, DEUTSCHLAND

TÜBINGEN  
2021

GEDRUCKT MIT GENEHMIGUNG DER MATHEMATISCH-NATURWISSENSCHAFTLICHEN  
FAKULTÄT DER EBERHARD KARLS UNIVERSITÄT TÜBINGEN.

TAG DER MÜNDLICHEN QUALIFIKATION: 16.02.2022

DEKAN:

1. BERICHTERSTATTER:

2. BERICHTERSTATTERIN:

3. BERICHTERSTATTERIN:

PROF. DR. THILO STEHLE

PROF. FELIX A. WICHMANN, DPHIL

PROF. DR. ULRIKE VON LUXBURG

PROF. DR. GEMMA ROIG

OF ALL THINGS THE MEASURE IS MAN,  
OF THE THINGS THAT ARE, THAT THEY ARE,  
AND OF THE THINGS THAT ARE NOT, THAT THEY ARE NOT

PROTAGORAS



# Abstract

It is hard to imagine what a world without objects would look like. While being able to rapidly recognise objects seems deceptively simple to humans, it has long proven challenging for machines, constituting a major roadblock towards real-world applications. This has changed with recent advances in deep learning: Today, modern deep neural networks (DNNs) often achieve human-level object recognition performance. However, their complexity makes it notoriously hard to understand how they arrive at a decision, which carries the risk that machine learning applications outpace our understanding of machine decisions—without knowing when machines will fail, and why; when machines will be biased, and why; when machines will be successful, and why.

We here seek to develop a better understanding of machine decision-making by comparing it to human decision-making. Most previous investigations have compared intermediate representations (such as network activations to neural firing patterns), but ultimately, a machine’s behaviour (or output decision) has the most direct relevance: humans are affected by machine decisions, not by “machine thoughts”. Therefore, the focus of this thesis and its six constituent projects (1–6) is a *functional* comparison of human and machine decision-making. This is achieved by transferring methods from human psychophysics—a field with a proven track record of illuminating complex visual systems—to modern machine learning.

The starting point of our investigations is a simple question: How do DNNs recognise objects, by texture or by shape? Following behavioural experiments with cue-conflict stimuli, we show that the textbook explanation of machine object recognition—an increasingly complex hierarchy based on object parts and shapes—is inaccurate. Instead, standard DNNs simply exploit local image textures (1). Intriguingly, this difference between humans and DNNs can be overcome through data augmentation: Training DNNs on a suitable dataset induces a human-like shape bias and leads to emerging human-level distortion robustness in DNNs, enabling them to cope with unseen types of image corruptions much better than any previously tested model. Motivated by the finding that texture bias is pervasive throughout object classification and object detection (2), we then develop “error consistency”. Error consistency is an analysis to understand how machine decisions differ from one another depending on, for instance, model architecture or training objective. This analysis reveals remarkable similarities between feedforward vs. recurrent (3) and supervised vs. self-supervised models (4). At the same time, DNNs show little consistency with human observers, reinforcing our finding of fundamentally different decision-making between humans and machines. In the light of these results, we then take a step back, asking where these differences may originate from. We find that many DNN shortcomings can be seen as symptoms of the same underlying pattern: “shortcut learning”, a tendency to exploit unintended patterns that fail to generalise to unexpected input (5). While shortcut learning accounts for many functional differences between human and machine perception, some of them can be overcome: In our last investigation, a large-scale behavioural comparison, toolbox and benchmark (6), we report partial success in closing the gap between human and machine vision.

Taken together our findings indicate that our understanding of machine decision-making is riddled with (often untested) assumptions. Putting these on a solid empirical footing, as done here through rigorous quantitative experiments and functional comparisons with human decision-making, is key: for when humans better understand machines, we will be able to build machines that better understand humans—and the world we all share.

# Zusammenfassung

Es lässt sich nur schwer vorstellen, wie eine Welt ohne Objekte aussehen würde. Während die schnelle Erkennung von Objekten für den Menschen täuschend einfach erscheint, war sie für Maschinen lange Zeit eine Herausforderung, was eine große Hürde für praktische Anwendungen dargestellt hat. Dies hat sich mit den jüngsten Fortschritten im Bereich Deep Learning geändert: Moderne Tiefe Neuronale Netze (TNNs) erreichen heute oft eine Objekterkennungsleistung auf menschlichem Niveau. Ihre Komplexität macht es jedoch notorisch schwer zu verstehen, wie sie zu einer Entscheidung kommen. Das birgt das Risiko, dass Anwendungen des maschinellen Lernens unser Verständnis von maschinellen Entscheidungen übersteigen—ohne zu wissen, wann Maschinen Fehler machen, und warum; wann Maschinen voreingenommen sind, und warum; wann man sich auf Maschinen verlassen kann, und warum.

Unser Ziel ist es, ein besseres Verständnis von maschinellen Entscheidungen zu entwickeln, indem wir sie mit menschlichen Entscheidungen vergleichen. Bisherige Vergleiche haben meist Zwischenrepräsentationen untersucht (wie beispielsweise Aktivierungsmuster eines neuronalen Netzwerks mit neuronalen Feuermustern), aber letztendlich hat das Verhalten (oder die Entscheidung) einer Maschine die größte Relevanz: Menschen sind ganz konkret und direkt von maschinellen Entscheidungen betroffen, nicht von “maschinellen Gedanken”. Daher liegt der Schwerpunkt dieser Arbeit und ihrer sechs Teilprojekte (1–6) auf einem *verhaltensbasierten* Vergleich der menschlichen und maschinellen Entscheidungsfindung. Dies erreichen wir durch die Anpassung und den Transfer von Methoden aus der menschlichen Psychophysik, einem Gebiet, das große Erfahrung darin hat, komplexe visuelle Systeme zu verstehen.

Der Ausgangspunkt unserer Untersuchungen ist eine einfache Frage: Wie erkennen TNNs Objekte, anhand ihrer Form (wie allgemein angenommen) oder etwa ihrer Textur? Basierend auf Verhaltensexperimenten mit Konfliktbildern zeigen wir, dass die Lehrbuch-Erklärung der maschinellen Objekterkennung—eine zunehmend komplexe Hierarchie, die auf Objektteilen und Formen basiert—unzutreffend ist. Stattdessen nutzen Standard-TNNs einfach lokale Texturinformationen aus (1). Interessanterweise kann dieser Unterschied zwischen Menschen und TNNs durch Anpassen der Trainingsdaten überwunden werden: Das Training von TNNs auf einem geeigneten Datensatz verursacht einen menschenähnlichen Fokus auf die Objektform und führt gleichzeitig dazu, dass TNNs deutlich robuster darin werden, Objekte trotz lokaler Bildstörungen zu erkennen—deutlich besser als jedes zuvor getestete Modell, und nahe am menschlichen Niveau. Motiviert durch die Erkenntnis, dass der Texturfokus in der Objektklassifikation und Objekterkennung allgegenwärtig ist (2), entwickeln wir anschließend “error consistency”. Error consistency (oder Fehlerkonsistenz) ist eine Analyse, um zu verstehen, wie sich maschinelle Entscheidungen voneinander unterscheiden, zum Beispiel in Abhängigkeit von der Modellarchitektur oder dem Trainingsziel. Diese Analyse zeigt bemerkenswerte Ähnlichkeiten zwischen rein vorwärtsgerichteten versus rekurrenten TNNs (3) und überwachten versus selbstüberwachten Modellen (4). Gleichzeitig zeigen TNNs kaum Fehlerkonsistenz mit menschlichen Versuchspersonen, was unseren Befund einer grundlegend unterschiedlichen Entscheidungsfindung zwischen Mensch und Maschine untermauert. Im Lichte dieser Ergebnisse gehen wir dann einen Schritt zurück und fragen, woher diese Unterschiede stammen könnten. Wir beobachten, dass viele verschiedene Defizite von TNNs als Symptome ein und desselben zugrunde liegenden Ursache gesehen werden können: “Shortcut learning” (oder Abkürzungslernen), eine Tendenz, unbeabsichtigte Muster in den Daten auszunutzen, die bei unerwarteten Eingaben schnell zu Fehlern führen (5). Das Abkürzungslernen

ist für viele Verhaltensunterschiede zwischen menschlicher und maschineller Wahrnehmung verantwortlich, allerdings können einige dieser Unterschiede überwunden werden: In unserer letzten Untersuchung, einem umfassenden Verhaltensvergleich inklusive Software-Toolbox und Benchmark (6), berichten wir über einen Teilerfolg: die Lücke zwischen menschlicher und maschineller Wahrnehmung beginnt, kleiner zu werden.

Zusammengenommen zeigen unsere Ergebnisse, dass unser Verständnis maschineller Entscheidungsfindung von (oft ungeprüften) Annahmen durchdrungen ist. Diese auf eine solide empirische Basis zu stellen, wie es hier durch rigorose quantitative Experimente und verhaltensbasierte Vergleiche mit der menschlichen Entscheidungsfindung geschieht, ist der Schlüssel zum Erfolg: Denn wenn Menschen Maschinen besser verstehen, werden wir in der Lage sein, Maschinen zu bauen, die Menschen besser verstehen—und die Welt, die wir alle teilen.

# List of publications

## Peer-reviewed conference and journal publications

[Oral] Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., and Brendel, W. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2019a

<https://github.com/rgeirhos/texture-vs-shape/>  
<https://github.com/rgeirhos/Stylized-ImageNet/>

*The above article was accepted as “Oral” at the International Conference on Learning Representations (ICLR 2019); additionally an abstract based on this article was accepted as “Oral” at VSS 2019, the 19th Annual Meeting of the Vision Sciences Society (Geirhos et al., 2019b).*

Geirhos, R., Meding, K., and Wichmann, F. A. Beyond accuracy: quantifying trial-by-trial behaviour of CNNs and humans by measuring error consistency. In *Advances in Neural Information Processing Systems*, 2020b

<https://github.com/wichmann-lab/error-consistency>

Geirhos, R., Jacobsen, J.-H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., and Wichmann, F. A. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2:665–673, 2020a

<https://github.com/rgeirhos/shortcut-perspective>

[Oral] Geirhos, R., Narayanappa, K., Mitzkus, B., Thieringer, T., Bethge, M., Wichmann, F. A., and Brendel, W. Partial success in closing the gap between human and machine vision. In *Advances in Neural Information Processing Systems*, 2021



<https://github.com/bethgelab/model-vs-human>

*The above article was accepted as “Oral” at the Conference on Neural Information Processing Systems (NeurIPS 2021), additionally an abstract based on this article was accepted as “Oral” at VSS 2022, the 22nd Annual Meeting of the Vision Sciences Society (Geirhos et al., 2022).*



### Peer-reviewed workshop publications

Michaelis, C., Mitzkus, B., Geirhos, R., Rusak, E., Bringmann, O., Ecker, A. S., Bethge, M., and Brendel, W. Benchmarking Robustness in Object Detection: Autonomous Driving when Winter is Coming. In *NeurIPS Workshop on Machine Learning for Autonomous Driving*, 2019


-  <https://github.com/bethgelab/robust-detection-benchmark>
-  <https://github.com/bethgelab/imagecorruptions>
-  <https://github.com/bethgelab/styleize-datasets>

**[Oral]** Geirhos, R., Narayanappa, K., Mitzkus, B., Bethge, M., Wichmann, F. A., and Brendel, W. On the surprising similarities between supervised and self-supervised models. In *NeurIPS Workshop on Shared Visual Representations in Human & Machine Intelligence*, 2020c


### Publications not part of this thesis

Since my acceptance to the PhD program of the International Max Planck Research School of Intelligent Systems, I have co-authored the following publications which are not a part of this thesis:


Geirhos, R., Medina Temme, C. R., Rauber, J., Schütt, H. H., Bethge, M., and Wichmann, F. A. Generalisation in humans and deep neural networks. In *Advances in Neural Information Processing Systems*, 2018

-  <https://github.com/rgeirhos/generalisation-humans-DNNs/>

Borowski, J., Zimmermann, R., Schepers, J., Geirhos, R., Wallis, T. S. A., Bethge, M., and Brendel, W. Exemplary natural images explain CNN activations better than feature visualizations. In *International Conference on Learning Representations*, 2021

-  [https://github.com/bethgelab/testing\\_visualizations/](https://github.com/bethgelab/testing_visualizations/)

**[Spotlight]** Zimmermann, R. S., Borowski, J., Geirhos, R., Bethge, M., Wallis, T. S. A., and Brendel, W. How well do feature visualizations support causal understanding of CNN activations? In *Advances in Neural Information Processing Systems*, 2021

-  <https://github.com/brendel-group/causal-understanding-via-visualizations>

Huber, L. S., Geirhos, R., and Wichmann, F. A. Out-of-distribution robustness: Limited image exposure of a four-year-old is enough to outperform ResNet-50. In *NeurIPS Workshop on Shared Visual Representations in Human & Machine Intelligence*, 2021

Meding, K., Buschoff, L. M. S., Geirhos, R., and Wichmann, F. A. Trivial or impossible–dichotomous data difficulty masks model differences (on ImageNet and beyond). In *International Conference on Learning Representations*, 2022

Additional authorship information (joint first/senior authors, corresponding author) can be found on the first page of every publication.

## Statement of author contributions

This statement lists author contributions for collaborative projects according to §6 Abs. 2 Satz 3, Promotionsordnung der Mathematisch-Naturwissenschaftlichen Fakultät der Universität Tübingen. Frequently, the published versions of my articles already include author contribution statements; in those cases, the statement is reprinted here.

Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., and Brendel, W. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations, 2019a*

*The project was conceived and led by R.G., who also designed and curated the stimuli. The extensive human data was collected by P.R. under the guidance of R.G. and F.A.W. based on an experiment coded by R.G. Funding for this set of experiments, as well as the psychophysical laboratory resources, were provided by F.A.W. The idea for “Stylized-ImageNet” was conceived by Alexander Ecker, and the computational resources were provided by M.B. The computational experiments were conducted by R.G. under the guidance of W.B. and M.B., except for the object detection experiments which were conducted by C.M. The data was analysed and visualised by R.G. with input from M.B., F.A.W. and W.B. Pre-trained models, data, code and materials were made openly accessible by R.G. The project was supervised by F.A.W. and M.B. in the first stage (psychophysical experiments) and by W.B. with input from M.B. and F.A.W. in the second stage (computational experiments). The paper draft was written by R.G. based on discussions with C.M. (who pointed out important connections to other projects), M.B., F.A.W. and W.B. except for the object detection experiments which were described by C.M. and the psychophysical methods section which was jointly written by P.R. and R.G. The draft was substantially edited by W.B.*

Michaelis, C., Mitzkus, B., Geirhos, R., Rusak, E., Bringmann, O., Ecker, A. S., Bethge, M., and Brendel, W. Benchmarking Robustness in Object Detection: Autonomous Driving when Winter is Coming. In *NeurIPS Workshop on Machine Learning for Autonomous Driving, 2019*

*The initial project idea for improving detection robustness was developed by E.R., R.G. and C.M. The initial idea of benchmarking detection robustness was developed by C.M., B.M., R.G., E.R. & W.B. The overall research focus on robustness was collaboratively developed in the Bethge, Bringmann and Wichmann labs. The Robust Detection Benchmark was jointly designed by C.M., B.M., R.G. & E.R.; including selecting datasets, corruptions, metrics and models. B.M. and E.R. jointly developed the pip-installable package to corrupt arbitrary images. B.M. developed code to stylize arbitrary datasets with input from R.G. and C.M.; C.M. and B.M. developed code to evaluate the robustness of arbitrary object detection models. B.M. prototyped the core experiments; C.M. ran the reported experiments. The results were jointly analysed and visualized by C.M., R.G. and B.M. with input from E.R., M.B. and W.B.; C.M., B.M., R.G. & E.R. worked towards making our work reproducible, i.e. making data, code and benchmark openly accessible and (hopefully) user-friendly. Senior support, funding acquisition and infrastructure were provided by O.B., A.S.E.,*

M.B. and W.B. The illustratory figures were designed by E.R., C.M. and R.G. with input from B.M. and W.B. The paper was jointly written by R.G., C.M., E.R. and B.M. with input from all other authors.

Geirhos, R., Meding, K., and Wichmann, F. A. Beyond accuracy: quantifying trial-by-trial behaviour of CNNs and humans by measuring error consistency. In *Advances in Neural Information Processing Systems*, 2020b

*Based on ideas from Schönfeldner & Wichmann (2013) and Meding et al. (2019), R.G. first applied trial-by-trial analysis ideas to CNNs. Thereafter, all three authors jointly initiated the project. R.G. and K.M. jointly led the project. K.M. derived the bounds, performed and visualised the simulations and acquired the Brain-Score data (with input from R.G.). The CNN experiments were performed, analysed and visualised by R.G. (with input from K.M.). F.A.W. provided guidance, feedback, and pointed out the link to molecular psychophysics. All three authors planned and structured the manuscript. R.G. and K.M. wrote the paper with active input from F.A.W.*

Geirhos, R., Narayanappa, K., Mitzkus, B., Bethge, M., Wichmann, F. A., and Brendel, W. On the surprising similarities between supervised and self-supervised models. In *NeurIPS Workshop on Shared Visual Representations in Human & Machine Intelligence*, 2020c

*Project idea: R.G. and W.B.; project lead: R.G.; implementing and training self-supervised models: K.N.; model evaluation pipeline: R.G., K.N. with input from W.B.; data visualisation: R.G. and B.M. with input from M.B., F.A.W. and W.B.; guidance, feedback, infrastructure & funding acquisition: M.B., F.A.W. and W.B.; paper writing: R.G. with input from all other authors.*

Geirhos, R., Jacobsen, J.-H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., and Wichmann, F. A. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2:665–673, 2020a

*The project was initiated by R.G. and C.M. and led by R.G. with support from C.M. and J.J.; F.A.W. added the cognitive science and neuroscience connection; M.B. and W.B. reshaped the initial thrust of the perspective and together with R.Z. supervised the machine learning components. The toy experiment was conducted by J.J. with input from R.G. and C.M. Most figures were designed by R.G. and W.B. with input from all other authors. Figure 2 (left) was conceived by M.B. The first draft was written by R.G., J.J. and C.M. with input from F.A.W. All authors contributed to the final version and provided critical revisions from different perspectives.*

Geirhos, R., Narayanappa, K., Mitzkus, B., Thieringer, T., Bethge, M., Wichmann, F. A., and Brendel, W. Partial success in closing the gap between human and machine vision. In *Advances in Neural Information Processing Systems*, 2021

*Project idea: R.G. and W.B.; project lead: R.G.; coding toolbox and model evaluation pipeline: R.G., K.N. and B.M. based on a prototype by R.G.; training models: K.N. with input from R.G., W.B. and M.B.; data visualisation: R.G., B.M. and K.N. with input from M.B., F.A.W. and W.B.; psychophysical data collection: T.T. (12 datasets) and B.M. (2 datasets) under the guidance of R.G. and F.A.W.; curating stimuli: R.G.; interpreting analyses and findings: R.G., M.B., F.A.W. and W.B.; guidance, feedback, infrastructure & funding acquisition: M.B., F.A.W. and W.B.; paper writing: R.G. with help from F.A.W. and W.B. and input from all other authors.*



# *Acknowledgments*

## **Advisors.**

I am incredibly fortunate to have had the chance to learn from no fewer than three mentors: Felix Wichmann, Matthias Bethge and Wieland Brendel. Together, they formed an incredibly inspiring and supportive group. I could neither have hoped nor wished for a better collaboration.

To Felix, I owe much more than just the start of my scientific career. He took a risk on me when sending me to deep learning workshops and vision science conferences when I was working on my very first project in the lab, at a time when I had not even completed my studies. If one day I am even half the mentor he is, my future students or collaborators can consider themselves lucky. Felix' unique combination of a critical mindset ("many more things are known to be true than are", as Felix used to quote his own advisor), his curiosity and contagious passion for science, and a deep interest in the personal development of his students made him a truly exceptional advisor.

Many of my projects aim to "connect the dots", to discover unifying patterns behind previously disconnected areas and findings. This would not have been possible without the tireless work of Matthias, who created an inspiring environment where connections between different projects and areas can be harnessed naturally since so much knowledge is concentrated in a single group. Matthias also played a crucial role in making Tübingen a fantastic place to be for any machine learning PhD student. A few years ago during my studies, I went to Amsterdam for an exchange semester since Tübingen, at the time, didn't offer any courses on deep learning. Today, Tübingen has a vibrant ML community, an achievement in which Matthias has played a significant role. His deep knowledge of the field and his drive for conceptual clarity have shaped many of my projects.

Wieland, although not officially my PhD advisor, unofficially very much took on the role of one. Whether I had questions about science or software engineering, tools or talks, rebuttals or reviews, collaborations or career choices: the regular "slow dating" meetings with Wieland have always been a highlight of my week, and I won't even try to describe how much I've learned from him. His sharp mind and enthusiasm for my projects made it a true pleasure to work together.

## **TAC members and thesis examiners.**

I would like to thank Andreas Geiger for his time, input and support during my thesis advisory committee (TAC) meetings: His fresh and knowledgeable perspective has been very helpful. Furthermore, I am grateful to Ulrike von Luxburg and Gemma Roig for taking on the roles of second and third examiners for this thesis, and to the members of my thesis defense committee (Felix, Matthias, Ule and Tom).

## **Co-Authors.**

During my PhD I have been fortunate enough to work and publish with, in alphabetical order: Matthias Bethge, Judy Borowski, Wieland Brendel, Oliver Bringmann, Jörn-Henrik Jacobsen, Alexander Ecker, Lukas

Huber, Kristof Meding, Carlos Medina Temme, Claudio Michaelis, Benjamin Mitzkus, Kantharaju Narayanappa, Jonas Rauber, Patricia Rubisch, Evgenia Rusak, Judith Schepers, Luca Scholze Buschoff, Heiko Schütt, Tizian Thieringer, Tom Wallis, Felix Wichmann, Shuchen Wu, Richard Zemel, and Roland Zimmermann. Whether we collaborated on a conference abstract, a workshop paper or a full-blown conference or journal article: it has been a tremendous pleasure to work alongside some of the most gifted scientists, writers, experimentalists, and software engineers that I have come to know!

### **Colleagues.**

Science is at least as much a social endeavour as it is a solitary one, and behind most scientific publications there is a story: sometimes this story starts with a chance encounter in the coffee lounge, sometimes with a journal club discussion, with a curious question during a meeting, or a chat following an inspiring talk. My colleagues from both the Wichmann and Bethge labs have constantly supported me throughout my PhD journey, and contributed much to making it a time that I will always remember fondly. Describing how much I owe to every single one of them would fill more than a few thesis chapters.

### **Administrative and technical support team.**

I am very grateful to all those who enabled me to focus on scientific topics through their highly competent help with administrative, formal, technical and organisational matters: Silke and Uli, without whom our psychophysical laboratory, lab website and desks would have been empty; Heike, Melanie, Judith, Moni and Georg, who were always there to answer any question; Leila and Sara, who have made it a pleasure to be part of IMPRS-IS; our CIN IT and ML-Cloud admin team, without whom my computational experiments would have been mostly restricted to MNIST.

### **Students.**

Patricia, Benny, Shuchen, Ole, Tizian, Lukas: It has been a pleasure to work with such a talented group of students! Without them, my PhD would have been a lot less rewarding and fun. While I probably mentioned that the most important aspect of their project (next to it being interesting, sound and relevant) is that they can learn as much as possible, I cannot help but admit that I learned just as much from and through them. I am grateful to all of my former and current students for placing their trust in me.

### **Institutions.**

I am grateful to the University of Tübingen and the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for the opportunity to spend time in an inspiring and highly collaborative environment. I've been very fortunate to have experienced this spirit since the very beginning of my scientific endeavours. When I first started to work with deep learning in January 2016 as a lab rotation student with Felix Wichmann, we were enthusiastic about our project idea, but there was one problem: Back then, I had never worked with deep learning before, and neither had the lab. Within a week I was able to meet seasoned researchers from the MPI-IS and University labs, who all dedicated their time to answering the questions of a 5th semester Bachelor student; I was even invited to join the "Deep Journal Club", a regular meeting of deep learning researchers from various labs in Tübingen. Looking back, a substantial part of the success of this project is owed to this early support and collaborative spirit which, in my experience, is so distinctive of the research landscape in Tübingen. It is only consequent that this collaborative atmosphere has now become institutionalized at the IMPRS-IS, where a vivid exchange between scholars is the norm rather than the exception, encouraged through various workshops, the yearly Boot Camp, and inter-institutional mentors.

**Tax payers.**

I am yet to meet a person who enjoys paying taxes on their hard-earned money. I am all the more indebted to the taxpayers who funded my research, through the University and through an IMPRS-IS scholarship. Being able to attend international conferences, having access to a large compute cluster, or simply not having to worry about funds for remunerating experimental participants—these are just some of the privileges that I am very grateful for. My friends from countries that cannot afford this luxury know that this is not something we can simply take for granted.

**Open Source contributors.**

Many of my projects would have been impossible without the work of those who made their models, tools and data openly accessible. Moreover, I am grateful to Kevin Godby, Bil Kleb and Bill Wood for providing a Tufte-inspired L<sup>A</sup>T<sub>E</sub>X-template which provided a beautiful starting point for this thesis (licensed under the Apache License, Version 2.0, <http://www.apache.org/licenses/LICENSE-2.0>).

**Reviewers.**

There is not a single manuscript in this thesis that has not improved through the feedback of thoughtful reviewers, which helped a lot in improving clarity, writing, interpretation, figures and experiments. Furthermore, I would like to thank Lukas Schott, Caroline Seidel and Felix Wichmann for their insightful comments on my thesis draft.

**Family, Friends, Partner.**

I am immensely grateful to my family, who have supported me in pursuing my dreams for as long as I can remember, no matter what: Bettina, Walter and Katja; and to my dear grandmother Cathrin, who never had the chance to study and yet knows much more than I do. My time as a PhD student would not have been half as joyful without my truly wonderful friends, with whom I shared many a laugh, run and adventure. Finally, I am incredibly grateful to share my life with my loving partner Caroline.





# Contents

<b>1</b>	<b>Introduction</b>	<b>17</b>
1.1	Motivation: the future of machine learning . . . . .	17
1.2	Why compare machines against humans? . . . . .	19
1.3	Why study behaviour? . . . . .	23
1.4	Why study visual object recognition? . . . . .	25
1.5	Outline . . . . .	27
<b>2</b>	<b>Publications</b>	<b>29</b>
2.1	ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness . . . . .	29
2.2	Benchmarking Robustness in Object Detection: Autonomous Driving when Winter is Coming	53
2.3	Beyond accuracy: quantifying trial-by-trial behaviour of CNNs and humans by measuring error consistency . . . . .	75
2.4	On the surprising similarities between supervised and self-supervised models . . . . .	102
2.5	Shortcut Learning in Deep Neural Networks . . . . .	113
2.6	Partial success in closing the gap between human and machine vision . . . . .	143
<b>3</b>	<b>Discussion</b>	<b>175</b>
3.1	The inductive bias perspective . . . . .	175
3.2	The model of human object recognition perspective . . . . .	178
3.3	Limitations . . . . .	179
<b>4</b>	<b>Outlook: “Big Questions” for the future</b>	<b>181</b>
4.1	How can the contradiction between behavioural and neural results be resolved? . . . . .	182
4.2	What does a network’s representation tell us about the network’s behaviour? . . . . .	191
4.3	Concluding remarks . . . . .	198
	<b>Bibliography</b>	<b>201</b>



# 1 Introduction

## 1.1 Motivation: the future of machine learning

TIME FLIES FAST—BUT NOT LIKE AN ARROW.

Just like an arrow never reverses direction, time never moves backwards: there is only one history. And yet, in liberating contrast to the predetermined trajectory of an arrow in the air, there are inconceivably many possible futures ahead of us. We cannot change the past, but we can shape the future. However, at the same time, with choice comes responsibility. From the joy of a moment, to the regret over a day lost unproductive, to the fulfilment of a life with a legacy, to concerns over the environmental trajectory of humankind itself—when the future has become the past, we cannot change it anymore.

This responsibility can increasingly be felt within the field of machine learning, which has transitioned from a flourishing niche to a key driver of technologies that affect billions of people around the globe. This impact goes well beyond seemingly innocuous aspects such as which products are being recommended to a user but also affects, potentially, whether one is invited for a job interview, which diagnosis one receives in the hospital, or whether one receives a bank loan. In this context, machine learning, the science and art of teaching algorithms how to learn from data, is currently at a crucial crossroads where two very different paths for the future are possible: a bright one and a rather dim one. On the bright path, machines promise to solve some of humankind’s most challenging riddles, shed light onto our own brains and minds, drastically reduce the number of traffic accidents, and overcome language barriers. On the dim path, however, machines might just as well be used to make automated decisions that humans can neither follow nor contend, increasing social disparities by exploiting predictive patterns of biased datasets. In the following, I will argue that our ability to choose between those two very different futures crucially hinges on one challenge: whether our *understanding of machine decision-making* will be able to keep pace with applications of machine learning.

OF COURSE, obtaining a better understanding of machine decision-making is not going to solve all problems of machine learning. In particular, the impact of a tool is always going to depend on its user, and there are many questionable or malicious use cases for which machine learning can be exploited. This makes it all the more important that we understand the limitations, biases and robustness of those machine learning applications that are developed with good intentions. In a particularly sad example of what the consequences are when this is not the case, Joshua D. Brown, 40, died on 7 May 2016 when his Tesla autopilot failed to identify a large white truck against the bright sky (Vlasic & Boudette, 2016). This shows the pivotal role of understanding what can, and cannot, be expected from a particular application of machine learning.

MACHINE LEARNING is currently the most successful paradigm within the broader context of artificial intelligence (AI). “Artificial intelligence” has become somewhat of a loaded term since it all too easily evokes the expectation of human-level intelligence through mysterious yet powerful machines. While many machine learning startups appear happy to tap into this expectation when registering their “.ai” domain, sky-high expectations are easy to raise but hard to meet (e.g. Mitchell, 2021): As a field, artificial intelligence has already experienced two extended phases of enthusiasm each followed by a so-called “AI winter” characterised by disappointment and a drastic reduction in funding, both public and private. The first winter came around 1973, when governments, venture capitalists and even the general public realized that in contrast to large expectations and even bigger promises, AI methods of the time worked only on toy problems and were thus unlikely to lead to widespread applications.<sup>1</sup> About two decades later, following an extended period of renewed interest and funding, the second AI winter was around the corner when it was realized that then-prominent methods such as expert systems were useful indeed but only for very limited and narrow use cases.

Fast-forwarding a few decades, we have now reached a stage where machine learning is successfully being used in many fields and application areas—is it, finally, time for an eternal summer? Critical voices advise caution:

*“In spite of all the commercial hustle and bustle around AI these days, there’s a mood that I’m sure many of you are familiar with of deep unease among AI researchers who have been around more than the last four years or so. This unease is due to the worry that perhaps expectations about AI are too high, and that this will eventually result in disaster ...”*

<sup>1</sup> The beginning of this first AI winter is often credited to the “Lighthill Report” named after Sir James Lighthill, who conducted a widely influential and deeply critical survey of the current state of AI on behalf of the British Science Research Council (Lighthill, 1973). The repercussions of this report were felt well beyond the UK, as global funding of AI research entered a steep decline.

The fact that this sentiment certainly does not ring entirely false in today’s ears, even though it was voiced nearly *four decades ago* (McDermott et al., 1985, p. 122), gives reason for concern. If we had to make an educated guess about the most likely cause of a new, third AI winter: where would we see the biggest potential for disappointment and disaster?

Probably not because today’s methods were to face strict theoretical limits that strongly constrain their usefulness, similar to how one-layer neural networks with monotonic nonlinearities were shown to be provably unable to represent the simple XOR-function, a major drawback of early perceptrons (Minsky & Papert, 1969). Likewise, probably neither because today’s methods were only applicable to highly limited and narrow use cases—currently, most methods require just a little fine-tuning when applied to a different problem. In fact, it is precisely their widespread use and deceptively good performance on demanding tasks that might carry the greatest risk: the risk that machine learning applications outpace our understanding of machine decisions, that we lull ourselves into a sense of security without knowing when machines will fail, and why; when machines will be biased, and why; when machines will be successful, and why. Answering these questions will be of decisive importance if we are to avoid a third wave of disappointment, and instead set out to use machine learning for a brighter future.

IT IS THE AIM of this thesis to improve our understanding of machine decision-making.<sup>2</sup> For reasons outlined in the following sections, this will be achieved by comparing machine behaviour against human behaviour on visual object recognition.

<sup>2</sup> The term “machine decision” will be used to refer to any kind of decision by algorithms (not just those that control physical devices).

## 1.2 *Why compare machines against humans?*

AT FIRST, DEVELOPING a better understanding of machine decision-making sounds like an entirely *technical* endeavour—and indeed, many existing approaches attempt to understand machines “in isolation”, without taking their relationships to humans into account. However, there are a number of (explicit and implicit) ways in which humans exert a decisive influence over actual, perceived and presumed machine decision-making. Jointly, the following factors explain why it is often helpful, and sometimes even indispensable, to compare machines against humans.

### (1.) *Humans as annotators define ground truth*

What exactly constitutes a machine “error” or “success” is in most

cases defined by humans, irrespective of whether we would like to build a machine that mimics typical human output (e.g. creating a bot capable of small talk) or whether we would like to obtain a machine that improves over human decision-making (e.g. in the medical context, or in safety-critical applications like autonomous driving). Furthermore, how errors are weighted relative to each other (e.g. whether a decision is regarded as an innocuous mistake or a fatal error) often depends on human values.

(2.) *Humans as engineers create, train and evaluate models*

Perhaps the most obvious way in which humans influence model decision-making is in their role as researchers, engineers or data scientists who create, train and evaluate models. This influence extends beyond neutral technical choices, and cognitive psychology knows a number of well-established ways in which humans are likely to bias the results of an experiment. For instance, *experimenter bias* is a (usually unconscious) tendency of an experimenter to “obtain from his subjects the data he expects or wants to obtain” (Rosenthal & Fode, 1961, p. 183). As a result, triple-blind experiments are often considered the gold standard (participants do not know whether they e.g. receive a placebo or treatment, neither does the experimenter, nor the data analyst). In machine learning, experiments are usually single-blind: the machine learning model does not have meta-knowledge, but experimenter and data analyst are one and the same “non-blind” person. When training multiple models, *cherry picking* (e.g. a tendency to report only the best model / feature visualisation, effectively treating random seeds as hyperparameters) and *confirmation bias* (“a general tendency for people to believe too much in their favored hypothesis”, cf. Klayman, 1995, p. 385) may present themselves as subconscious allies of competitive acceptance rates and a publish-or-perish culture (Smaldino & McElreath, 2016; Frith, 2020). Finally, the *law of the instrument* (Kaplan, 1964; Maslow, 1966) may influence which machine learning models are trained/evaluated in the first place: “If all you have is a hammer, everything looks like a nail”, the proverb knows. (Any parallels to the ubiquitous use of deep learning, whether warranted or unwarranted, must surely be coincidental.)

(3.) *Humans as benchmarks and baselines*

“Of all things the measure is Man”, Protagoras famously said (quoted from Silvermintz, 2015, p. viii, preface). To a certain degree, this statement is relevant in machine learning as well, where many applications need to be measured relative to human performance. While we may be able to forgive a machine for making a driving error that most humans would have made, we would never forgive ourselves for al-

lowing our children to ride in an autonomous car that ends up making a mistake we would not have made ourselves. Reaching or surpassing human performance is a goal in many areas where machine learning is applied; however, in the light of this interest, it is astonishing how rarely human performance is measured thoroughly. For instance, ResNets—a deep learning model family—were famously reported to surpass human-level accuracies on the challenging ImageNet dataset by He et al. (2015), but the performance of “human object recognition” is based on the self-reported accuracy of a single researcher, Andrej Karpathy (Russakovsky et al., 2015), trained on just 500 images.<sup>3</sup> It remains speculative to ask why human performance is not evaluated more frequently; it might well be the case that the careful and time-consuming work to obtain high-quality human data appears daunting. Nonetheless, surpassing human-level performance is a shared goal across many application areas, ranging from games like chess (Campbell et al., 2002), Atari (Mnih et al., 2015) and Go (Silver et al., 2016) to pneumonia detection in the medical context (Rajpurkar et al., 2017), face verification (Taigman et al., 2014), grammatical error correction (Grundkiewicz & Junczys-Dowmunt, 2018), or language understanding (McClelland et al., 2020), just to name a few.

<sup>3</sup> You may wonder whether training on 500 images or 0.04% of the full ImageNet training dataset already makes you an “expert annotator” as described by Russakovsky et al. (2015)—a concern that isn’t helped by the fact that the dataset has twice as many classes (1,000); i.e. at least 50% of classes were not represented in the human training set at all.

#### (4.) *Humans as role models*

Human abilities not only serve as challenging benchmark baselines for tasks that machines can already tackle; they also serve as role models for tasks or capabilities that are still beyond reach. Here, humans are often taken as a proof-of-concept. For instance, in their 2015 review of deep learning, LeCun et al. (2015, p. 442) describe how humans and animals “discover the structure of the world by observing it, not by being told the name of every object”. This references the difference between supervised learning, the dominant deep learning paradigm of the time, and unsupervised or self-supervised learning, the challenge of learning useful representations without receiving external labels (such as object names). At the time of writing their review, it was clear that it *could* be achieved (taking biological learning as a role model), but it was not yet clear *how*. In cases like these, machine learning often takes inspiration from human perception (e.g. Oord et al., 2018; Lotter et al., 2020; Orhan et al., 2020), in the hope that this will point the way towards improved machine perception.

#### (5.) *Humans as metaphors*

Historically, machines have often been used as metaphors for our own brains and minds (Smith, 1993). These metaphors continually evolved; usually, the most advanced machines and technologies of a time replaced or refined older metaphors. This tradition dates back to at least

Réne Descartes, who coined the *mechanistic metaphor*: “I suppose the body to be nothing but a statue or machine made of earth. [...] We see clocks, artificial fountains, mills, and other such machines which, although only man-made, have the power to move of their own accord in many different ways. But I am supposing this machine to be made by the hands of God, and so I think you may reasonably think it capable of a greater variety of movements than I could possibly imagine in it, and of exhibiting more artistry than I could possibly ascribe to it.” (Descartes, 1662, p. 15).

With the advent of modern times, clocks and mills had been replaced by computers as the most fascinating and complex machinery; consequently, brains were described as *file systems* (Gregory, 1967) or *massively parallel distributed processors* (McClelland & Rumelhart, 1986; Rumelhart et al., 1986a). Today, terms like file systems and processors perhaps sound a bit outdated, but we have found a new metaphor: the *deep learning metaphor* (e.g. Kriegeskorte, 2015). Excitingly, deep neural networks for computer vision are image-computable, making them possible candidates for models of human visual perception. However, this relationship is far from unidirectional: today, machines are regarded as metaphors and computational models for aspects of human perception (Yamins et al., 2014; Khaligh-Razavi & Kriegeskorte, 2014; Cadieu et al., 2014; Kriegeskorte, 2015; Yamins & DiCarlo, 2016; Kubilius et al., 2016), but at the same time, our understanding of the brain also influences how we perceive machines—albeit a lot less explicitly. This may be facilitated by the fact that we have reached a stage where understanding the machines we build is often no longer possible “by design”. For instance, the architectural primitives from which modern deep neural networks are constructed typically do not suffice to explain or predict their emergent behaviour (Lillicrap & Kording, 2019). Therefore, our interpretation may fall back to what we believe and assume, rather than what we have established scientifically.<sup>4</sup> Here, metaphors come into play, and “while we cannot do without metaphors they can easily betray us” (Smith, 1993, p. 27): the deep learning metaphor of the brain opens the doors for *anthropomorphism*, which is a well-established tendency to ascribe human traits and characteristics to non-humans such as animals or machines. This cognitive bias is the daily bread and butter of those who tell tales and fables, but not even deep learning researchers are immune to its effects (Buckner, 2019). As we will see in this thesis, this can reach the point where we unknowingly use our own perception as models for machine perception. It is all the more important to understand which assumptions humans have when reasoning about machines.

<sup>4</sup>In science, metaphors can be much more than just a figurative comparison. In the extreme case, “consistency with an accepted underlying metaphor goes far towards determining what will and what will not be accepted as an explanation” (quoted from Smith, 1993, p. 284, who credit this thought to Kuhn (1962) studying the history of scientific revolutions).



(6.) *Humans as those who need to understand*

Finally, we need to recall that at its heart, “understanding” is a *human* concept: whether a better understanding of machine decision-making can be achieved is only measurable by human standards. Since in many cases, humans are also those who are affected by machine decisions, being able to understand how a specific decision was reached is not only a scientific, but very much a societal necessity—and, in the European Union, to a certain degree even a legal requirement through the so-called “right to explanation”, which sets high standards for the interpretability of machine decisions (GDPR, 2016; Goodman & Flaxman, 2017).

TAKEN TOGETHER, we have seen that learning machines have a deeply intertwined relationship with humans, whether explicit (humans influence which machine decides, what a machine decides, and how a machine decides) or implicit (humans influence how machine decisions are interpreted). Therefore, the focus of my thesis is the comparison of machine and human decision-making. Measuring exactly where they agree and how they differ will enable us to draw conclusions based on data, rather than implicit assumptions.

### 1.3 *Why study behaviour?*

WHILE THERE ARE many different ways of comparing humans and machines, most early human-machine comparisons have focused on comparing *representations*, such as activity patterns in neural network layers to activity patterns in the brain (e.g. Yamins et al., 2013, 2014; Cadieu et al., 2014; Khaligh-Razavi & Kriegeskorte, 2014; Güçlü & van Gerven, 2015). In contrast, the analysis of *behaviour* (i.e. comparing network decisions to human decisions) has been understudied and not received the attention it deserves. In this thesis, analysing behaviour will be a central methodological approach. As I explain below, analysing behaviour is not only timely but also sensible for a number of principled reasons.

First of all, a machine’s behaviour (or output decision) has the most direct practical relevance: humans are affected by machine decisions, not by “machine thoughts”. If we are to understand decision-making, we are best served by taking the actual decisions, i.e. the output of a system (not just its internal representation), into account. This is exactly what behavioural analyses are designed for.

Second, behavioural comparisons between humans and machines can be achieved without invasive methods such as multielectrode ar-

ray recordings that are often used for comparisons at the representational level. Invasive methods come with ethical challenges, drastically limit the number of experiments one can perform, and in many cases also involve drawing inferences about human perception without actually studying human perception (e.g. via analogies to non-human primates). Moreover, the behavioural approach of studying machines at the output level additionally has the advantage that we can investigate pre-trained networks without any modification or re-training, which limits the influence of ad hoc choices (e.g. training hyperparameters, layer selection) and might even reduce experimenter bias since the number of experimental knobs one needs to turn or tweak becomes much smaller.

Third, the complexity of standard CNNs makes it notoriously difficult to understand how they arrive at a decision, and which aspects of an image determine their behaviour. Illuminating exactly these properties of complex systems is one of the core competencies of psychophysics, a field that has developed rich methods for analysing perceptual decision-making based on complex sensory input (Wichmann & Jäkel, 2018). This creates opportunities for cross-fertilisation by transferring well-established methods from a psychophysicist's toolbox to the analysis of CNNs. "Innovation happens at the fringes, not at the center" (Hall & Yoon, 2017, p. 1), and often there are exciting insights to be gained when the fringes of two originally very different fields start to meet.

Fourth, analysing behaviour is perfectly suited for a *functional* comparison of humans and machines. Kay (2018, p. 7) defines the difference between a functional and a mechanistic model as follows: "a functional model attempts only to match the outputs of a system given the same inputs provided to the system, whereas a mechanistic model attempts to also use components that parallel the actual physical components of the system." While in principle obtaining both mechanistic and functional models of human perception are laudable long-term goals, mechanistic models of biological systems always come with the unavoidable question of what the right level of detail is: Brain areas? Smaller neural circuits? Individual neurons? Neurotransmitters? Or perhaps even individual atoms? There is no agreed-upon answer to this question, but not incorporating certain details risks ending up with a mechanistic model that does not faithfully account for biological processes. On the other hand, attempting to incorporate all details is nearly impossible, and even if one succeeded in this daunting task, one would risk ending up with a "Map of the Empire whose size was that of the Empire" (Borges, 1998, p. 325), i.e. with a model that stops being useful since it is as complex as the original system it was intended to model. Moreover, while CNNs are to a certain degree brain-

inspired, they incorporate numerous design choices made purely for engineering reasons such as ReLU activation functions or the use of back-propagation (e.g. [Werbos, 1974](#); [Rumelhart et al., 1986b](#)), rendering them poor candidates for mechanistic models: if anything, current CNNs will be useful as functional models. In terms of Marr’s levels of analysis ([Marr, 1982](#)), CNNs clearly differ from human perception on the *implementational* level, but it is an open question whether they show similarities on the *algorithmic* level. Analyses at this level seek to understand how a system algorithmically transforms input to output, irrespective of how the computation might be implemented physically or biologically. We know that “all models are wrong” ([Box, 1976](#), p. 792), but it is an important open question whether CNNs will be *useful* as algorithmic/functional models.

In conclusion, there are a number of compelling reasons for the study of behaviour when it comes to comparing human and machine decision-making: the approach is understudied and non-invasive, has direct practical relevance, allows for the transfer of well-established psychophysical methods, and targets the promising functional level of comparison.

#### 1.4 *Why study visual object recognition?*

MACHINE BEHAVIOUR CAN BE COMPARED to human behaviour on many different tasks. In my thesis, I chose to compare them specifically on visual object recognition. Vision is a very special sense: you might still remember the difficulties of learning the first foreign language, or of trying to understand mathematical concepts in school—but I doubt anyone remembers how difficult it was to learn how to recognise objects. While it is hard to tell in hindsight (after all, we were still infants when we “learned to see”), there are good reasons to believe that it might not even have appeared difficult to us in the first place: the human brain is among the most intricate systems that evolution has developed, and even though an adult brain weighs just over 1 kg ([Hartmann et al., 1994](#)), it consumes a remarkable 20% of the energy provided by oxygen ([Kety, 1957](#); [Raichle & Gusnard, 2002](#)). For comparison, imagine an airport where one-fifth of the fuel would be used up to power the control tower alone! In this regard, brains are extremely expensive, and a large part of the primate brain either directly receives visual input or is connected to visual areas ([Felleman & Van Essen, 1991](#)). Consequently, visual tasks like “recognising objects” that seem perfectly easy to us only appear to be effortless since our brains are devoting enormous resources to this end: “We are all prodigious olympians in perceptual and motor areas, so good that we make

the difficult look easy.” (Moravec, 1988, p. 15f.). This has led to difficulties in estimating which tasks are currently feasible for machines, and which ones beyond reach. As a consequence, the term *Moravec’s paradox* was coined: tasks that seem highly advanced and complex for humans, such as playing challenging board games, are comparatively easy for machines to master—while tasks that humans do effortlessly, like recognising objects robustly and reliably, have proven very challenging for machines (Moravec, 1988).<sup>5</sup>

Thus, vision is special, and within vision, object recognition takes on a very important role: “At a functional level, visual object recognition is at the centre of understanding how we think about what we see. Object identification is a primary end state of visual processing and a critical precursor to interacting with and reasoning about the world” (Peissig & Tarr, 2007, p. 76). Given the relevance of object recognition for understanding and interacting with the world around us, it is easy to comprehend the considerable excitement following the breakthrough performance of a machine model, AlexNet (Krizhevsky et al., 2012), on a challenging 1,000-class object recognition task, the ImageNet Large-Scale Visual Recognition Challenge (Russakovsky et al., 2015). For the last decade, object recognition has been at the very forefront of progress: representations learned through object recognition were successfully transferred to other tasks such as instance semantic segmentation (Noh et al., 2015) or saliency prediction (Kümmerer et al., 2015). Furthermore, important advances such as vision transformers, a new type of architecture that does not rely on convolution operations (Dosovitskiy et al., 2020) were first developed for object recognition and later adapted for other tasks. In computer vision, object recognition is continuing to set standards; investigating models trained on object recognition means being able to study the very latest developments on a task that is both well-established and important to the community.

Last but certainly not least, visual object recognition is an area where there have been particularly promising findings (and, occasionally, particularly broad-ranging claims) regarding CNNs as faithful computational models of human object recognition (e.g. Yamins et al., 2013, 2014; Kriegeskorte, 2015; Kubilius et al., 2016, 2019; Zhuang et al., 2021). The combination of these factors—the importance of vision and object recognition for human perception, the central role of object recognition within computer vision, and the fact that CNNs trained on object recognition are being proposed as models for primate ventral stream object recognition—collectively render visual object recognition a very useful (if not the current best) area for comparing human against machine behaviour.

<sup>5</sup> One particularly remarkable example of this paradox is the “Summer Vision Project”, a 1966 MIT project for summer interns tasked with developing a computer vision program capable of recognising objects over the course of a few months (Papert, 1966). Needless to say, the task turned out to be a *tad* more complex than anticipated ... (Hutton, 2011).

## 1.5 Outline

MACHINE LEARNING has a growing impact on our everyday lives, and whether this influence will be net positive depends on our ability to understand machine decision-making. In the introduction I have argued that understanding machine decision-making is more than a purely technical endeavour—machines have a deeply intertwined relationship with humans, and just as we sometimes only truly understand our own culture when living abroad, machine decision-making can best be understood when comparing it to our own human decision-making.

Figure 1.1 presents a schematic overview of the projects in this thesis. Projects P1 and P2 ask: “How do models recognise objects?”. The question is simple, but the answer unexpected: contrary to what was assumed previously (and in stark contrast to human perception), object textures rather than object shapes are the behaviour-determining features in object classification (P1) and object detection (P2). Given that all investigated models are biased towards textures, what sets models apart behaviourally—in other words, “How do models differ from one another?” This is the question that projects P3 and P4 set out to answer. We will see that even radically different models such as feedforward vs. recurrent models (P3) and supervised vs. self-supervised models (P4) systematically make similar errors: behaviourally, most models seem to be created equal (but different from humans). Following these “How” questions, project P5 then presents the concept of shortcut learning as an integrative perspective tackling the question of “Why do machines decide the way they do”. The answer sheds new light on projects P1–P4 since it may explain why models often learn the same strategies, but it also exposes further systematic differences to human behaviour. This leads us to the final question, “How can we make progress?”, which will be (partially) answered by project P6 presenting a comprehensive human-vs-machine toolbox to benchmark and scrutinise object recognition behaviour. Testing a range of promising machine learning developments on this benchmark, we will see that we are finally making progress in closing the gap between human and machine behaviour when it comes to robust visual object recognition.

After presenting these projects in Chapter 2, my thesis will conclude with a general discussion (Chapter 3) and an outlook to some of the big next questions that arise as a result of my experimental findings (Chapter 4).

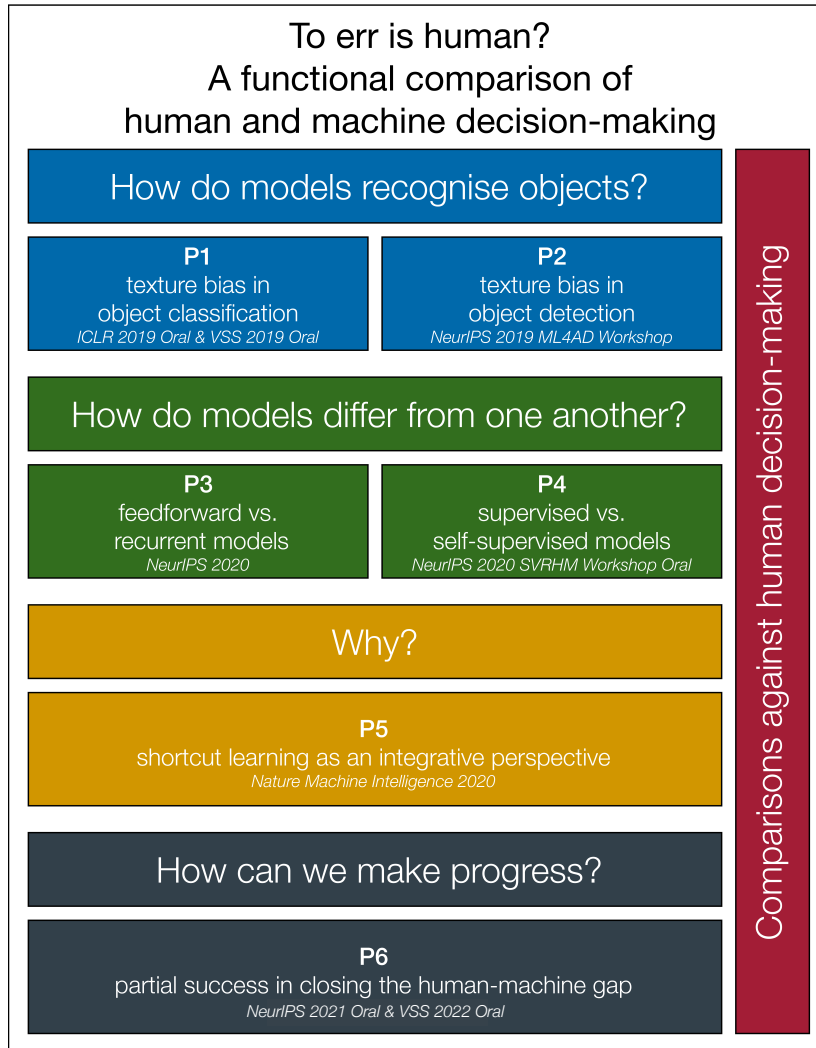


Figure 1.1: Schematic overview of the projects presented in this thesis. Projects P1–P4 ask “How do machines decide”, project P5 presents the concept of shortcut learning as an integrative perspective tackling the question “Why do machines decide the way they do”. Finally, project P6 presents a comprehensive benchmark to measure future progress, and reports first signs of (partial) success. Throughout the thesis, machine decision-making will be compared against human decision-making.

## 2 *Publications*

### 2.1 *ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness*

*Transparency notice* This publication is based, in part, on my MSc thesis titled “Out of shape: quantifying and overcoming texture bias in convolutional neural networks” (University of Tübingen, 2018). The MSc thesis was written as a publication draft. Based on this draft, a number of changes and additions were implemented afterwards, including (but not limited to): completely new structure; visualisation improvements to main figures; text changes to title, abstract, main paper and appendix; more concise presentation of datasets; making code, datasets & weights publicly available; incorporating robustness results for networks trained on Stylized-ImageNet when tested on ImageNet-C (Hendrycks & Dietterich, 2019); adding results for a very deep network (ResNet-152), a very wide network (DenseNet-121) and a highly compressed network (SqueezeNet1\_1); testing a network trained on a different dataset (Open Images); training AlexNet and VGG-16 on Stylized-ImageNet to make sure the results are not limited to ResNet-50; including results for transfer learning on a different object detection data set (MS COCO); testing accuracy and object detection performance for ResNet-152 (a much deeper object detection backbone); enhancing the presentation of claims and contributions; performing a reaction time analysis; investigating a correlation between accuracy on “edge” images and texture bias.

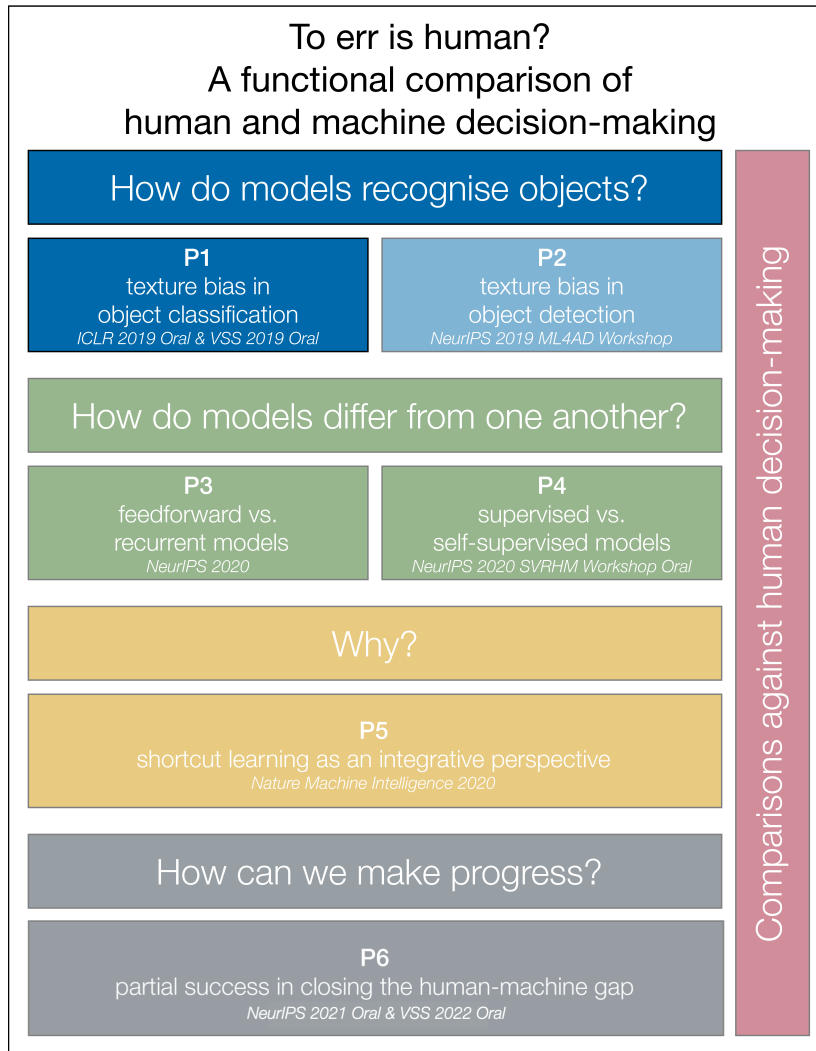


Figure 2.1: Schematic overview of the projects presented in this thesis. Projects P1–P4 ask “How do machines decide”, project P5 presents the concept of shortcut learning as an integrative perspective tackling the question “Why do machines decide the way they do”. Finally, project P6 presents a comprehensive benchmark to measure future progress, and reports first signs of (partial) success. Throughout the thesis, machine decision making will be compared against human decision making.



Published as a conference paper at ICLR 2019

# IMAGENET-TRAINED CNNs ARE BIASED TOWARDS TEXTURE; INCREASING SHAPE BIAS IMPROVES ACCURACY AND ROBUSTNESS

**Robert Geirhos**

University of Tübingen & IMPRS-IS  
robert.geirhos@bethgelab.org

**Patricia Rubisch**

University of Tübingen & U. of Edinburgh  
p.rubisch@sms.ed.ac.uk

**Claudio Michaelis**

University of Tübingen & IMPRS-IS  
claudio.michaelis@bethgelab.org

**Matthias Bethge\***

University of Tübingen  
matthias.bethge@bethgelab.org

**Felix A. Wichmann\***

University of Tübingen  
felix.wichmann@uni-tuebingen.de

**Wieland Brendel\***

University of Tübingen  
wieland.brendel@bethgelab.org

## ABSTRACT

Convolutional Neural Networks (CNNs) are commonly thought to recognise objects by learning increasingly complex representations of object shapes. Some recent studies suggest a more important role of image textures. We here put these conflicting hypotheses to a quantitative test by evaluating CNNs and human observers on images with a texture-shape cue conflict. We show that ImageNet-trained CNNs are strongly biased towards recognising textures rather than shapes, which is in stark contrast to human behavioural evidence and reveals fundamentally different classification strategies. We then demonstrate that the same standard architecture (ResNet-50) that learns a texture-based representation on ImageNet is able to learn a shape-based representation instead when trained on ‘Stylized-ImageNet’, a stylized version of ImageNet. This provides a much better fit for human behavioural performance in our well-controlled psychophysical lab setting (nine experiments totalling 48,560 psychophysical trials across 97 observers) and comes with a number of unexpected emergent benefits such as improved object detection performance and previously unseen robustness towards a wide range of image distortions, highlighting advantages of a shape-based representation.

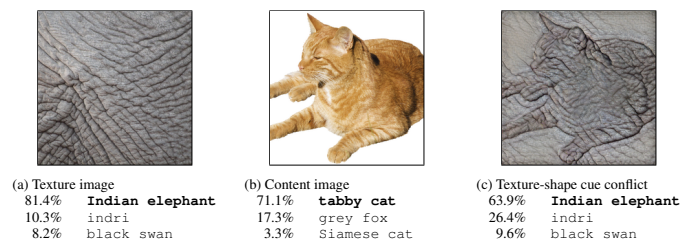


Figure 1: Classification of a standard ResNet-50 of (a) a texture image (elephant skin: only texture cues); (b) a normal image of a cat (with both shape and texture cues), and (c) an image with a texture-shape cue conflict, generated by style transfer between the first two images.

\* Joint senior authors

Published as a conference paper at ICLR 2019

## 1 INTRODUCTION

How are Convolutional Neural Networks (CNNs) able to reach impressive performance on complex perceptual tasks such as object recognition (Krizhevsky et al., 2012) and semantic segmentation (Long et al., 2015)? One widely accepted intuition is that CNNs combine low-level features (e.g. edges) to increasingly complex shapes (such as wheels, car windows) until the object (e.g. car) can be readily classified. As Kriegeskorte (2015) puts it, “the network acquires complex knowledge about the kinds of shapes associated with each category. [...] High-level units appear to learn representations of shapes occurring in natural images” (p. 429). This notion also appears in other explanations, such as in LeCun et al. (2015): Intermediate CNN layers recognise “parts of familiar objects, and subsequent layers [...] detect objects as combinations of these parts” (p. 436). We term this explanation the *shape hypothesis*.

This hypothesis is supported by a number of empirical findings. Visualisation techniques like Deconvolutional Networks (Zeiler & Fergus, 2014) often highlight object parts in high-level CNN features.<sup>1</sup> Moreover, CNNs have been proposed as computational models of human shape perception by Kubilius et al. (2016), who conducted an impressive number of experiments comparing human and CNN shape representations and concluded that CNNs “implicitly learn representations of shape that reflect human shape perception” (p. 15). Ritter et al. (2017) discovered that CNNs develop a so-called “shape bias” just like children, i.e. that object shape is more important than colour for object classification (although see Hosseini et al. (2018) for contrary evidence). Furthermore, CNNs are currently the most predictive models for human ventral stream object recognition (e.g. Cadieu et al., 2014; Yamins et al., 2014); and it is well-known that object shape is the single most important cue for human object recognition (Landau et al., 1988), much more than other cues like size or texture (which may explain the ease at which humans recognise line drawings or millennia-old cave paintings).

On the other hand, some rather disconnected findings point to an important role of object textures for CNN object recognition. CNNs can still classify texturised images perfectly well, even if the global shape structure is completely destroyed (Gatys et al., 2017; Brendel & Bethge, 2019). Conversely, standard CNNs are bad at recognising object sketches where object shapes are preserved yet all texture cues are missing (Ballester & de Araújo, 2016). Additionally, two studies suggest that local information such as textures may actually be sufficient to “solve” ImageNet object recognition: Gatys et al. (2015) discovered that a linear classifier on top of a CNN’s texture representation (Gram matrix) achieves hardly any classification performance loss compared to original network performance. More recently, Brendel & Bethge (2019) demonstrated that CNNs with explicitly constrained receptive field sizes throughout all layers are able to reach surprisingly high accuracies on ImageNet, even though this effectively limits a model to recognising small local patches rather than integrating object parts for shape recognition. Taken together, it seems that local textures indeed provide sufficient information about object classes—ImageNet object recognition *could*, in principle, be achieved through texture recognition alone. In the light of these findings, we believe that it is time to consider a second explanation, which we term the *texture hypothesis*: in contrast to the common assumption, object textures are more important than global object shapes for CNN object recognition.

Resolving these two contradictory hypotheses is important both for the deep learning community (to increase our understanding of neural network decisions) as well as for the human vision and neuroscience communities (where CNNs are being used as computational models of human object recognition and shape perception). In this work we aim to shed light on this debate with a number of carefully designed yet relatively straightforward experiments. Utilising style transfer (Gatys et al., 2016), we created images with a texture-shape cue conflict such as the cat shape with elephant texture depicted in Figure 1c. This enables us to quantify texture and shape biases in both humans and CNNs. To this end, we perform nine comprehensive and careful psychophysical experiments comparing humans against CNNs on exactly the same images, totalling 48,560 psychophysical trials across 97 observers. These experiments provide behavioural evidence in favour of the texture hypothesis: A cat with an elephant texture is an elephant to CNNs, and still a cat to humans. Beyond quantifying existing biases, we subsequently present results for our two other main contributions:

<sup>1</sup>To avoid any confusion caused by different meanings of the term ‘feature’, we consistently use it to refer to properties of CNNs (learned features) rather than to object properties (such as colour). When referring to physical objects, we use the term ‘cue’ instead.

Published as a conference paper at ICLR 2019

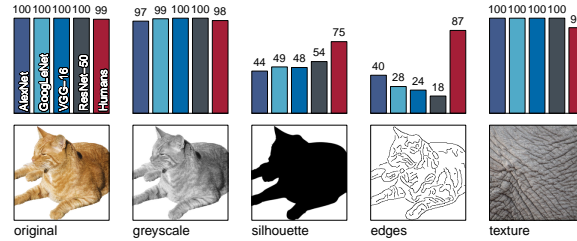


Figure 2: Accuracies and example stimuli for five different experiments without cue conflict.

changing biases, and discovering emergent benefits of changed biases. We show that the texture bias in standard CNNs can be overcome and changed towards a shape bias if trained on a suitable data set. Remarkably, networks with a higher shape bias are inherently more robust to many different image distortions (for some even reaching or surpassing human performance, *despite never being trained on any of them*) and reach higher performance on classification and object recognition tasks.

## 2 METHODS

In this section we outline the core elements of paradigm and procedure. Extensive details to facilitate replication are provided in the Appendix. Data, code and materials are available from this repository: <https://github.com/rgeirhos/texture-vs-shape>

### 2.1 PSYCHOPHYSICAL EXPERIMENTS

All psychophysical experiments were conducted in a well-controlled psychophysical lab setting and follow the paradigm of Geirhos et al. (2018), which allows for direct comparisons between human and CNN classification performance on exactly the same images. Briefly, in each trial participants were presented a fixation square for 300 ms, followed by a 300 ms presentation of the stimulus image. After the stimulus image we presented a full-contrast pink noise mask ( $1/f$  spectral shape) for 200 ms to minimise feedback processing in the human visual system and to thereby make the comparison to feedforward CNNs as fair as possible. Subsequently, participants had to choose one of 16 entry-level categories by clicking on a response screen shown for 1500 ms. On this screen, icons of all 16 categories were arranged in a  $4 \times 4$  grid. Those categories were airplane, bear, bicycle, bird, boat, bottle, car, cat, chair, clock, dog, elephant, keyboard, knife, oven and truck. Those are the so-called “16-class-ImageNet” categories introduced in Geirhos et al. (2018).

The same images were fed to four CNNs pre-trained on standard ImageNet, namely AlexNet (Krizhevsky et al., 2012), GoogLeNet (Szegedy et al., 2015), VGG-16 (Simonyan & Zisserman, 2015) and ResNet-50 (He et al., 2015). The 1,000 ImageNet class predictions were mapped to the 16 categories using the WordNet hierarchy (Miller, 1995)—e.g. ImageNet category *tabby cat* would be mapped to *cat*. In total, the results presented in this study are based on 48,560 psychophysical trials and 97 participants.

### 2.2 DATA SETS (PSYCHOPHYSICS)

In order to assess texture and shape biases, we conducted six major experiments along with three control experiments, which are described in the Appendix. The first five experiments (samples visualised in Figure 2) are simple object recognition tasks with the only difference being the image features available to the participant:

**Original** 160 natural colour images of objects (10 per category) with white background.

Published as a conference paper at ICLR 2019

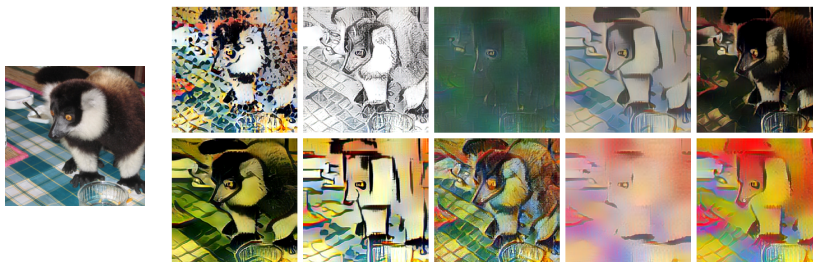


Figure 3: Visualisation of Stylized-ImageNet (SIN), created by applying AdaIN style transfer to ImageNet images. Left: randomly selected ImageNet image of class `ring-tailed lemur`. Right: ten examples of images with content/shape of left image and style/texture from different paintings. After applying AdaIN style transfer, local texture cues are no longer highly predictive of the target class, while the global shape tends to be retained. Note that within SIN, every source image is stylized only once.

**Greyscale** Images from *Original* data set converted to greyscale using `skimage.color.rgb2gray`. For CNNs, greyscale images were stacked along the colour channel.

**Silhouette** Images from *Original* data set converted to silhouette images showing an entirely black object on a white background (see Appendix A.6 for procedure).

**Edges** Images from *Original* data set converted to an edge-based representation using `Canny edge extractor` implemented in MATLAB.

**Texture** 48 natural colour images of textures (3 per category). Typically the textures consist of full-width patches of an animal (e.g. skin or fur) or, in particular for man-made objects, of images with many repetitions of the same objects (e.g. many bottles next to each other, see Figure 7 in the Appendix).

It is important to note that we only selected object and texture images that were correctly classified by all four networks. This was made to ensure that our results in the sixth experiment on cue conflicts, which is most decisive in terms of the shape vs texture hypothesis, are fully interpretable. In the cue conflict experiment we present images with contradictory features (see Figure 1) but still ask the participant to assign a single class. Note that the instructions to human observers were entirely neutral w.r.t. shape or texture (“click on the object category that you see in the presented image; guess if unsure. There is no right or wrong answer, we are interested in your subjective impression”).

**Cue conflict** Images generated using iterative style transfer (Gatys et al., 2016) between an image of the *Texture* data set (as style) and an image from the *Original* data set (as content). We generated a total of 1280 cue conflict images (80 per category), which allows for presentation to human observers within a single experimental session.

We define “silhouette” as the bounding contour of an object in 2D (i.e., the outline of object segmentation). When mentioning “object shape”, we use a definition that is broader than just the silhouette of an object: we refer to the set of contours that describe the 3D form of an object, i.e. including those contours that are not part of the silhouette. Following Gatys et al. (2017), we define “texture” as an image (region) with spatially stationary statistics. Note that on a very local level, textures (according to this definition) can have non-stationary elements (such as a local shape): e.g. a single bottle clearly has non-stationary statistics, but many bottles next to each other are perceived as a texture: “things” become “stuff” (Gatys et al., 2017, p. 178). For an example of a “bottle texture” see Figure 7.

Published as a conference paper at ICLR 2019

### 2.3 STYLIZED-IMAGENET

Starting from ImageNet we constructed a new data set (termed Stylized-ImageNet or SIN) by stripping every single image of its original texture and replacing it with the style of a randomly selected painting through AdaIN style transfer (Huang & Belongie, 2017) (see examples in Figure 3) with a stylization coefficient of  $\alpha = 1.0$ . We used Kaggle’s *Painter by Numbers* data set<sup>2</sup> as a style source due to its large style variety and size (79,434 paintings). We used AdaIN fast style transfer rather than iterative stylization (e.g. Gatys et al., 2016) for two reasons: Firstly, to ensure that training on SIN and testing on cue conflict stimuli is done using different stylization techniques, such that the results do not rely on a single stylization method. Secondly, to enable stylizing entire ImageNet, which would take prohibitively long with an iterative approach. We provide code to create Stylized-ImageNet here:

<https://github.com/rgeirhos/Stylized-ImageNet>

## 3 RESULTS

### 3.1 TEXTURE VS SHAPE BIAS IN HUMANS AND IMAGENET-TRAINED CNNs

Almost all object and texture images (*Original* and *Texture* data set) were recognised correctly by both CNNs and humans (Figure 2). Greyscale versions of the objects, which still contain both shape and texture, were recognised equally well. When object outlines were filled in with black colour to generate a silhouette, CNN recognition accuracies were much lower than human accuracies. This was even more pronounced for edge stimuli, indicating that human observers cope much better with images that have little to no texture information. One confound in these experiments is that CNNs tend not to cope well with domain shifts, i.e. the large change in image statistics from natural images (on which the networks have been trained) to sketches (which the networks have never seen before).

We thus devised a cue conflict experiment that is based on images with a natural statistic but contradicting texture and shape evidence (see Methods). Participants and CNNs have to classify the images based on the features (shape or texture) that they most rely on. The results of this experiment are visualised in Figure 4. Human observers show a striking bias towards responding with the shape category (95.9% of correct decisions).<sup>3</sup> This pattern is reversed for CNNs, which show a clear bias towards responding with the texture category (VGG-16: 17.2% shape vs. 82.8% texture; GoogLeNet: 31.2% vs. 68.8%; AlexNet: 42.9% vs. 57.1%; ResNet-50: 22.1% vs. 77.9%).

### 3.2 OVERCOMING THE TEXTURE BIAS OF CNNs

The psychophysical experiments suggest that ImageNet-trained CNNs, but not humans, exhibit a strong texture bias. One reason might be the training task itself: from Brendel & Bethge (2019) we know that ImageNet can be solved to high accuracy using only local information. In other words, it might simply suffice to integrate evidence from many local texture features rather than going through the process of integrating and classifying global shapes. In order to test this hypothesis we train a ResNet-50 on our Stylized-ImageNet (SIN) data set in which we replaced the object-related local texture information with the uninformative style of randomly selected artistic paintings.

A standard ResNet-50 trained and evaluated on Stylized-ImageNet (SIN) achieves 79.0% top-5 accuracy (see Table 1). In comparison, the same architecture trained and evaluated on ImageNet (IN) achieves 92.9% top-5 accuracy. This performance difference indicates that SIN is a much harder task than IN since textures are no longer predictive, but instead a nuisance factor (as desired). Intriguingly, ImageNet features generalise poorly to SIN (only 16.4% top-5 accuracy); yet features learned on SIN generalise very well to ImageNet (82.6% top-5 accuracy without any fine-tuning).

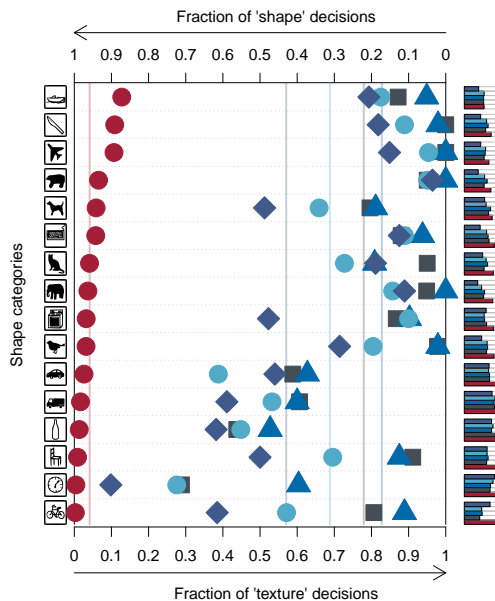
In order to test whether local texture features are still sufficient to “solve” SIN we evaluate the performance of so-called *BagNets*. Introduced recently by Brendel & Bethge (2019), BagNets have a ResNet-50 architecture but their maximum receptive field size is limited to  $9 \times 9$ ,  $17 \times 17$  or  $33 \times 33$

<sup>2</sup><https://www.kaggle.com/c/painter-by-numbers/> (accessed on March 1, 2018).

<sup>3</sup>It is important to note that a substantial fraction of the images (automatically generated with style transfer between randomly selected object image and texture image) seemed hard to recognise for both humans and CNNs, as depicted by the fraction of incorrect classification choices in Figure 4.

Published as a conference paper at ICLR 2019

Figure 4: Classification results for human observers (red circles) and ImageNet-trained networks AlexNet (purple diamonds), VGG-16 (blue triangles), GoogLeNet (turquoise circles) and ResNet-50 (grey squares). Shape vs. texture biases for stimuli with cue conflict (sorted by human shape bias). Within the responses that corresponded to either the correct texture or correct shape category, the fractions of texture and shape decisions are depicted in the main plot (averages visualised by vertical lines). On the right side, small barplots display the proportion of correct decisions (either texture or shape correctly recognised) as a fraction of all trials. Similar results for ResNet-152, DenseNet-121 and Squeezenet1.1 are reported in the Appendix, Figure 13.



pixels. This precludes BagNets from learning or using any long-range spatial relationships for classification. While these restricted networks can reach high accuracies on ImageNet, they are unable to achieve the same on SIN, showing dramatically reduced performance with smaller receptive field sizes (such as 10.0% top-5 accuracy on SIN compared to 70.0% on ImageNet for a BagNet with receptive field size of  $9 \times 9$  pixels). This is a clear indication that the SIN data set we propose does actually remove local texture cues, forcing a network to integrate long-range spatial information.

Most importantly, the SIN-trained ResNet-50 shows a much stronger shape bias in our cue conflict experiment (Figure 5), which increases from 22% for a IN-trained model to 81%. In many categories the shape bias is almost as strong as for humans.

### 3.3 ROBUSTNESS AND ACCURACY OF SHAPE-BASED REPRESENTATIONS

Does the increased shape bias, and thus the shifted representations, also affect the performance or robustness of CNNs? In addition to the IN- and SIN-trained ResNet-50 architecture we here additionally analyse two joint training schemes:

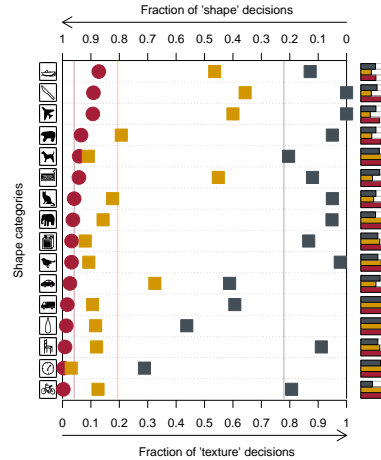
- Training jointly on SIN and IN.
- Training jointly on SIN and IN with fine-tuning on IN. We refer to this model as *Shape-ResNet*.

architecture	IN→IN	IN→SIN	SIN→SIN	SIN→IN
ResNet-50	92.9	16.4	79.0	82.6
BagNet-33 (mod. ResNet-50)	86.4	4.2	48.9	53.0
BagNet-17 (mod. ResNet-50)	80.3	2.5	29.3	32.6
BagNet-9 (mod. ResNet-50)	70.0	1.4	10.0	10.9

Table 1: Stylized-ImageNet cannot be solved with texture features alone. Accuracy comparison (in percent; top-5 on validation data set) of a standard ResNet-50 with Bag of Feature networks (BagNets) with restricted receptive field sizes of  $33 \times 33$ ,  $17 \times 17$  and  $9 \times 9$  pixels. Arrows indicate: train data→test data, e.g. IN→SIN means training on ImageNet and testing on Stylized-ImageNet.

Published as a conference paper at ICLR 2019

Figure 5: Shape vs. texture biases for stimuli with a texture-shape cue conflict after training ResNet-50 on Stylized-ImageNet (orange squares) and on ImageNet (grey squares). Plotting conventions and human data are identical to Figure 4. Similar results for other networks are reported in the Appendix, Figure 11.



name	training	fine-tuning	top-1 IN accuracy (%)	top-5 IN accuracy (%)	Pascal VOC mAP50 (%)	MS COCO mAP50 (%)
vanilla ResNet	IN	-	76.13	92.86	70.7	52.3
	SIN	-	60.18	82.62	70.6	51.9
	SIN+IN	-	74.59	92.14	74.0	53.8
Shape-ResNet	SIN+IN	IN	<b>76.72</b>	<b>93.28</b>	<b>75.1</b>	<b>55.2</b>

Table 2: Accuracy comparison on the ImageNet (IN) validation data set as well as object detection performance (mAP50) on PASCAL VOC 2007 and MS COCO. All models have an identical ResNet-50 architecture. Method details reported in the Appendix, where we also report similar results for ResNet-152 (Table 4).

We then compared these models with a vanilla ResNet-50 on three experiments: (1) classification performance on IN, (2) transfer to Pascal VOC 2007 and (3) robustness against image perturbations.

**Classification performance** Shape-ResNet surpasses the vanilla ResNet in terms of top-1 and top-5 ImageNet validation accuracy as reported in Table 2. This indicates that SIN may be a useful data augmentation on ImageNet that can improve model performance without any architectural changes.

**Transfer learning** We tested the representations of each model as backbone features for Faster R-CNN (Ren et al., 2017) on Pascal VOC 2007 and MS COCO. Incorporating SIN in the training data substantially improves object detection performance from 70.7 to 75.1 mAP50 (52.3 to 55.2 mAP50 on MS COCO) as shown in Table 2. This is in line with the intuition that for object detection, a shape-based representation is more beneficial than a texture-based representation, since the ground truth rectangles encompassing an object are by design aligned with global object shape.

**Robustness against distortions** We systematically tested how model accuracies degrade if images are distorted by uniform or phase noise, contrast changes, high- and low-pass filtering or eidolon perturbations.<sup>4</sup> The results of this comparison, including human data for reference, are visualised in Figure 6. While lacking a few percent accuracy on undistorted images, the SIN-trained network outperforms the IN-trained CNN on almost all image manipulations. (Low-pass filtering / blurring is the only distortion type on which SIN-trained networks are more susceptible, which might be due to the over-representation of high frequency signals in SIN through paintings and the reliance on

<sup>4</sup>Our comparison encompasses all distortions reported by Geirhos et al. (2018) with more than five different levels of signal strength. Data from human observers included with permission from the authors (see appendix).

Published as a conference paper at ICLR 2019

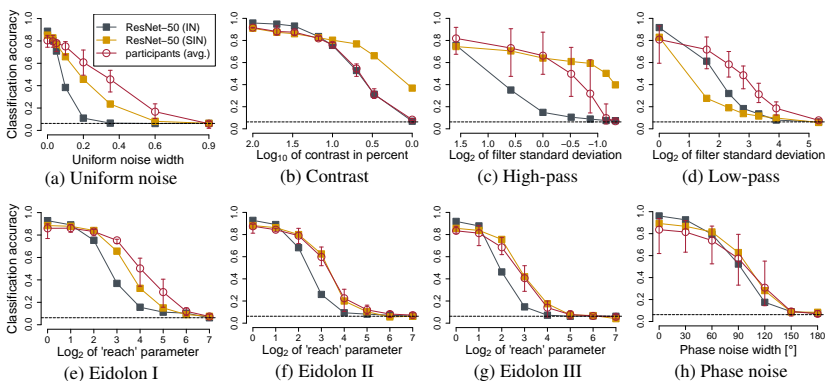


Figure 6: Classification accuracy on parametrically distorted images. ResNet-50 trained on Stylized-ImageNet (SIN) is more robust towards distortions than the same network trained on ImageNet (IN).

sharp edges.) The SIN-trained ResNet-50 approaches human-level distortion robustness—*despite never seeing any of the distortions during training.*

Furthermore, we provide robustness results for our models tested on ImageNet-C, a comprehensive benchmark of 15 different image corruptions (Hendrycks & Dietterich, 2019), in Table 5 of the Appendix. Training jointly on SIN and IN leads to strong improvements for 13 corruption types (Gaussian, Shot and Impulse noise; Defocus, Glas and Motion blur; Snow, Frost and Fog weather types; Contrast, Elastic, Pixelate and JPEG digital corruptions). This substantially reduces overall corruption error from 76.7 for a vanilla ResNet-50 to 69.3. Again, none of these corruption types were explicitly part of the training data, reinforcing that incorporating SIN in the training regime improves model robustness in a very general way.

#### 4 DISCUSSION

As noted in the Introduction, there seems to be a large discrepancy between the common assumption that CNNs use increasingly complex shape features to recognise objects and recent empirical findings which suggest a crucial role of object textures instead. In order to explicitly probe this question, we utilised style transfer (Gatys et al., 2016) to generate images with conflicting shape and texture information. On the basis of extensive experiments on both CNNs and human observers in a controlled psychophysical lab setting, we provide evidence that unlike humans, ImageNet-trained CNNs tend to classify objects according to local textures instead of global object shapes. In combination with previous work which showed that changing other major object dimensions such as colour (Geirhos et al., 2018) and object size relative to the context (Eckstein et al., 2017) do not have a strong detrimental impact on CNN recognition performance, this highlights the special role that local cues such as textures seem to play in CNN object recognition.

Intriguingly, this offers an explanation for a number of rather disconnected findings: CNNs match texture appearance for humans (Wallis et al., 2017), and their predictive power for neural responses along the human ventral stream appears to be largely due to human-like texture representations, but not human-like contour representations (Laskar et al., 2018; Long & Konkle, 2018). Furthermore, texture-based generative modelling approaches such as style transfer (Gatys et al., 2016), single image super-resolution (Gondal et al., 2018) as well as static and dynamic texture synthesis (Gatys et al., 2015; Funke et al., 2017) all produce excellent results using standard CNNs, while CNN-based shape transfer seems to be very difficult (Gokaslan et al., 2018). CNNs can still recognise images with scrambled shapes (Gatys et al., 2017; Brendel & Bethge, 2019), but they have much more difficulties recognising objects with missing texture information (Ballester & de Araújo, 2016; Yu et al., 2017). Our hypothesis might also explain why an image segmentation model trained on a database of synthetic texture images transfers to natural images and videos (Ustyuzhaninov et al.,



Published as a conference paper at ICLR 2019

2018). Beyond that, our results show marked behavioural differences between ImageNet-trained CNNs and human observers. While both human and machine vision systems achieve similarly high accuracies on standard images (Geirhos et al., 2018), our findings suggest that the underlying classification strategies might actually be very different. This is problematic, since CNNs are being used as computational models for human object recognition (e.g. Cadieu et al., 2014; Yamins et al., 2014).

In order to reduce the texture bias of CNNs we introduced Stylized-ImageNet (SIN), a data set that removes local cues through style transfer and thereby forces networks to go beyond texture recognition. Using this data set, we demonstrated that a ResNet-50 architecture can indeed learn to recognise objects based on object shape, revealing that the texture bias in current CNNs is not by design but induced by ImageNet training data. This indicates that standard ImageNet-trained models may be taking a “shortcut” by focusing on local textures, which could be seen as a version of Occam’s razor: If textures are sufficient, why should a CNN learn much else? While texture classification may be easier than shape recognition, we found that shape-based features trained on SIN generalise well to natural images.

Our results indicate that a more shape-based representation can be beneficial for recognition tasks that rely on pre-trained ImageNet CNNs. Furthermore, while ImageNet-trained CNNs generalise poorly towards a wide range of image distortions (e.g. Dodge & Karam, 2017; Geirhos et al., 2017; 2018), our ResNet-50 trained on Stylized-ImageNet often reaches or even surpasses human-level robustness (without ever being trained on the specific image degradations). This is exciting because Geirhos et al. (2018) showed that networks trained on specific distortions in general do not acquire robustness against other unseen image manipulations. This emergent behaviour highlights the usefulness of a shape-based representation: While local textures are easily distorted by all sorts of noise (including those in the real world, such as rain and snow), the object shape remains relatively stable. Furthermore, this finding offers a compellingly simple explanation for the incredible robustness of humans when coping with distortions: a shape-based representation.

## 5 CONCLUSION

In summary, we provided evidence that machine recognition today overly relies on object textures rather than global object shapes as commonly assumed. We demonstrated the advantages of a shape-based representation for robust inference (using our Stylized-ImageNet data set<sup>5</sup> to induce such a representation in neural networks). We envision our findings as well as our openly available model weights, code and behavioural data set (49K trials across 97 observers)<sup>6</sup> to achieve three goals: Firstly, an improved understanding of CNN representations and biases. Secondly, a step towards more plausible models of human visual object recognition. Thirdly, a useful starting point for future undertakings where domain knowledge suggests that a shape-based representation may be more beneficial than a texture-based one.

### ACKNOWLEDGMENTS

This work has been funded, in part, by the German Research Foundation (DFG; Sachbeihilfe Wi 2103/4-1 and SFB 1233 on “Robust Vision”). The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting R.G. and C.M.; M.B. acknowledges support by the Centre for Integrative Neuroscience Tübingen (EXC 307) and by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/Interior Business Center (DoI/IBC) contract number D16PC00003.

We would like to thank Dan Hendrycks for providing the results of Table 5 (corruption robustness of our models on ImageNet-C). Furthermore, we would like to express our gratitude towards Alexander Ecker, Leon Gatys, Tina Gauger, Silke Gramer, Heike König, Jonas Rauber, Steffen Schneider, Heiko Schütt, Tom Wallis and Uli Wannek for support and/or useful discussions.

<sup>5</sup>Available from <https://github.com/rgeirhos/Stylized-ImageNet>

<sup>6</sup>Available from <https://github.com/rgeirhos/texture-vs-shape>

Published as a conference paper at ICLR 2019

---

## REFERENCES

- Pedro Ballester and Ricardo Matsumura de Araújo. On the performance of GoogLeNet and AlexNet applied to sketches. In *AAAI*, pp. 1124–1128, 2016.
- Wieland Brendel and Matthias Bethge. Approximating CNNs with bag-of-local-features models works surprisingly well on ImageNet. In *International Conference on Learning Representations*, 2019.
- Charles F Cadieu, H Hong, D L K Yamins, N Pinto, D Ardila, E A Solomon, N J Majaj, and J J DiCarlo. Deep neural networks rival the representation of primate IT cortex for core visual object recognition. *PLoS Computational Biology*, 10(12), 2014.
- Samuel Dodge and Lina Karam. A study and comparison of human and deep learning recognition performance under visual distortions. arXiv preprint arXiv:1705.02498, 2017.
- Miguel P Eckstein, Kathryn Koehler, Lauren E Welbourne, and Emre Akbas. Humans, but not deep neural networks, often miss giant targets in scenes. *Current Biology*, 27(18):2827–2832, 2017.
- Christina M Funke, Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Synthesising dynamic textures using convolutional neural networks. arXiv preprint arXiv:1702.07006, 2017.
- Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Texture synthesis using convolutional neural networks. In *Advances in Neural Information Processing Systems*, pp. 262–270, 2015.
- Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2414–2423, 2016.
- Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Texture and art with deep neural networks. *Current Opinion in Neurobiology*, 46:178–186, 2017.
- Robert Geirhos, David HJ Janssen, Heiko H Schütt, Jonas Rauber, Matthias Bethge, and Felix A Wichmann. Comparing deep neural networks against humans: object recognition when the signal gets weaker. arXiv preprint arXiv:1706.06969, 2017.
- Robert Geirhos, Carlos M. Medina Temme, Jonas Rauber, Heiko H Schütt, Matthias Bethge, and Felix A Wichmann. Generalisation in humans and deep neural networks. arXiv preprint arXiv:1808.08750, 2018.
- Aaron Gokaslan, Vivek Ramanujan, Daniel Ritchie, Kwang In Kim, and James Tompkin. Improving shape deformation in unsupervised image-to-image translation. arXiv preprint arXiv:1808.04325, 2018.
- Muhammad W Gondal, Bernhard Schölkopf, and Michael Hirsch. The unreasonable effectiveness of texture transfer for single image super-resolution. arXiv preprint arXiv:1808.00043, 2018.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. arXiv preprint arXiv:1512.03385, 2015.
- Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019.
- Hossein Hosseini, Baicen Xiao, Mayoore Jaiswal, and Radha Poovendran. Assessing shape bias property of Convolutional Neural Networks. arXiv preprint arXiv:1803.07739, 2018.
- Xun Huang and Serge J Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*, pp. 1510–1519, 2017.
- Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM International Conference on Multimedia*, pp. 675–678. ACM, 2014.

Published as a conference paper at ICLR 2019

- Ivan Krasin, Tom Duerig, Neil Alldrin, Vittorio Ferrari, Sami Abu-El-Hajja, Alina Kuznetsova, Hassan Rom, Jasper Uijlings, Stefan Popov, Andreas Veit, Serge Belongie, Victor Gomes, Abhinav Gupta, Chen Sun, Gal Chechik, David Cai, Zheyun Feng, Dhyanesh Narayanan, and Kevin Murphy. OpenImages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from <https://github.com/openimages>*, 2017.
- N. Kriegeskorte. Deep neural networks: A new framework for modeling biological vision and brain information processing. *Annual Review of Vision Science*, 1(15):417–446, 2015.
- A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pp. 1097–1105, 2012.
- Jonas Kubilius, Stefania Bracci, and Hans P Op de Beeck. Deep neural networks as a computational model for human shape sensitivity. *PLoS Computational Biology*, 12(4):e1004896, 2016.
- Barbara Landau, Linda B Smith, and Susan S Jones. The importance of shape in early lexical learning. *Cognitive Development*, 3(3):299–321, 1988.
- Md Nasir Uddin Laskar, Luis G Sanchez Giraldo, and Odelia Schwartz. Correspondence of deep neural networks and the brain for visual textures. arXiv preprint arXiv:1806.02888, 2018.
- Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.
- Bria Long and Talia Konkle. The role of textural statistics vs. outer contours in deep CNN and neural responses to objects. [http://konklab.fas.harvard.edu/ConferenceProceedings/Long\\_2018\\_CCN.pdf](http://konklab.fas.harvard.edu/ConferenceProceedings/Long_2018_CCN.pdf), 2018.
- Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431–3440, 2015.
- George A Miller. WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41, 1995.
- Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch, 2017.
- Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, pp. 1137–1149, 2017.
- Samuel Ritter, David GT Barrett, Adam Santoro, and Matt M Botvinick. Cognitive psychology for deep neural networks: A shape bias case study. arXiv preprint arXiv:1706.08606, 2017.
- Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2015.
- Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–9, 2015.
- Ivan Ustyuzhaninov, Claudio Michaelis, Wieland Brendel, and Matthias Bethge. One-shot texture segmentation. arXiv preprint arXiv:1807.02654, 2018.
- Thomas SA Wallis, Christina M Funke, Alexander S Ecker, Leon A Gatys, Felix A Wichmann, and Matthias Bethge. A parametric texture model based on deep convolutional features closely matches texture appearance for humans. *Journal of Vision*, 17(12):5–5, 2017.
- Daniel LK Yamins, Ha Hong, Charles F Cadieu, Ethan A Solomon, Darren Seibert, and James J DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, 111(23):8619–8624, 2014.

Published as a conference paper at ICLR 2019

Qian Yu, Yongxin Yang, Feng Liu, Yi-Zhe Song, Tao Xiang, and Timothy M Hospedales. Sketch-a-net: A deep neural network that beats humans. *International Journal of Computer Vision*, 122(3):411–425, 2017.

Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*, pp. 818–833. Springer, 2014.

## A APPENDIX

### A.1 REPRODUCIBILITY & ACCESS TO CODE / MODELS / DATA

In this Appendix, we report experimental details for human and CNN experiments. All trained model weights reported in this paper as well as our human behavioural data set (48,560 psychophysical trials across 97 observers) are openly available from this repository:  
<https://github.com/rgeirhos/texture-vs-shape>

### A.2 PROCEDURE

We followed the paradigm of Geirhos et al. (2018) for maximal comparability. A trial consisted of 300 ms presentation of a fixation square and a 200 ms presentation of the stimulus image, which was followed by a full-contrast pink noise mask ( $1/f$  spectral shape) of the same size lasting for 200 ms. Participants had to choose one of 16 entry-level categories by clicking on a response screen shown for 1500 ms. On this screen, icons of all 16 categories were arranged in a  $4 \times 4$  grid. The experiments were not self-paced and therefore one trial always lasted 2200 ms (300 ms + 200 ms + 200 ms + 1500 ms = 2200 ms). The necessary time to complete an experiment with 1280 stimuli was 47 minutes, for 160 stimuli six minutes, and for 48 stimuli two minutes. In the experiments with 1280 trials, observers were given the possibility of taking a brief break after every block of 256 trials (five blocks in total).

As preparation, participants were shown the response screen prior to an experiment and were asked to name all 16 categories in order to get an overview over the possible stimuli categories and to make sure that all categories were clear from the beginning. They were instructed to click on the category they believed was presented. Responses through clicking on a response screen could be changed within the 1500 ms response interval, only the last entered response was counted as the answer. Prior to the real experiment a practice session was performed for the participants to get used to the time course of the experiment and the position of category items on the response screen. This screen was shown for an additional 300 ms in order to provide feedback and indicate whether the entered answer was incorrect. In that case, a short low beep sound occurred and the correct category was highlighted by setting its background to white. The practice session consisted of 320 trials. After 160 trials the participants had the chance to take a short break. In the break, their performance of the first block was shown on the screen along the percentage of trials where no answer was entered. After the practice blocks, observers were shown an example image of the manipulation (not used in the experiment) to minimise surprise. Images used in the practice session were natural images from 16-class-ImageNet (Geirhos et al., 2018), hence there was no overlap with images or manipulations used in the experiments.

### A.3 APPARATUS

Observers were shown the  $224 \times 224$  pixels stimuli in a dark cabin on a 22", 120 Hz VIEWPixx LCD monitor (VPixx Technologies, Saint-Bruno, Canada). The screen of size  $484 \times 302$  mm corresponds to  $1920 \times 1200$  pixels, although stimuli were only presented foveally at the center of the screen ( $3 \times 3$  degrees of visual angle at a viewing distance of 107 cm) while the background was set to a grey value of 0.7614 in the  $[0, 1]$  range, the average greyscale value of all stimuli used in the original experiment. Participants used a chin rest to keep their head position static during an experiment. Stimulus presentation was conducted with the Psychophysics Toolbox (version 3.0.12) in MATLAB (Release 2016a, The MathWorks, Inc., Natick, Massachusetts, United States) using a 12-core desktop computer (AMD HD7970 graphics card "Tahiti" by AMD, Sunnyvale, California, United States) running Kubuntu 14.04 LTS. Participants clicked on a response screen, showing an

Published as a conference paper at ICLR 2019

experiment	instruction	# p.	#♀	#♂	age range	mean age	# stimuli	rt
original	neutral	5	5	0	21–27	24.2	160	772
greyscale	neutral	5	4	1	20–26	23.4	160	811
texture	neutral	5	2	3	23–36	29.0	48	769
silhouette	neutral	10	9	1	21–37	24.1	160	861
edge	neutral	10	6	4	18–30	23.0	160	791
cue conflict	neutral	10	7	3	20–29	23.0	1280	828
cue conflict control	texture	10	5	5	23–32	26.6	1280	942
cue conflict control	shape	10	9	1	18–25	21.8	1280	827
filled silhouette	neutral	32	22	10	18–30	22.3	160	881
overall		97	69	28	18–37	23.5	48,560 trials	857

Table 3: Characteristics of human participants (p.) across experiments. The symbol ‘#’ refers to ‘‘number of’’; ‘rt’ stands for ‘‘median reaction time (ms)’’ in an experiment.

iconic representation of all of the 16 object categories as reported in Geirhos et al. (2018), with a normal computer mouse.

#### A.4 PARTICIPANTS

In total, 97 human observers participated in the study. For a detailed overview about how they were distributed across experiments see Table 3. No observer participated in more than one experiment, and all participants reported normal or corrected-to-normal vision. Observers participating in experiments with a cue conflict manipulation were paid € 10 per hour or gained course credit. Observers measured in all other experiments (with a clear ground truth category) were able to earn an additional bonus up to € 5 or equivalent further course credit based on their performance. This motivation scheme was applied to ensure reliable answer rates, and explained to observers in advance. Participation bonus, in these cases, was calculated as follows: The base level with a bonus of € 0 was set to 50% accuracy. For every additional 5% of accuracy, participants gained a € 0.50 bonus. This means that with a performance above 95%, an observer was able to gain the full bonus of € 5 or equivalent course credit. Overall, we took the following steps to prevent low quality human data: 1. using a controlled lab environment instead of an online crowdsourcing platform; 2. the payment motivation scheme as explained above; 3. displaying observer performance on the screen at regular intervals during the practice session; and 4. splitting longer experiments into five blocks, where participants could take a break in between blocks.

#### A.5 CNN MODELS & TRAINING DETAILS

**ResNet-50** We used a standard ResNet-50 architecture from PyTorch (Paszke et al., 2017), the `torchvision.models.resnet50` implementation. For the comparison against BagNets reported in Table 1, results for IN training correspond to a ResNet-50 pre-trained on ImageNet without any modifications (model weights from `torchvision.models`). Reported results for SIN training correspond to the same architecture trained on SIN for 60 epochs with Stochastic Gradient Descent (`torch.optim.SGD`) using a momentum term of 0.9, weight decay ( $1e-4$ ) and a learning rate of 0.1 which was multiplied by a factor of 0.1 after 20 and 40 epochs of training. We used a batch size of 256. This SIN-trained model is the same model that is reported in Figures 5 and 6 as well as in Table 2. In the latter, this corresponds to the second row (training performed on SIN, no fine-tuning on ImageNet). For the model reported in the third row, training was jointly performed on SIN and on IN. This means that both training data sets were treated as one big data set (exactly twice the size of the IN training data set), on which training was performed for 45 epochs with identical hyperparameters as described above, except that the initial learning rate of 0.1 was multiplied by 0.1 after 15 and 30 epochs. The weights of this model were then used to initialise the model reported in the fourth row of Table 2, which was fine-tuned for 60 epochs on ImageNet (identical hyperparameters except that the initial learning rate of 0.01 was multiplied by 0.1 after 30 epochs). We compared training models from scratch versus starting from an ImageNet-pretrained model. Empirically, using features pre-trained on ImageNet led to better results across experiments, which is

Published as a conference paper at ICLR 2019

why we used ImageNet pre-training throughout experiments and models (for both ResNet-50 and restricted ResNet-50 models).

**BagNets** Model weights (pre-trained on ImageNet) and architectures for BagNets (results reported in Table 1) were kindly provided by Brendel & Bethge (2019). For SIN training, identical settings as for the SIN-trained ResNet-50 were used to ensure comparability (training for 60 epochs with SGD and identical hyperparameters as reported above).

**Faster R-CNN** We used the Faster R-CNN implementation from <https://github.com/jwyang/faster-rcnn.pytorch> (commit 21f28986) with all hyperparameters kept at default. The only changes we made to the model is replacing the encoder with ResNet-50 (respectively ResNet-152 for the results in Table 4) and applying custom input whitening. For Pascal VOC 2007 we trained the model for 7 epochs with a batch size of 1, a learning rate of 0.001 and a learning rate decay step after epoch 5. Images were resized to have a short edge of 600 pixels. For MS COCO we trained the same model on the 2017 train/val split for training and testing respectively. We trained for 6 epochs with a batch size of 16 on 8 GPUs employing a learning rate of 0.02 and a decay step after 4 epochs. Images were resized to have a short edge of 800 pixels.

**Pre-trained AlexNet, GoogLeNet, VGG-16** We used AlexNet (Krizhevsky et al., 2012), GoogLeNet (Szegedy et al., 2015) and VGG-16 (Simonyan & Zisserman, 2015) for the evaluation reported in Figure 4. Evaluation was performed using Caffe (Jia et al., 2014). Network weights (training on ImageNet) were obtained from <https://github.com/BVLC/caffe/wiki/Model-Zoo> (AlexNet & GoogLeNet) and <http://www.robots.ox.ac.uk/> (VGG-16).

**ResNet-101 pre-trained on Open Images V2** For our comparison of biases in ImageNet vs. OpenImages (Figure 13 right) the ResNet-101 pretrained on Open Images V2 (Krasin et al., 2017) was used. It was obtained from <https://github.com/openimages/dataset/blob/master/READMEV2.md> along with the inference code provided by the authors. In order to map predictions to the 16 classes, we used the parameters  $top-k = 100000$  and  $score\_threshold = 0.0$  to obtain as all predictions, and then mapped the responses to our 16 classes using the provided label map. 15 out of our 16 classes are classes in Open Images as well; the remaining class `keyboard` was mapped to Open Images class `computer_keyboard` (in this case, Open Images makes a finer distinction to separate musical keyboards from computer keyboards).

**ResNet-101, ResNet-152, DenseNet-121, SqueezeNet1.1** For the comparison to other models pre-trained on ImageNet (Figure 13 left), we evaluated the pre-trained networks provided by `torchvision.models`.

**Training AlexNet, VGG-16 on SIN** For the evaluation of model biases after training on SIN (Figure 11), we obtained the model architectures from `torchvision.models` and trained the networks under identical circumstances as ResNet-50. This includes identical hyperparameter settings, except for the learning rate. The learning rate for AlexNet was set to 0.001 and for VGG-16 to 0.01 initially; both learning rates were multiplied by 0.1 after 20 and 40 epochs of training (60 epochs in total).

#### A.6 IMAGE MANIPULATIONS AND IMAGE DATABASE

In total, we conducted nine different experiments. Here is an overview of the images and / or image manipulations for all of them. All images were saved in the `png` format and had a size of  $224 \times 224$  pixels. Original, texture and cue conflict images are visualised in Figure 7.

**Original experiment** This experiment consisted of 160 coloured images, 10 per category. All of them had a single, unmanipulated object (belonging to one category) in front of a white background. This white background was especially important since these stimuli were being used as content images for style transfer, and we thus made sure that the background was neutral to produce better style transfer results. The images for this experiment as well as for the texture experiment

Published as a conference paper at ICLR 2019

described below were carefully selected using Google advanced image search with the criteria “labelled for noncommercial reuse with modification (free to use, share and modify)” and the search term “<entity> white background” (original) or “<entity> texture” (texture). In some cases where this did not lead to sufficient results, we used images from the ImageNet validation data set which were manually modified to have a white background if necessary. We made sure that both the images from this experiment as well as the texture images were all correctly recognised by all four pre-trained CNNs (if an image was not correctly recognised, we replaced it by another one). This was used to ensure that our results for cue conflict experiments are fully interpretable: if, e.g., a texture image was not correctly recognised by CNNs, there would be no point in using it as a texture (style) source for style transfer.

**Greyscale experiment** This experiment used the same images as the original experiment with the difference that they were converted to greyscale using `skimage.color.rgb2gray`. For CNNs, greyscale images were stacked three times along the colour channel.

**Silhouette experiment** The images from the original experiment were transformed into silhouette images showing an entirely black object on a white background. We used the following transformation procedure: First, images were converted to `bmp` using command line utility (`convert`). They were then converted to `svg` using `potrace`, and then to `png` using `convert` again. Since an entirely automatic binarization pipeline is not feasible (it takes domain knowledge to understand that a car wheel should, but a doughnut should not be filled with black colour), we then manually checked every single image and adapted the silhouette using `GIMP` if necessary.

**Edge experiment** The stimuli shown in this condition were generated by applying the “Canny” edge extractor implemented in MATLAB (Release 2016a, The MathWorks, Inc., Natick, Massachusetts, United States) to the images used in the original experiment. No further manipulations were performed on this data set. This line of code was used to detect edges and generate the stimuli used in this experiment:

```
imwrite(1-edge(imgaussfilt(rgb2gray(imread(filename)), 2),
'Canny'), targetFilename);
```

**Texture experiment** Images were selected using the procedure outlined above for the original experiment. Some objects have a fairly stationary texture (e.g. animals), which makes it easy to find texture images for them. For the more difficult case (e.g. man-made objects), we made use of the fact that every object can become a texture if it is used not in isolation, but rather in a clutter of many objects of the same kind (e.g. Gatys et al., 2017). That is, for a `bottle` texture we used images with many bottles next to each other (as visualised in Figure 7).

**Cue conflict experiment** This experiment used images with a texture-shape cue conflict. They were generated using iterative style transfer (Gatys et al., 2016) between a texture image (from the texture experiment described above) and a content image (from the original experiment) each. While 48 texture images and 160 content images would allow for a total of  $48 \times 160 = 7680$  cue conflict images (480 per category), we used a balanced subset of 1280 images instead (80 per category), which allows for presentation to human observers within a single experimental session. The procedure for selecting the style and content images was done as follows. For all possible  $16 \times 16$  combinations of style and texture categories, exactly five cue conflict images were generated by randomly sampling style and content images from their respective categories. Sampling was performed without replacement for as long as possible, and then without replacement for the remaining images. The same stimuli acquired with this method were used for the cue conflict control experiments, where participants saw exactly these images but with different instructions biased towards shape and towards texture (results described later). For our analysis of texture vs. shape biases (Figure 4), we excluded trials for which no cue conflict was present (i.e., those trials where a bicycle content image was fused with a bicycle texture image, hence no texture-shape cue conflict present).

**Filled silhouette experiment** Style transfer is not the only possibility to generate a texture-shape cue conflict, and we here aimed at testing one other method to generate such stimuli: cropping texture images with a shape mask, such that the silhouette of an object and its texture constitute a cue conflict (visualised in Figure 7). Stimuli were generated by using the silhouette images from the

Published as a conference paper at ICLR 2019

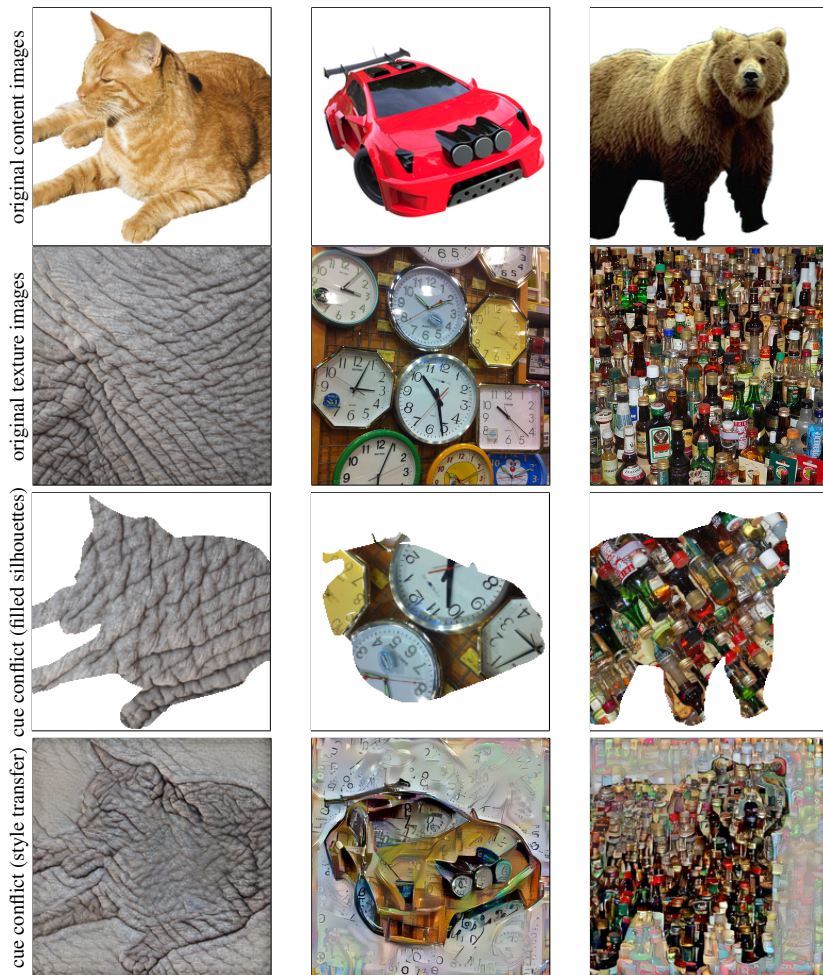


Figure 7: Visualisation of stimuli in data sets. Top two rows: content and texture images. Bottom rows: cue conflict stimuli generated from the texture and content images above (silhouettes filled with rotated textures; style transfer stimuli).



Published as a conference paper at ICLR 2019

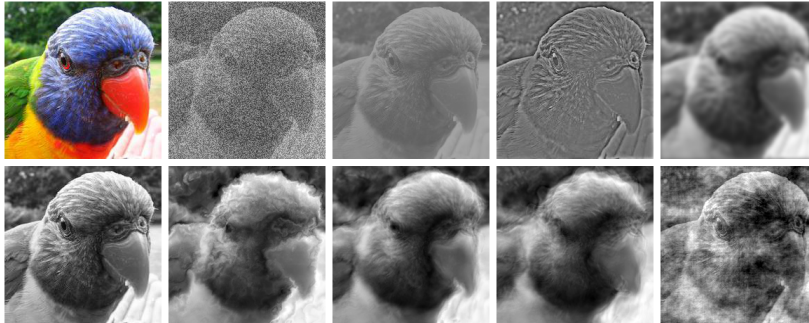


Figure 8: Visualisation of image distortions. One exemplary image (class `bird`, original image in colour at the top left) is manipulated as follows. From left to right: additive uniform noise, low contrast, high-pass filtering, low-pass filtering. In the row below, a greyscale version for comparison; the other manipulations from left to right are: Eidolon manipulations I, II and III as well as phase noise. Figure adapted from Geirhos et al. (2018) with the authors' permission.

silhouette experiment as a mask for texture images. If the silhouette image at a certain location has a black pixel, the texture was used at this location, and for white pixels the resulting target image pixel was white. In order to have a larger variety of textures than the 48 textures used in the texture experiment, the texture database was augmented by rotating all textures with ten different previously chosen angles uniformly distributed between 0 and 360 degrees, resulting in a texture database of 480 images. Results for this control experiment, not part of the main paper, are reported later. We ensured that no silhouette was seen more than once per observer.

**Robustness experiment (distorted images)** For this experiment, human accuracies for reference were provided by Geirhos et al. (2018). Human 'error bars' indicate the full range of results for human observers. CNNs were then evaluated on different image manipulations applied to natural images as outlined in the paper. For maximal comparability, we also used the same images. For a description of the parametric distortion we kindly refer the reader to Geirhos et al. (2018). In Figure 8, we plot one example image across manipulations.

#### A.7 STYLIZED-IMAGENET (SIN)

We used AdaIN style transfer (Huang & Belongie, 2017) to generate Stylized-ImageNet. More specifically, the AdaIN implementation from <https://github.com/naoto0804/pytorch-AdaIN> (commit 31e769c159d4c8639019f7db7e035a7f938a6a46) was employed to stylize the entire ImageNet training and validation data sets. Style transfer was performed once per ImageNet image. As a style source, we used images from Kaggle's `Painter by Numbers` data set (<https://www.kaggle.com/c/painter-by-numbers/>, accessed on March 1, 2018). Style selection was performed randomly with replacement. Every ImageNet image was stylized once and only once. Paintings from the Kaggle data set were used if at least  $224 \times 224$  pixels in size; the largest possible square crop was then downsampled to this size prior to using it as a style image. All accuracies are reported on the respective validation data sets. Code to generate Stylized-ImageNet from ImageNet (and the Kaggle paintings) is available on github in this repository: <https://github.com/rgeirhos/Stylized-ImageNet>

#### A.8 RESULTS: CUE CONFLICT CONTROL EXPERIMENTS (DIFFERENT INSTRUCTIONS)

We investigated the effect of different instructions to human observers. The results presented in the main paper for cue conflict stimuli correspond all to a neutral instruction, not biased w.r.t. texture or shape. In two separate experiments, participants were explicitly instructed to ignore the textures and click on the shape category of cue conflict stimuli, and vice versa. The results, presented in

Published as a conference paper at ICLR 2019

training	fine-tuning	top-1 IN accuracy (%)	top-5 IN accuracy (%)	Pascal VOC mAP50 (%)
IN (vanilla ResNet-152)	-	78.31	94.05	76.9
SIN	-	65.26	86.31	75.0
SIN+IN	-	77.62	93.59	77.3
SIN+IN	IN	<b>78.87</b>	<b>94.41</b>	<b>78.3</b>

Table 4: Accuracy and object detection performance for ResNet-152. Accuracy comparison on the ImageNet (IN) validation data set as well as object detection performance (mAP50) on PASCAL VOC 2007. All models have an identical ResNet-152 architecture.

training	ft	mCE	Noise			Blur			
			Gaussian	Shot	Impulse	Defocus	Glas	Motion	Zoom
IN (vanilla ResNet-50)	-	76.7	79.8	81.6	82.6	74.7	88.6	78.0	79.9
SIN	-	77.3	71.2	73.3	72.1	88.8	85.0	79.7	90.9
SIN+IN	-	<b>69.3</b>	<b>66.2</b>	<b>66.8</b>	<b>68.1</b>	<b>69.6</b>	<b>81.9</b>	<b>69.4</b>	80.5
SIN+IN	IN	73.8	75.9	77.0	77.5	71.7	86.0	74.0	<b>79.7</b>

training	ft	Weather				Digital			
		Snow	Frost	Fog	Brightness	Contrast	Elastic	Pixelate	JPEG
IN (vanilla ResNet-50)	-	77.8	74.8	66.1	56.6	71.4	84.8	76.9	76.8
SIN	-	71.8	74.4	66.0	79.0	<b>63.6</b>	81.1	72.9	89.3
SIN+IN	-	<b>68.0</b>	<b>70.6</b>	<b>64.7</b>	57.8	66.4	<b>78.2</b>	<b>61.9</b>	<b>69.7</b>
SIN+IN	IN	74.5	72.3	66.2	<b>55.7</b>	67.6	80.8	75.0	73.2

Table 5: Corruption error (lower=better) on ImageNet-C (Hendrycks & Dietterich, 2019), consisting of different types of noise, blur, weather and digital corruptions. Abbreviations: mCE = mean Corruption Error (average of the 15 individual corruption error values); SIN = Stylized-ImageNet; IN = ImageNet; ft = fine-tuning. Results kindly provided by Dan Hendrycks.

Published as a conference paper at ICLR 2019

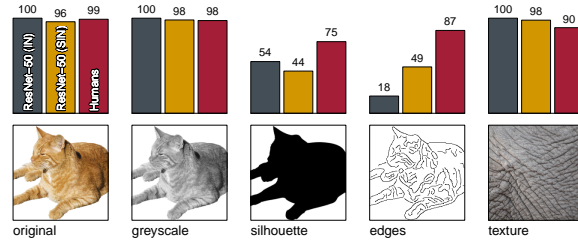


Figure 9: Accuracies and example stimuli for five different experiments without cue conflict, comparing training on ImageNet (IN) to training on Stylized-ImageNet (SIN).

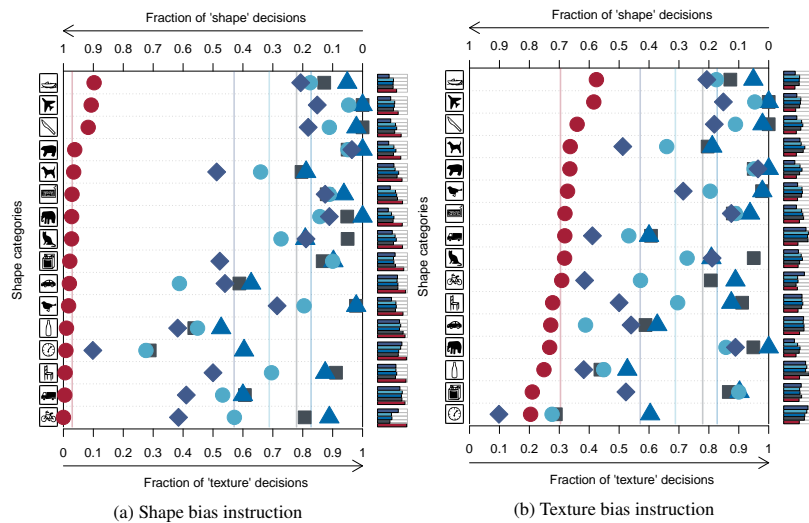


Figure 10: Classification results for human observers (red circles) and ImageNet-trained networks AlexNet (purple diamonds), VGG-16 (blue triangles), GoogLeNet (turquoise circles) and ResNet-50 (grey squares) on stimuli with a texture-shape cue conflict generated with style transfer, and *biased* rather than neutral instructions to human observers. Plotting conventions and CNN data as in Figure 4.

Figure 10, indicate that for a shape bias instruction, human data are almost exactly the same as for the neutral instruction reported earlier (indicating that human observers are indeed using shapes per default); and if they are instructed to ignore the shapes and click on the texture category, they *still* show a substantial shape bias (indicating that even if they seek to ignore shapes, they find it extremely difficult to do so).

#### A.9 RESULTS: FILLED SILHOUETTE EXPERIMENT

This experiment was conducted as a control experiment to make sure that the strong differences between humans and CNNs when presented with cue conflict images are not merely an artefact of the particular setup that we employed. Stimuli are visualised in Figure 7; results in Figure 12. In a nutshell, we also find a shape bias in humans when stimuli are not generated via style transfer but instead through cropping texture images with a shape mask, such that the silhouette of an object and its texture constitute a cue conflict. CNNs have a less pronounced texture bias in these experiments;

Published as a conference paper at ICLR 2019

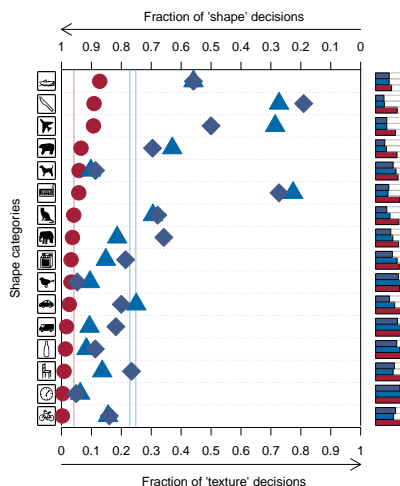


Figure 11: Texture vs shape biases on of AlexNet and VGG-16 after training on Stylized-ImageNet. Plotting conventions as in Figures 4 and 5. Plot shows biases for AlexNet (purple diamonds), VGG-16 (blue triangles) and human observers (red circles) for comparison. For GoogLeNet, no data is available since network training was performed in PyTorch and `torchvision.models` unfortunately does not provide a GoogLeNet (inception\_v1) architecture.

ResNet-50 trained on SIN still responds with the shape category more than ResNet-50 trained on IN. Overall, these results are much more difficult to interpret since the texture-silhouette cue conflict stimuli, visualised in Figure 7, do not have a clear-cut texture-shape distinction like the cue conflict stimuli generated via style transfer. Still, they are largely in accord with the style transfer results presented in the main paper.

#### A.10 IMAGE RIGHTS & ATTRIBUTION

The images presented in Figure 7 were collected from different origins. We here indicate their URL, creator and license terms (if applicable). Some of the images presented in Figure 7 also appear in Figures 1, 2 and 9; the terms below apply accordingly. Top row, cat image: <https://pixabay.com/p-964343/>, released under the CC0 creative commons license as indicated on the website. The CC0 creative commons license is accessible from <https://creativecommons.org/publicdomain/zero/1.0/legalcode>. Car image: <https://pixabay.com/p-1930237/>, released under the CC0 creative commons license as indicated on the website. Bear image: ImageNet image n02132136-871.JPEG, manually modified to have a white background. Second row, elephant texture: cropped from <https://www.flickr.com/photos/flowcomm/5089601226>, released under the CC BY 2.0 license by user flowcomm as indicated on the website. The license is accessible from <https://creativecommons.org/licenses/by/2.0/legalcode>. Clock texture: cropped from [https://commons.wikimedia.org/wiki/File:HK\\_Sheung\\_Wan\\_%E4%B8%AD%E6%BA%90%E4%B8%AD%E5%BF%83\\_Midland\\_Plaza\\_shop\\_Japan\\_Home\\_City\\_clocks\\_displayed\\_for\\_sale\\_April-2011.jpg](https://commons.wikimedia.org/wiki/File:HK_Sheung_Wan_%E4%B8%AD%E6%BA%90%E4%B8%AD%E5%BF%83_Midland_Plaza_shop_Japan_Home_City_clocks_displayed_for_sale_April-2011.jpg), released under the Creative Commons Attribution-Share Alike 3.0 Unported, 2.5 Generic, 2.0 Generic and 1.0 Generic licenses by user Ho Mei Danniell as indicated on the website. The CC Attribution-Share Alike 3.0 license is accessible from <https://creativecommons.org/licenses/by-sa/3.0/legalcode>. Bottle texture: cropped from [https://commons.wikimedia.org/wiki/File:Liquor\\_bottles.jpg](https://commons.wikimedia.org/wiki/File:Liquor_bottles.jpg), released under the CC BY 2.0 license by user scottfeldstein as indicated on the website. The CC BY 2.0 license is accessible from <https://creativecommons.org/licenses/by/2.0/legalcode>.

Published as a conference paper at ICLR 2019

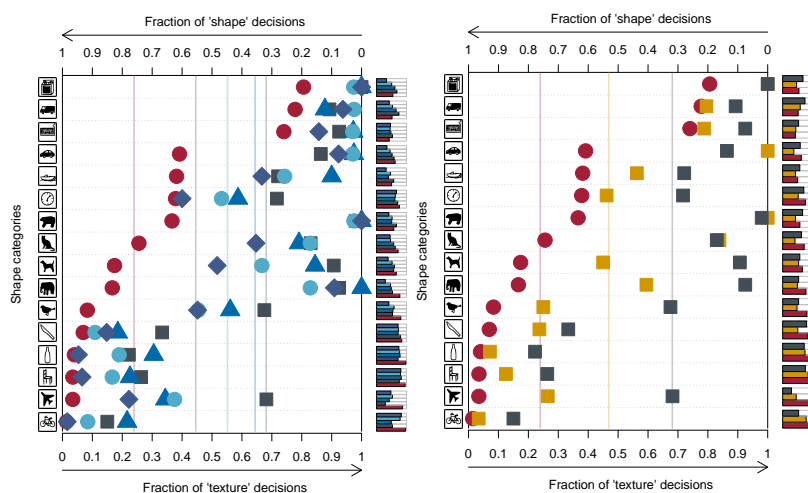


Figure 12: Classification results for human observers and CNNs on stimuli with a texture-silhouette cue conflict (filled silhouette experiment). Plotting conventions as in Figures 4 and 5.  
**Left:** Human observers (red circles) and ImageNet-trained networks AlexNet (purple diamonds), VGG-16 (blue triangles), GoogLeNet (turquoise circles) and ResNet-50 (grey squares).  
**Right:** Human observers (red circles, data identical to the left) and ResNet-50 trained on ImageNet (grey squares) vs. ResNet-50 trained on Stylized-ImageNet (orange squares).

Published as a conference paper at ICLR 2019

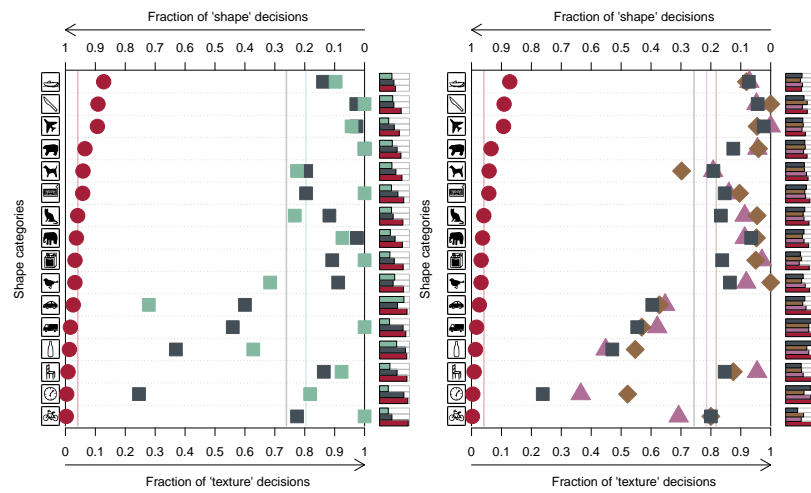


Figure 13: The texture bias on cue conflict stimuli is not specific to ImageNet-trained networks (left) and also occurs in very deep, wide and compressed networks (right).

**Left:** The texture bias is not specific to ImageNet-trained networks. Comparison of texture-shape biases on cue conflict stimuli generated with style transfer for ResNet-101 trained on ImageNet (grey squares) and ResNet-101 trained on the Open Images Dataset V2 (green squares) along with human data for comparison (red circles). Both networks have a qualitatively similar texture bias. We use a ResNet-101 architecture here since Open Images has released a pre-trained ResNet-101.

**Right:** The texture bias also appears in a very deep network (ResNet-152, grey squares), a very wide one (DenseNet-121, purple triangles), and a very compact one (SqueezeNet1.1, brown diamonds). Human data for comparison (red circles). All networks are pre-trained on ImageNet.

## 2.2 Benchmarking Robustness in Object Detection: Autonomous Driving when Winter is Coming

*Additional authorship information* The star symbol (\*) on the next page indicates equal contribution (C.M., B.M., R.G. and E.R.); the dagger symbol (†) indicates joint senior authors (O.B., A.S.E., M.B. and W.B.). This was explicitly stated in [arXiv version v1](#); the explanation was shortened to the symbols for the camera-ready version of the paper.

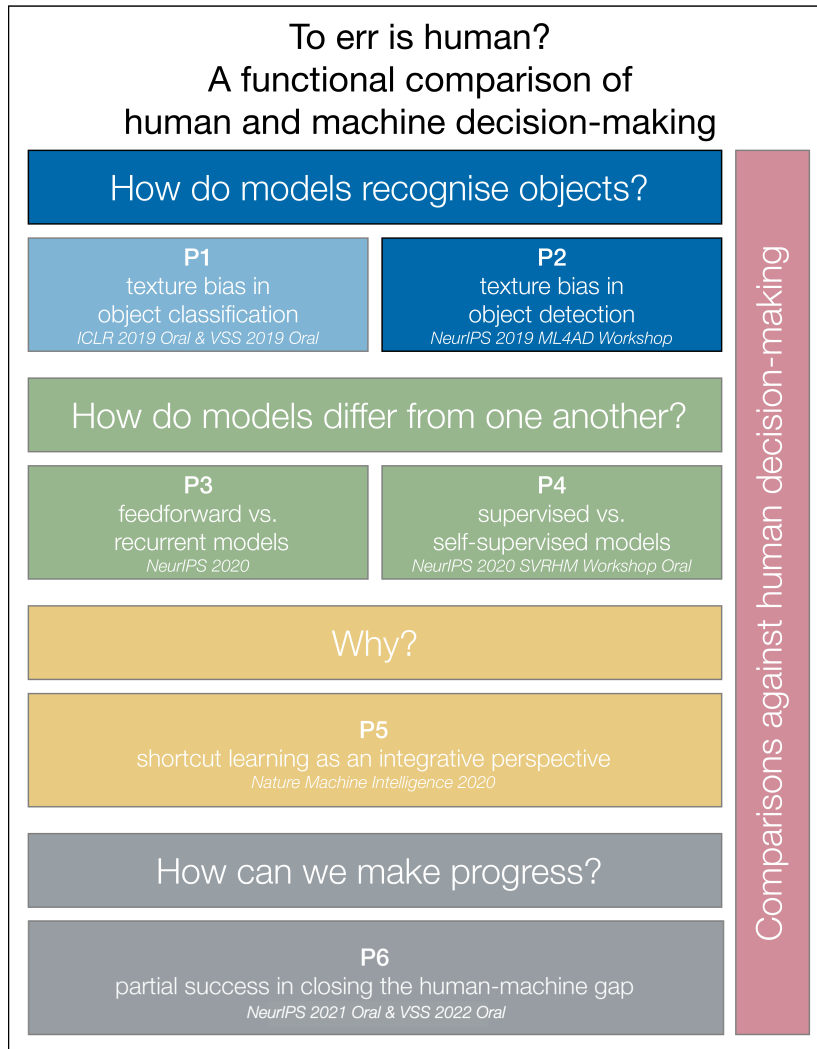


Figure 2.2: Schematic overview of the projects presented in this thesis. Projects P1–P4 ask “How do machines decide”, project P5 presents the concept of shortcut learning as an integrative perspective tackling the question “Why do machines decide the way they do”. Finally, project P6 presents a comprehensive benchmark to measure future progress, and reports first signs of (partial) success. Throughout the thesis, machine decision making will be compared against human decision making.

---

## Benchmarking Robustness in Object Detection: Autonomous Driving when Winter is Coming

---

Claudio Michaelis\* Benjamin Mitzkus\* Robert Geirhos\* Evgenia Rusak\*

Oliver Bringmann†

Alexander S. Ecker†

Matthias Bethge†

Wieland Brendel†

University of Tübingen

first.last@uni-tuebingen.de

### Abstract

The ability to detect objects regardless of image distortions or weather conditions is crucial for real-world applications of deep learning like autonomous driving. We here provide an easy-to-use benchmark to assess how object detection models perform when image quality degrades. The three resulting benchmark datasets, termed Pascal-C, Coco-C and Cityscapes-C, contain a large variety of image corruptions. We show that a range of standard object detection models suffer a severe performance loss on corrupted images (down to 30–60% of the original performance). However, a simple data augmentation trick—stylizing the training images—leads to a substantial increase in robustness across corruption type, severity and dataset. We envision our comprehensive benchmark to track future progress towards building robust object detection models. Benchmark, code and data will be made publicly available.



Figure 1: Mistaking a dragon for a bird (left) may be dangerous but missing it altogether because of snow (right) means playing with fire. Sadly, this is exactly the fate that an autonomous agent relying on a state-of-the-art object detection system would suffer. Predictions generated using Faster R-CNN; best viewed on screen.

### 1 Introduction

*A day in the near future: Autonomous vehicles are swarming the streets all over the world, tirelessly collecting data. But on this cold November afternoon traffic comes to an abrupt halt as it suddenly begins to snow: winter is coming. Huge snowflakes are falling from the sky and the cameras of autonomous vehicles are no longer able to make sense of their surroundings, triggering immediate emergency brakes. A day*

Machine Learning for Autonomous Driving Workshop at the 33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.





Figure 2: Expect the unexpected: To ensure safety, an autonomous vehicle must be able to recognize objects even in challenging outdoor conditions such as fog, rain, snow and at night.<sup>1</sup>

*later, an investigation of this traffic disaster reveals that the unexpectedly large size of the snowflakes was the cause of the chaos: While state-of-the-art vision systems had been trained on a variety of common weather types, their training data contained hardly any snowflakes of this size...*

This fictional example highlights the problems that arise when Convolutional Neural Networks (CNNs) encounter settings that were not explicitly part of their training regime. For example, state-of-the-art object detection algorithms such as Faster R-CNN [Ren et al., 2015] fail to recognize objects when snow is added to an image (as shown in Figure 1), even though the objects are still clearly visible to a human eye. At the same time, augmenting the training data with several types of distortions is not a sufficient solution to achieve general robustness against previously unknown corruptions: It has recently been demonstrated that CNNs generalize poorly to novel distortion types, despite being trained on a variety of other distortions [Geirhos et al., 2018].

On a more general level, CNNs often fail to generalize outside of the training domain or training data distribution. Examples include the failure to generalize to images with uncommon poses of objects [Alcorn et al., 2019] or to cope with small distributional changes [e.g. Zech et al., 2018, Touvron et al., 2019]. One of the most extreme cases are adversarial examples [Szegedy et al., 2013]: images with a domain shift so small that it is imperceptible for humans yet sufficient to fool a DNN. We here focus on the less extreme but far more common problem of perceptible image distortions like blurry images, noise or natural distortions like snow.

As an example, autonomous vehicles need to be able to cope with wildly varying outdoor conditions such as fog, frost, snow, sand storms, or falling leaves, just to name a few (as visualized in Figure 2). One of the major reasons why autonomous cars have not yet gone mainstream is the inability of their recognition models to function well in adverse weather conditions [Dai and Van Gool, 2018]. Getting data for unusual weather conditions is hard and while many common environmental conditions can (and have been) modelled, including fog [Sakaridis et al., 2018a], rain [Hospach et al., 2016], snow [Bernuth et al., 2019] and daytime to nighttime transitions [Dai and Van Gool, 2018], it is impossible to foresee all potential conditions that might occur “in the wild”.

If we could build models that are robust to every possible image corruption, it is to be expected that weather changes would not be an issue. However, in order to assess the robustness of models one first needs to define a measure. While testing models on the set of all possible corruption types is impossible. We therefore propose to evaluate models on a diverse range of corruption types that were not part of the training data and demonstrate that this is a useful approximation for predicting performance under natural distortions like rain, snow, fog or the transition between day and night.

More specifically we propose three easy-to-use benchmark datasets termed PASCAL-C, COCO-C and Cityscapes-C to assess distortion robustness in object detection. Each dataset contains versions of the original object detection dataset which are corrupted with 15 distortions, each spanning five levels of severity. This approach follows Hendrycks and Dietterich [2019], who introduced corrupted versions of commonly used *classification* datasets (ImageNet-C, CIFAR10-C) as standardized benchmarks. After evaluating standard object detection algorithms on these benchmark datasets, we show how a simple data augmentation technique—stylizing the training images—can strongly improve robustness across corruption type, severity and dataset.

<sup>1</sup>Outdoor hazards have been directly linked to increased mortality rates [Lystad and Brown, 2018].

### 1.1 Contributions

Our contributions can be summarized as follows:

1. We demonstrate that a broad range of object detection and instance segmentation models suffer severe performance impairments on corrupted images.
2. To quantify this behaviour and to enable tracking future progress, we propose the Robust Detection Benchmark, consisting of three benchmark datasets termed PASCAL-C, COCO-C & Cityscapes-C.
3. We demonstrate that improved performance on this benchmark of synthetic corruptions corresponds to increased robustness towards real-world “natural” distortions like rain, snow and fog.
4. We use the benchmark to show that corruption robustness scales with performance on clean data and that a simple data augmentation technique—stylizing the training data—leads to large robustness improvements for all evaluated corruptions without any additional labelling costs or architectural changes.
5. We make our benchmark, corruption and stylization code openly available in an easy-to-use fashion:
  - Benchmark, <sup>2</sup> data and data analysis are available at <https://github.com/bethgelab/robust-detection-benchmark>.
  - Our pip installable image corruption library is available at <https://github.com/bethgelab/imagecorruptions>.
  - Code to stylize arbitrary datasets is provided at <https://github.com/bethgelab/stylize-datasets>.

### 1.2 Related Work

**Benchmarking corruption robustness** Several studies investigate the vulnerability of CNNs to common corruptions. Dodge and Karam [2016] measure the performance of four state-of-the-art image recognition models on out-of-distribution data and show that CNNs are in particular vulnerable to blur and Gaussian noise. Geirhos et al. [2018] show that CNN performance drops much faster than human performance for the task of recognizing corrupted images when the perturbation level increases across a broad range of corruption types. Azulay and Weiss [2018] investigate the lack of invariance of several state-of-the-art CNNs to small translations. A benchmark to evaluate the robustness of recognition models against common corruptions was recently introduced by Hendrycks and Dietterich [2019].

**Improving corruption robustness** One way to restore the performance drop on corrupted data is to preprocess the data in order to remove the corruption. Mukherjee et al. [2018] propose a DNN-based approach to restore image quality of rainy and foggy images. Bahnsen and Moeslund [2018] and Bahnsen et al. [2019] propose algorithms to remove rain from images as a preprocessing step and report a subsequent increase in recognition rate. A challenge for these approaches is that noise removal is currently specific to a certain distortion type and thus does not generalize to other types of distortions. Another line of work seeks to enhance the classifier performance by the means of data augmentation, i.e. by directly including corrupted data into the training. Vasiljevic et al. [2016] study the vulnerability of a classifier to blurred images and enhance the performance on blurred images by fine-tuning on them. Geirhos et al. [2018] examine the generalization between different corruption types and find that fine-tuning on one corruption type does not enhance performance on other corruption types. In a different study, Geirhos et al. [2019] train a recognition model on a stylized version of the ImageNet dataset [Russakovsky et al., 2015], reporting increased general robustness against different corruptions as a result of a stronger bias towards ignoring textures and focusing on object shape. Hendrycks and Dietterich [2019] report several methods leading to enhanced performance on their corruption benchmark: Histogram Equalization, Multiscale Networks, Adversarial Logit Pairing, Feature Aggregating and Larger Networks.

<sup>2</sup>Our evaluation code to assess performance under corruption has been integrated into one of the most widely used detection toolboxes. The code can be found here: <https://github.com/bethgelab/mmdetection>

**Evaluating robustness to environmental changes in autonomous driving** In recent years, weather conditions turned out to be a central limitation for state-of-the-art autonomous driving systems [Sakaridis et al., 2018a, Volk et al., 2019, Dai and Van Gool, 2018, Chen et al., 2018, Lee et al., 2018]. While many specific approaches like modelling weather conditions [Sakaridis et al., 2018a,b, Volk et al., 2019, Bernuth et al., 2019, Hospach et al., 2016, Bernuth et al., 2018] or collecting real [Wen et al., 2015, Yu et al., 2018, Che et al., 2019, Caesar et al., 2019] and artificial [Gaidon et al., 2016, Ros et al., 2016, Richter et al., 2017, Johnson-Roberson et al., 2017] datasets with varying weather conditions, no general solution towards the problem has yet emerged. Radecki et al. [2016] experimentally test the performance of various sensors and object recognition and classification models in adverse weather and lighting conditions. Bernuth et al. [2018] report a drop in the performance of a Recurrent Rolling Convolution network trained on the KITTI dataset when the camera images are modified by simulated raindrops on the windshield. Pei et al. [2017] introduce VeriVis, a framework to evaluate the security and robustness of different object recognition models using real-world image corruptions such as brightness, contrast, rotations, smoothing, blurring and others. Machiraju and Channappayya [2018] propose a metric to evaluate the degradation of object detection performance of an autonomous vehicle in several adverse weather conditions evaluated on the Virtual KITTI dataset. Building upon Hospach et al. [2016], Volk et al. [2019] study the fragility of an object detection model against rainy images, identify corner cases where the model fails and include images with synthetic rain variations into the training set. They report enhanced performance on real rain images. Bernuth et al. [2019] model photo-realistic snow and fog conditions to augment real and virtual video streams. They report a significant performance drop of an object detection model when evaluated on corrupted data.

## 2 Methods

### 2.1 Robust Detection Benchmark

We introduce the **Robust Detection Benchmark** inspired by the ImageNet-C benchmark for object classification [Hendrycks and Dietterich, 2019] to assess object detection robustness on corrupted images.

**Corruption types** Following Hendrycks and Dietterich [2019], we provide 15 corruptions on five severity levels each (visualized in Figure 3) to assess the effect of a broad range of different corruption types on object detection models.<sup>3</sup> The corruptions are sorted into four groups: noise, blur, digital and weather groups (as defined by Hendrycks and Dietterich [2019]). It is important to note that the corruption types are *not* meant to be used as a training data augmentation toolbox, but rather to measure a model’s robustness against *previously unseen* corruptions. Thus, training should be done without using any of the provided corruptions. For model validation, four separate corruptions are provided (Speckle Noise, Gaussian Blur, Spatter, Saturate). The 15 corruptions described above should only be used to test the final model performance.

**Benchmark datasets** The **Robust Detection Benchmark** consists of three benchmark datasets: PASCAL-C, COCO-C and Cityscapes-C. Among the vast number of available object detection datasets [Everingham et al., 2010, Geiger et al., 2012, Lin et al., 2014, Cordts et al., 2016, Zhou et al., 2017, Neuhold et al., 2017, Krasin et al., 2017], we chose to use PASCAL VOC [Everingham et al., 2010], MS COCO [Lin et al., 2014] and Cityscapes [Cordts et al., 2016] as they are the most commonly used datasets for general object detection (PASCAL & COCO) and street scenes (Cityscapes). We follow common conventions to select the tests splits: VOC2007 test set for PASCAL-C, the COCO 2017 validation set for COCO-C and the Cityscapes validation set for Cityscapes-C.

**Metrics** Since performance measures differ between the original datasets, the dataset-specific performance (P) measures are adopted as defined below:

$$P := \begin{cases} AP^{50}(\%) & \text{PASCAL VOC} \\ AP(\%) & \text{MS COCO \& Cityscapes} \end{cases}$$

<sup>3</sup>These corruption types were introduced by Hendrycks and Dietterich [2019] and modified by us to work with images of arbitrary dimensions. Our generalized corruptions can be found at <https://github.com/bethgelab/imagecorruptions> and installed via `pip3 install imagecorruptions`.

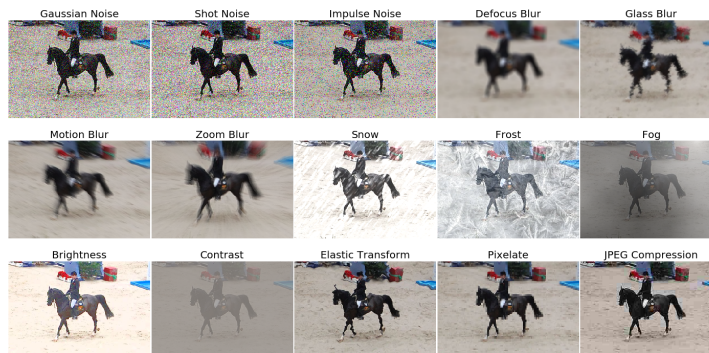


Figure 3: 15 corruption types from Hendrycks and Dieterich [2019], adapted to corrupt arbitrary images (example: randomly selected PASCAL VOC image, center crop, severity 3). Best viewed on screen.

where  $AP^{50}$  stands for the PASCAL ‘Average Precision’ metric at 50% Intersection over Union (IoU) and AP stands for the COCO ‘Average Precision’ metric which averages over IoUs between 50% and 95%. On the corrupted data, the benchmark performance is measured in terms of mean performance under corruption (mPC):

$$mPC = \frac{1}{N_c} \sum_{c=1}^{N_c} \frac{1}{N_s} \sum_{s=1}^{N_s} P_{c,s} \quad (1)$$

Here,  $P_{c,s}$  is the dataset-specific performance measure evaluated on test data corrupted with corruption  $c$  under severity level  $s$  while  $N_c = 15$  and  $N_s = 5$  indicate the number of corruptions and severity levels, respectively. In order to measure relative performance degradation under corruption, the relative performance under corruption (rPC) is introduced as defined below:

$$rPC = \frac{mPC}{P_{clean}} \quad (2)$$

rPC measures the relative degradation of performance on corrupted data compared to clean data.

**Submissions** Submissions to the benchmark should be handed in as a simple pull request to the **Robust Detection Benchmark**<sup>4</sup> and need to include all three performance measures: clean performance ( $P_{clean}$ ), mean performance under corruption (mPC) and relative performance under corruption (rPC). While mPC is the metric used to rank models on the **Robust Detection Benchmark**, the other measures provide additional insights, as they disentangle gains from higher clean performance (as measured by  $P_{clean}$ ) and gains from better generalization performance to corrupted data (as measured by rPC).

**Baseline models** We provide baseline results for a set of common object detection models including Faster R-CNN [Ren et al., 2015], Mask R-CNN [He et al., 2017], Cascade R-CNN [Cai and Vasconcelos, 2018], Cascade Mask R-CNN [Chen et al., 2019a], RetinaNet [Lin et al., 2017a] and Hybrid Task Cascade [Chen et al., 2019a]. We use a ResNet50 [He et al., 2016] with Feature Pyramid Networks [Lin et al., 2017b] as backbone for all models except for Faster R-CNN where we additionally test ResNet101 [He et al., 2016], ResNeXt101-32x4d [Xie et al., 2017] and ResNeXt-64x4d [Xie et al., 2017] backbones. We additionally provide results for Faster R-CNN and Mask R-CNN models with deformable convolutions [Dai et al., 2017, Zhu et al., 2018] in Appendix D. Models were evaluated using the `mmDetection toolbox` [Chen et al., 2019b]; all models were trained and tested with standard hyperparameters. The details can be found in Appendix A.

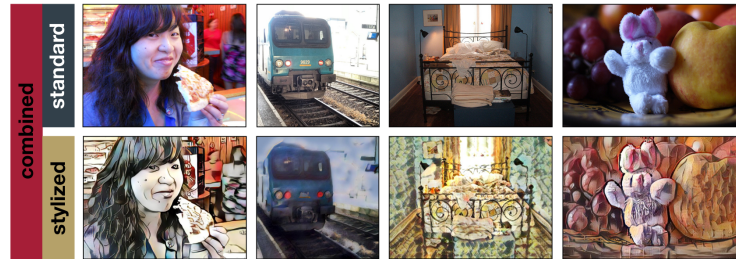


Figure 4: Training data visualization for COCO and Stylized-COCO. The three different training settings are: standard data (top row), stylized data (bottom row) and the concatenation of both (termed ‘combined’ in plots).

## 2.2 Style transfer as data augmentation

For image classification, style transfer [Gatys et al., 2016]—the method of combining the content of an image with the style of another image—has been shown to strongly improve corruption robustness [Geirhos et al., 2019]. We here transfer this method to object detection datasets testing two settings: (1) Replacing each training image with a stylized version and (2) adding a stylized version of each image to the existing dataset. We apply the fast style transfer method AdaIN [Huang and Belongie, 2017] with hyperparameter  $\alpha = 1$  to the training data, replacing the original texture with the randomly chosen texture information of Kaggle’s Painter by Numbers<sup>5</sup> dataset. Examples for the stylization of COCO images are given in Figure 4. We provide ready-to-use code for the stylization of arbitrary datasets at <https://github.com/bethgelab/stylize-datasets>.

## 2.3 Natural Distortions

**Foggy Cityscapes** Foggy Cityscapes Sakaridis et al. [2018a] is a version of Cityscapes with synthetic fog in three severity levels (given by the attenuation coefficient  $\beta = 0.005m^{-1}$ ,  $0.01m^{-1}$  and  $0.02m^{-1}$ ), that was carefully designed to look as realistic as possible. We use Foggy Cityscapes only at test time, testing the same models as used for our experiments with the original Cityscapes dataset and report results in the same AP metric.

**BDD100k** BDD100k Yu et al. [2018] is a driving dataset consisting of 100 thousand videos of driving scenes recorded in varying conditions including weather changes and different times of the day<sup>6</sup>. We use these annotations to perform experiments, on different weather conditions (“clear”, “rainy” and “snowy”) and on the transition from day to night. Training is performed on what we would consider “clean” data - clear for weather and daytime for time - and evaluation is performed on all three splits. We use Faster R-CNN with the same hyper-parameters as in our experiments on COCO. Details of the dataset preparation can be found in Appendix C.

# 3 Results

## 3.1 Image corruptions reduce model performance

In order to assess the effect of image corruptions, we evaluated a set of common object detection models on the three benchmark datasets defined in Section 2. Performance is heavily degraded on corrupted images (compare Table 1). While Faster R-CNN can retain roughly 60% relative performance (rPC) on the rather simple images in PASCAL VOC, the same model suffers a dramatic reduction to 33% rPC on the Cityscapes dataset, which contains many small objects. With some

<sup>4</sup><https://github.com/bethgelab/robust-detection-benchmark>

<sup>5</sup><https://www.kaggle.com/c/painter-by-numbers/>

<sup>6</sup>The frame at the 10th second of each video is annotated with additional information including bounding boxes which we use for our experiments

PASCAL VOC				
model	backbone	clean P [AP <sup>50</sup> ]	corrupted mPC [AP <sup>50</sup> ]	relative rPC [%]
Faster	r50	80.5	48.6	60.4

MS COCO				
model	backbone	clean P [AP]	corrupted mPC [AP]	relative rPC [%]
Faster	r50	36.3	18.2	50.2
Faster	r101	38.5	20.9	54.2
Faster	x101-32x4d	40.1	22.3	55.5
Faster	x101-64x4d	41.3	23.4	56.6
Mask	r50	37.3	18.7	50.1
Cascade	r50	40.4	20.1	49.7
Cascade Mask	r50	41.2	20.7	50.2
RetinaNet	r50	35.6	17.8	50.1
HTC	x101-64x4d	50.6	32.7	64.7

Cityscapes				
model	backbone	clean P [AP]	corrupted mPC [AP]	relative rPC [%]
Faster	r50	36.4	12.2	33.4
Mask	r50	37.5	11.7	31.1

Table 1: Object detection performance of various models. Backbones indicated with  $r$  are ResNet and  $x$  ResNeXt. All model names except for RetinaNet and HTC indicate the corresponding model from the R-CNN family. All COCO models were downloaded from the `mm detection` modelzoo. For all reported quantities: higher is better; square brackets denote metric.

variations, this effect is present in all tested models and also holds for instance segmentation tasks (for instance segmentation results, please see Appendix D).

### 3.2 Robustness increases with backbone capacity

We test variants of Faster R-CNN with different backbones (top of Table 1) and different head architectures (bottom of Table 1) on COCO. For the models with different backbones, we find that all image corruptions—except for the blur types—induce a fixed penalty to model performance, independent of the baseline performance on clean data:  $\Delta \text{mPC} \approx \Delta \text{P}$  (compare Table 1 and Appendix Figure 10). Therefore, models with more powerful backbones show a relative performance improvement under corruption.<sup>7</sup> In comparison, Mask R-CNN, Cascade R-CNN and Cascade Mask R-CNN which draw their performance increase from more sophisticated head architectures all have roughly the same rPC of  $\approx 50\%$ . The current state-of-the-art model Hybrid Task Cascade [Chen et al., 2019a] is in so far an exception as it employs a combination of a stronger backbone, improved head architecture and additional training data to not only outperform the strongest baseline model by 9% AP on clean data but distances itself on corrupted data by a similar margin, achieving a leading relative performance under corruption (rPC) of 64.7%. These results indicate that robustness in the tested regime can be improved primarily through a better image encoding, and better head architectures cannot extract more information if the primary encoding is already sufficiently impaired.

### 3.3 Training on stylized data improves robustness

In order to reduce the strong effect of corruptions on model performance observed above, we tested whether a simple approach (stylizing the training data) leads to a robustness improvement. We evaluate the exact same model (Faster R-CNN) with three different training data schemes (visualized in Figure 4):

- standard:** the unmodified training data of the respective dataset
- stylized:** the training data is stylized completely
- combined:** concatenation of standard and stylized training data

<sup>7</sup>This finding is further supported by investigating models with deformable convolutions (see Appendix D).

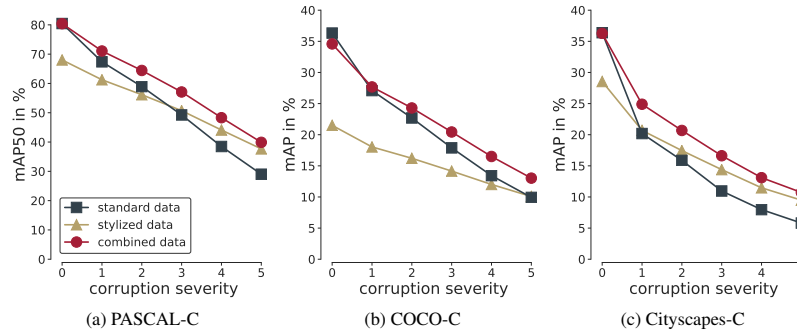


Figure 5: Training on stylized data improves test performance of Faster R-CNN on corrupted versions of PASCAL VOC, MS COCO and Cityscapes which include all 15 types of corruptions shown in Figure 3. Corruption severity 0 denotes clean data. Corruption specific performances are shown in the appendix (Figures 7, 8, 9).

train data	PASCAL VOC [AP <sup>50</sup> ]			MS COCO [AP]			Cityscapes [AP]		
	clean P	corr. mPC	rel. rPC [%]	clean P	corr. mPC	rel. rPC [%]	clean P	corr. mPC	rel. rPC [%]
standard	<b>80.5</b>	48.6	60.4	<b>36.3</b>	18.2	50.2	<b>36.4</b>	12.2	33.4
stylized	68.0	50.0	<b>73.5</b>	21.5	14.1	<b>65.6</b>	28.5	14.7	<b>51.5</b>
combined	80.4	<b>56.2</b>	69.9	34.6	<b>20.4</b>	58.9	36.3	<b>17.2</b>	47.4

Table 2: Object detection performance of Faster R-CNN trained on standard images, stylized images and the combination of both evaluated on standard test sets (test 2007 for PASCAL VOC; val 2017 for MS COCO, val for Cityscapes); higher is better.

The results across our three datasets PASCAL-C, COCO-C and Cityscapes-C are visualized in Figure 5. We observe a similar pattern as reported by Geirhos et al. [2019] for object classification on ImageNet—a model trained on stylized data suffers less from corruptions than the model trained only on the original clean data. However, its performance on clean data is much lower. Combining stylized and clean data seems to achieve the best of both worlds: high performance on clean data as well as strongly improved performance under corruption. From the results in Table 2, it can be seen that both stylized and combined training improve the relative performance under corruption (rPC). Combined training yields the highest absolute performance under corruption (mPC) for all three datasets. This pattern is fairly consistent. Detailed results across corruption types are reported in the Appendix (Figure 7, Figure 8 and Figure 9).

### 3.4 Training directly on stylized data is better than using stylized data only during pre-training

For comparison reasons, we reimplemented the object detection models from Geirhos et al. [2019] and tested them for corruption robustness. Those models use backbones which are pre-trained with Stylized-ImageNet, but the object detection models are trained on the standard clean training sets of Pascal VOC and COCO. In contrast, we here use backbones trained on standard “clean” ImageNet and train using stylized Pascal VOC and COCO. We find that stylized pre-training helps not only on clean data (as reported by Geirhos et al. [2019]) but also for corruption robustness (Table 3), albeit less than our approach of performing the final training on stylized data (compare to Table 2)<sup>8</sup>.

<sup>8</sup>Note that Geirhos et al. [2019] use Faster R-CNN without Feature Pyramids (FPN), which is why the baseline performance of these models is different from ours

train data	PASCAL VOC [AP <sup>50</sup> ]			MS COCO [AP]		
	clean P	corr. mPC	rel. rPC [%]	clean P	corr. mPC	rel. rPC [%]
IN	78.9	45.7	57.4	31.8	15.5	48.7
SIN	75.1	48.2	63.6	29.8	15.3	51.3
SIN+IN	78.0	<b>50.6</b>	<b>64.2</b>	31.1	16.0	<b>51.4</b>
SIN+IN ft IN	<b>79.0</b>	48.9	61.4	<b>32.3</b>	<b>16.2</b>	50.1

Table 3: Object detection performance of Faster R-CNN pre-trained on ImageNet (IN), Stylized ImageNet (SIN) and the combination of both evaluated on standard test sets (test 2007 for PASCAL VOC; val 2017 for MS COCO); higher is better.

train data	BDD100k [AP]					Weather			Day/Night		
	clear P	rainy mPC	rel. rPC [%]	snowy mPC	rel. rPC [%]	day P	night mPC	rel. rPC [%]	day P	night mPC	rel. rPC [%]
clean	<b>27.8</b>	27.6	99.3	23.6	84.9	<b>30.0</b>	21.5	71.7			
stylized	20.9	21.0	100.5	18.7	<b>89.5</b>	24.0	16.8	70.0			
combined	27.7	<b>28.0</b>	<b>101.1</b>	<b>24.2</b>	87.4	<b>30.0</b>	<b>22.5</b>	<b>75.0</b>			

Table 4: Performance of Faster R-CNN across different weather conditions and time changes when trained on standard images, stylized images and the combination of both evaluated on BDD100k (see Appendix C for dataset details); higher is better.

### 3.5 Robustness to natural distortions is connected to synthetic corruption robustness

A central question is whether results on the robust detection benchmark generalize to real-world natural distortions like rain, snow or fog as illustrated in Figure 2. We test this using BDD100k [Yu et al., 2018], a driving scene dataset with annotations for weather conditions. For our first experiment, we train a model only on images that are taken in “clear” weather. We also train models on a stylized version of the same images as well as the combination of both following the protocol from Section 3.3. We then test these models on images which are annotated to be “clear”, “rainy” or “snowy” (see Appendix C for details). We find that these weather changes have little effect on performance on all three models, but that combined training improves the generalization to “rainy” and “snowy” images (Table 4 Weather). It may be important to note that the weather changes of this dataset are often relatively benign (e.g., images annotated as rainy often show only wet roads instead of rain).

A stronger test is generalization of a model trained on images taken during daytime to images taken at night which exhibit a strong appearance change. We find that a model trained on images taken during the day performs much worse at night but combined training improves nighttime performance (Table 4 Day/Night and Appendix C).

As a third test of real-world distortions, we test our approach on Foggy Cityscapes Sakaridis et al. [2018a] which uses fog in three different strengths (given by the attenuation factor  $\beta = 0.005, 0.01$  or  $0.2m^{-1}$ ) as a highly realistic model of natural fog. Fog drastically reduces the performance of standard models trained on Cityscapes which was collected in clear conditions. The reduction is almost 50% for the strongest corruption, see Table 5. In this strong test for OOD (out-of-distribution) robustness, stylized training increases relative performance substantially from about 50% to over 70% (Table 5).

Taken together, these results suggest that there is a connection between performance on synthetic and natural corruptions. Our approach of combined training with stylized data improves performance in every single case with increasing gains in harder conditions.

### 3.6 Performance degradation does not simply scale with perturbation size

We investigated whether there is a direct relationship between the impact of a corruption on the pixel values of an image and the impact of a corruption on model performance. The left of Figure 6 shows the relative performance of Faster R-CNN on the corruptions in PASCAL-C dependent on the perturbation size of each corruption measured in Root Mean Square Error (RMSE). It can be seen that no simple relationship exists, counterintuitively robustness increases to corruption types with higher perturbation size (there is a weak positive correlation between rPC and RMSE,  $r = 0.45$ ).



Foggy Cityscapes [AP]		$\beta = 0.005$		$\beta = 0.01$		$\beta = 0.02$	
train data	clean P	corr. mPC	rel. rPC [%]	corr. mPC	rel. rPC [%]	corr. mPC	rel. rPC [%]
standard	<b>36.4</b>	30.2	83.0	25.1	69.0	18.7	51.4
stylized	28.5	26.2	<b>91.9</b>	24.7	<b>86.7</b>	22.5	<b>78.9</b>
combined	36.3	<b>32.2</b>	88.7	<b>29.9</b>	82.4	<b>26.2</b>	72.2

Table 5: Object detection performance of Faster R-CNN on Foggy Cityscapes when trained on Cityscapes with standard images, stylized images and the combination of both evaluated on the validation set; higher is better;  $\beta$  is the attenuation coefficient in  $m^{-1}$

This stems from the fact that corruptions like Fog or Brightness alter the image globally (resulting in high RMSE) while leaving local structure unchanged. Corruptions like Impulse Noise alter only a few pixels (resulting in low RMSE) but have a drastic impact on model performance.

To investigate further if classical perceptual image metrics are more predictive, we look at the relationship between the perceived image quality of the original and corrupted images measured in structural similarity (SSIM, higher value means more similar, Figure 6 on the right). There is a weak correlation between rPC and SSIM ( $r = 0.48$ ). This analysis shows that SSIM better captures the effect of the corruptions on model performance.

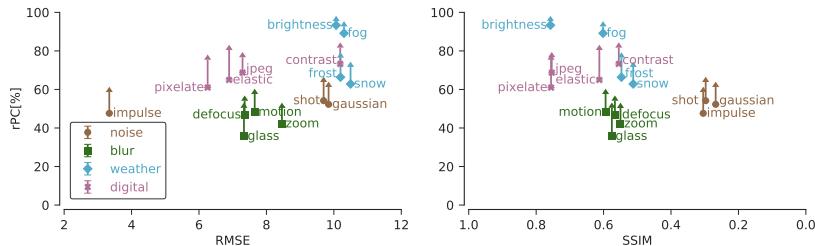


Figure 6: Relative performance under corruption (rPC) as a function of corruption RMSE (left, higher value=greater change in pixel space) and SSIM (right, higher value=higher perceived image quality) evaluated on PASCAL VOC. The dots indicate the rPC of Faster R-CNN trained on standard data; the arrows show the performance gained via training on ‘combined’ data. Corruptions are grouped into four corruption types: noise, blur, weather and digital.

## 4 Discussion

We here showed that object detection and instance segmentation models suffer severe performance impairments on corrupted images. This drop in performance has previously been observed in image recognition models [e.g. Geirhos et al., 2018, Hendrycks and Dietterich, 2019]. In order to track future progress on this important issue, we propose the Robust Detection Benchmark containing three easy-to-use benchmark datasets PASCAL-C, COCO-C and Cityscapes-C. We provide evidence that performance on our benchmarks predicts performance on natural distortions and show that robustness corresponds to model performance on clean data. Apart from providing baselines, we demonstrate how a simple data augmentation technique, namely adding a stylized copy of the training data in order to reduce a model’s focus on textural information, leads to strong robustness improvements. On corrupted images, we consistently observe a performance increase (about 16% for PASCAL, 12% for COCO, and 41% for Cityscapes) with small losses on clean data (0–2%). This approach has the benefit that it can be applied to any image dataset, requires no additional labelling or model tuning and, thus, comes basically for free. At the same time, our benchmark data shows that there is still space for improvement and it is yet to be determined whether the most promising robustness enhancement techniques will require architectural modifications, data augmentation schemes, modifications to the loss function, or a combination of these.

We encourage readers to expand the benchmark with novel corruption types. In order to achieve robust models, testing against a wide variety of different image corruptions is necessary—there is no ‘too much’. Since our benchmark is open source, we welcome new corruption types and look forward to your pull requests to <https://github.com/bethgelab/imagecorruptions>! We envision our comprehensive benchmark to track future progress towards building robust object detection models that can be reliably deployed ‘in the wild’, eventually enabling them to cope with unexpected weather changes, corruptions of all kinds and, if necessary, even the occasional dragonfire.

## References

- Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.
- Robert Geirhos, Carlos RM Temme, Jonas Rauber, Heiko H Schütt, Matthias Bethge, and Felix A Wichmann. Generalisation in humans and deep neural networks. In *NeurIPS*, 2018.
- Reidar P Lystad and Benjamin T Brown. “Death is certain, the time is not”: mortality and survival in Game of Thrones. *Injury epidemiology*, 5(1):44, 2018.
- Michael A Alcorn, Qi Li, Zhitao Gong, Chengfei Wang, Long Mai, Wei-Shinn Ku, and Anh Nguyen. Strike (with) a pose: Neural networks are easily fooled by strange poses of familiar objects. In *CVPR*, 2019.
- John R Zech, Marcus A Badgeley, Manway Liu, Anthony B Costa, Joseph J Titano, and Eric Karl Oermann. Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study. *PLoS medicine*, 15(11):e1002683, 2018.
- Hugo Touvron, Andrea Vedaldi, Matthijs Douze, and Hervé Jégou. Fixing the train-test resolution discrepancy. *arXiv:1906.06423*, 2019.
- Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv:1312.6199*, 2013.
- Dengxin Dai and Luc Van Gool. Dark model adaptation: Semantic image segmentation from daytime to nighttime. In *ITSC*, 2018.
- Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Semantic foggy scene understanding with synthetic data. *IJCV*, 2018a.
- Dennis Hospach, Stefan Müller, Wolfgang Rosenstiel, and Oliver Bringmann. Simulating photo-realistic snow and fog on existing images for enhanced CNN training and evaluation. In *DATE*, 2016.
- Alexander Von Bernuth, Georg Volk, and Oliver Bringmann. Simulating photo-realistic snow and fog on existing images for enhanced CNN training and evaluation. In *ITSC*, 2019.
- Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *ICLR*, 2019.
- Samuel Fuller Dodge and Lina J. Karam. Understanding how image quality affects deep neural networks. *QoMEX*, 2016.
- Aharon Azulay and Yair Weiss. Why do deep convolutional networks generalize so poorly to small image transformations? *arXiv:1805.12177*, 2018.
- Jashojit Mukherjee, K Praveen, and Venugopala Madumbu. Visual quality enhancement of images under adverse weather conditions. In *ITSC*, 2018.
- Chris H. Bahnsen and Thomas B. Moeslund. Rain removal in traffic surveillance: Does it matter? *arXiv:1810.12574*, 2018.
- Chris H. Bahnsen, David Vázquez, Antonio M. López, and Thomas B. Moeslund. Learning to remove rain in traffic surveillance by using synthetic data. In *VISIGRAPP*, 2019.

- Igor Vasiljevic, Ayan Chakrabarti, and Gregory Shakhnarovich. Examining the impact of blur on recognition by convolutional networks. *arXiv:1611.05760*, 2016.
- Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *ICLR*, 2019.
- Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3): 211–252, 2015.
- Georg Volk, Stefan Müller, Alexander von Bernuth, Dennis Hospach, and Oliver Bringmann. Towards robust CNN-based object detection through augmentation with synthetic rain variations. In *ITSC*, 2019.
- Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive faster R-CNN for object detection in the wild. In *CVPR*, 2018.
- Unghui Lee, Jiwon Jung, Seokwoo Jung, and David Hyunchul Shim. Development of a self-driving car that can handle the adverse weather. *International journal of automotive technology*, 2018.
- Christos Sakaridis, Dengxin Dai, Simon Hecker, and Luc Van Gool. Model adaptation with synthetic and real data for semantic dense foggy scene understanding. In *ECCV*, 2018b.
- Alexander Von Bernuth, Georg Volk, and Oliver Bringmann. Rendering physically correct raindrops on windshields for robustness verification of camera-based object recognition. *Intelligent Vehicles Symposium (IV)*, pages 922–927, 2018.
- Longyin Wen, Dawei Du, Zhaowei Cai, Zhen Lei, Ming-Ching Chang, Honggang Qi, Jongwoo Lim, Ming-Hsuan Yang, and Siwei Lyu. UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking. *arXiv:1511.04136*, 2015.
- Fisher Yu, Wenqi Xian, Yingying Chen, Fangchen Liu, Mike Liao, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving video database with scalable annotation tooling. *arXiv:1805.04687*, 2018.
- Zhengping Che, Guangyu Li, Tracy Li, Bo Jiang, Xuefeng Shi, Xinsheng Zhang, Ying Lu, Guobin Wu, Yan Liu, and Jieping Ye. D2-city: A large-scale dashcam video dataset of diverse traffic scenarios. *arXiv:1904.01975*, 2019.
- Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nusenes: A multimodal dataset for autonomous driving. *arXiv:1903.11027*, 2019.
- Adrien Gaidon, Qiao Wang, Yohann Cabon, and Eleonora Vig. Virtual worlds as proxy for multi-object tracking analysis. In *CVPR*, 2016.
- German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M. Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *CVPR*, 2016.
- Stephan R. Richter, Zeeshan Hayder, and Vladlen Koltun. Playing for benchmarks. In *ICCV*, 2017.
- M. Johnson-Roberson, Charles Barto, Rounak Mehta, Sharath Nittur Sridhar, Karl Rosaen, and Ram Vasudevan. Driving in the matrix: Can virtual worlds replace human-generated annotations for real world tasks? In *ICRA*, 2017.
- Peter Radecki, Mark Campbell, and Kevin Matzen. All weather perception: Joint data association, tracking, and classification for autonomous ground vehicles. *CoRR*, abs/1605.02196, 2016. URL <http://arxiv.org/abs/1605.02196>.
- Kexin Pei, Yinzi Cao, Junfeng Yang, and Suman Jana. Towards practical verification of machine learning: The case of computer vision systems. *arXiv:1712.01785*, 2017.

- Harshitha Machiraju and Sumohana Channappayya. An evaluation metric for object detection algorithms in autonomous navigation systems and its application to a real-time alerting system. In *25th IEEE International Conference on Image Processing (ICIP)*, 2018.
- Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 2010.
- Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *CVPR*, 2012.
- Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In *ECCV*, 2014.
- Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016.
- Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ADE20K dataset. In *CVPR*, 2017.
- Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulò, and Peter Kotschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *ICCV*, 2017.
- Ivan Krasin, Tom Duerig, Neil Alldrin, Vittorio Ferrari, Sami Abu-El-Haija, Alina Kuznetsova, Hassan Rom, Jasper Uijlings, Stefan Popov, Shahab Kamali, Matteo Mallocci, Jordi Pont-Tuset, Andreas Veit, Serge Belongie, Victor Gomes, Abhinav Gupta, Chen Sun, Gal Chechik, David Cai, Zheyun Feng, Dhyanesh Narayanan, and Kevin Murphy. Openimages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from <https://storage.googleapis.com/openimages/web/index.html>*, 2017.
- Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *ICCV*, 2017.
- Zhaowei Cai and Nuno Vasconcelos. Cascade R-CNN: Delving into high quality object detection. In *CVPR*, 2018.
- Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. Hybrid task cascade for instance segmentation. In *CVPR*, 2019a.
- Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal Loss for Dense Object Detection. *ICCV*, 2017a.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.
- Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature Pyramid Networks for Object Detection. In *CVPR*, 2017b.
- Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *CVPR*, 2017.
- Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *ICCV*, 2017.
- Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. *arXiv:1811.11168*, 2018.
- Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, et al. Mmdetection: Open mmlab detection toolbox and benchmark. *arXiv:1906.07155*, 2019b.
- Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016.

Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*, pages 1501–1510, 2017.

Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch SGD: Training ImageNet in 1 hour. *arXiv:1706.02677*, 2017.

Agrim Gupta, Piotr Dollar, and Ross Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *CVPR*, 2019.

## Appendix

### A Implementation details: Model training

We train all our models with two images per GPU which corresponds to a batch size of 16 on eight GPUs. On COCO, we resize images so that their short edge is 800 pixels and train for twelve epochs with a starting learning rate of 0.01 which is decreased by a factor of ten after eight and eleven epochs. On PASCAL VOC, images are resized so that their short edge is 600 pixels. Training is done for twelve epochs with a starting learning rate of 0.00125 with a decay step of factor ten after nine epochs. For Cityscapes, we stayed as close as possible to the procedure described in [He et al., 2017], rescaling images to a shorter edge size between 800 and 1024 pixels and train for 64 epochs (to match 24k steps at a batch size of eight) with an initial learning rate of 0.0025 and a decay step of factor ten after 48 epochs. For evaluation, only one scale (1024 pixels) is used. Specifically, we used four GPUs to train the COCO models and one GPU for all other models<sup>9</sup> Training with stylized data is done by simply exchanging the dataset folder or adding it to the list of dataset folders to consider. For all further details please refer to the config files in our implementation (which we will make available after the end of the anonymous review period).

### B Corrupting arbitrary images

In the original corruption benchmark of ImageNet-C [Hendrycks and Dietterich, 2019], two technical aspects are hard-coded: The image-dimensions and the number of channels. To allow for different data sets with different image dimensions, several corruption functions are defined independently of each other, such as `make_cifar_c`, `make_tinyimagenet_c`, `make_imagenet_c` and `make_imagenet_c_inception`. Additionally, many corruptions expect quadratic images. We have modified the code to resolve these constraints and now all corruptions can be applied to non-quadratic images with varying sizes, which is a necessary prerequisite for adapting the corruption benchmark to the PASCAL VOC and COCO datasets. For the corruption type Frost, crops from provided images of frost are added to the input images. Since images in PASCAL VOC and COCO have arbitrarily large dimensions, we resize the frost images to fit the largest input image dimension if necessary. The original corruption benchmark also expects RGB images. Our code now allows for grayscale images.<sup>10</sup> Both `motion_blur` and `snow` relied on the motion-blur functionality of `Imagemagick`, resulting in an external dependency that could not be resolved by standard Python package managers. For convenience, we reimplemented the motion-blur functionality in Python and removed the dependency on non-Python software.

### C BDD100k

We use the weather annotations present in the BDD100k dataset Yu et al. [2018] to split it in images with clear, rainy and snowy conditions. We disregard all images which are annotated to have any other weather condition (foggy, partly cloudy, overcast and undefined) to make the separation easier<sup>11</sup>. We use all images from the training set which are labeled having clear weather conditions for training. For testing, we created 3 subsets of the validation set each containing 725 images in clear, rainy or snowy conditions<sup>12</sup>. The sets were created to have the same size which was determined by the category with the least images (rainy). Having same sized test sets is important because evaluation under the AP metric leads to lower scores with increasing sequence length [Gupta et al., 2019].

<sup>9</sup>In all our experiments, we employ the linear scaling rule [Goyal et al., 2017] to select the appropriate learning rate.

<sup>10</sup>There are approximately 2–3% grayscale images in PASCAL VOC/MS COCO.

<sup>11</sup>It would have been great to combine the performance on natural fog with the results from Foggy Cityscapes but as there are only 13 foggy images in the validation set the results cannot be seen as representative in any way

<sup>12</sup>We will release the datasets splits at <https://github.com/bethgelab/robust-detection-benchmark>

MS COCO				
model	backbone	clean	corr.	rel.
		P [AP]	mPC [AP]	rPC [%]
Mask	r50	34.2	16.8	49.1
Cascade Mask	r50	35.7	17.6	49.3
HTC	x101-64x4d	43.8	28.1	64.0

Cityscapes				
model	backbone	clean	corr.	rel.
		P [AP]	mPC [AP]	rPC [%]
Mask	r50	32.7	10.0	30.5

Table 6: **Instance segmentation** performance of various models. Backbones indicated with  $r$ : ResNet. All model names indicate the corresponding model from the R-CNN family. All models were downloaded from the `mm detection` modelzoo.

train data	MS COCO			Cityscapes		
	clean [P]	corr. [mPC]	rel. [rPC]	clean [P]	corr. [mPC]	rel. [rPC]
standard	<b>34.2</b>	16.9	49.4	<b>32.7</b>	10.0	30.5
stylized	20.5	13.2	<b>64.1</b>	23.0	11.3	<b>49.2</b>
combined	32.9	<b>19.0</b>	57.7	32.1	<b>14.9</b>	46.3

Table 7: **Instance segmentation** performance of Mask R-CNN trained on standard images, stylized images and the combination of both evaluated on standard test sets (test 2007 for PASCAL VOC; val 2017 for MS COCO, val for Cityscapes).

## D Additional Results

### D.1 Instance Segmentation Results

We evaluated Mask R-CNN and Cascade Mask R-CNN on instance segmentation. The results are very similar to those on the object detection task with a slightly lower relative performance (1%, see Table 6). We also trained Mask R-CNN on the stylized datasets finding again very similar trends for the instance segmentation task as for the object detection task (Table 7). On the one hand, this is not very surprising as Mask R-CNN and Faster R-CNN are very similar. On the other hand, the contours of objects can change due to the stylization process, which would expectedly lead to poor segmentation performance when training only on stylized images. We do not see such an effect but rather find the instance segmentation performance of Mask R-CNN to mirror the object detection performance of Faster R-CNN when trained on stylized images.

### D.2 Deformable Convolutional Networks

We tested the effect of deformable convolutions [Dai et al., 2017, Zhu et al., 2018] on corruption robustness. Deformable convolutions are a modification of the backbone architecture exchanging some standard convolutions with convolutions that have adaptive filters in the last stages of the encoder. It has been shown that deformable convolutions can help on a range of tasks like object detection and instance segmentation. This is the case here too as networks with deformable convolutions do not only perform better on clean but also on corrupted images improving relative performance by 6-7% compared to the baselines with standard backbones (See Tables 8 and 9). The effect appears to be the same as for other backbone modifications such as using deeper architectures (See Section 3 in the main paper).

### Image rights & attribution

Figure 1: Home Box Office, Inc. (HBO).

MS COCO				
model	backbone	clean P [AP]	corr. mPC [AP]	rel. rPC [%]
Faster	r50-dcn	40.0	22.4	56.1
Faster	x101-64x4d-dcn	43.4	26.7	61.6
Mask	r50-dcn	41.1	23.3	56.7

Table 8: **Object detection** performance of models with deformable convolutions Dai et al. [2017]. Backbones indicated with *r* are ResNet, the addition *dcn* signifies deformable convolutions in stages *c3-c5*. All model names indicate the corresponding model from the R-CNN family. All models were downloaded from the `mm detection` modelzoo.

MS COCO				
model	backbone	clean P [AP]	corr. mPC [AP]	rel. rPC [%]
Mask	r50-dcn	37.2	20.7	55.7

Table 9: **Instance segmentation** performance of Mask R-CNN with deformable convolutions [Dai et al., 2017]. The backbone indicated with *r* is a ResNet 50, the addition *dcn* signifies deformable convolutions in stages *c3-c5*. The model was downloaded from the `mm detection` modelzoo.



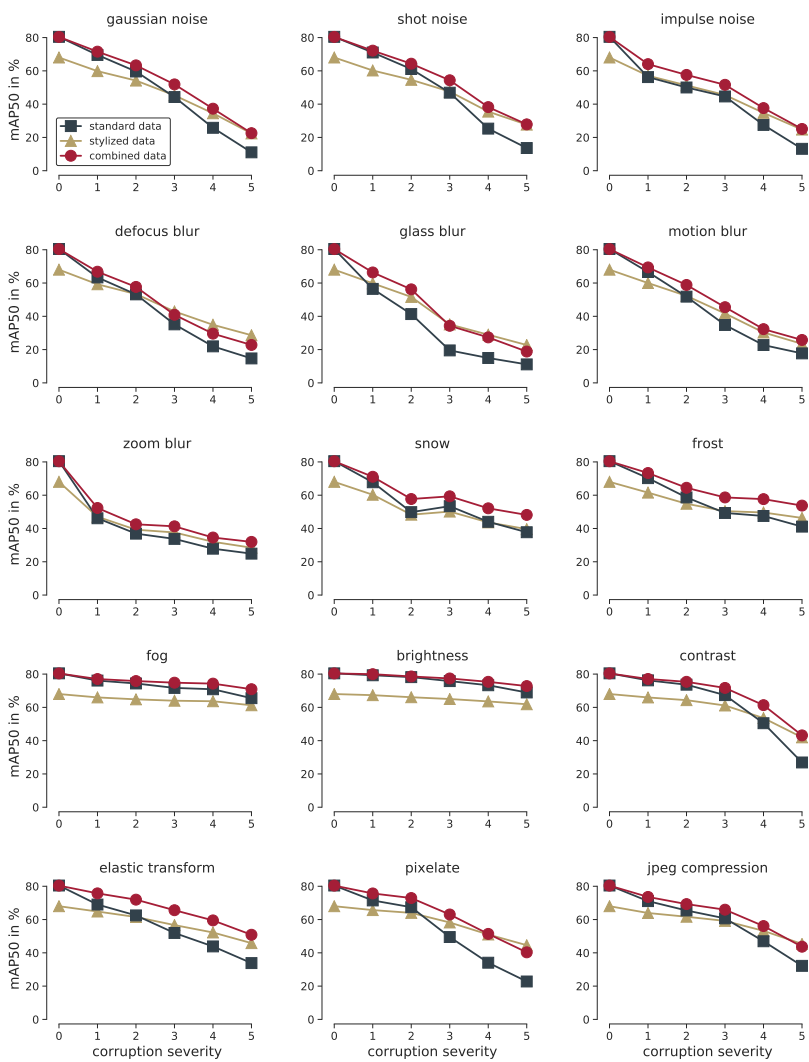


Figure 7: Results for each corruption type on PASCAL-C.

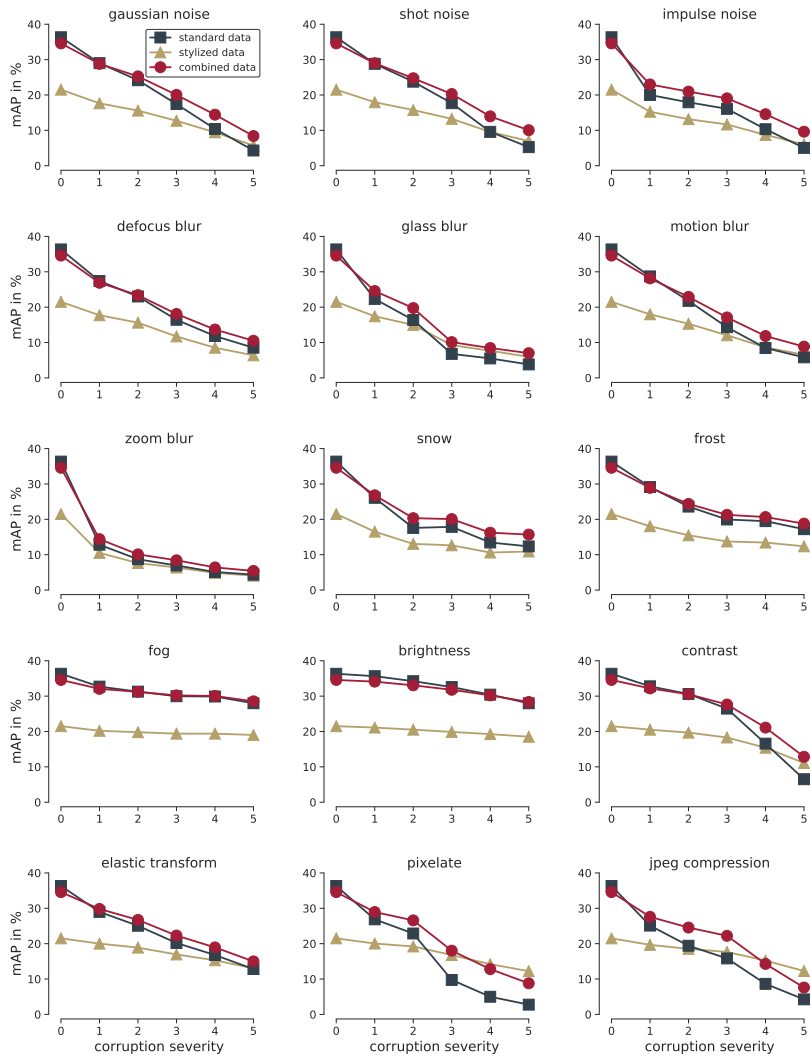


Figure 8: Results for each corruption type on COCO-C.

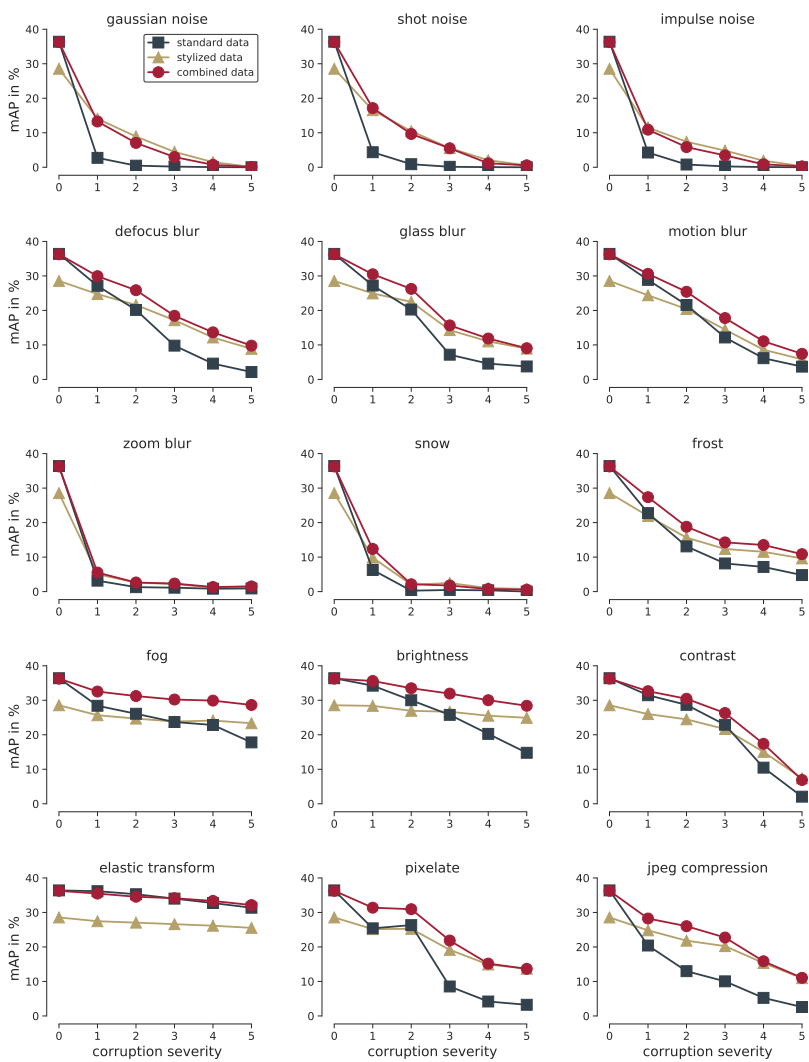


Figure 9: Results for each corruption type on Cityscapes-C.

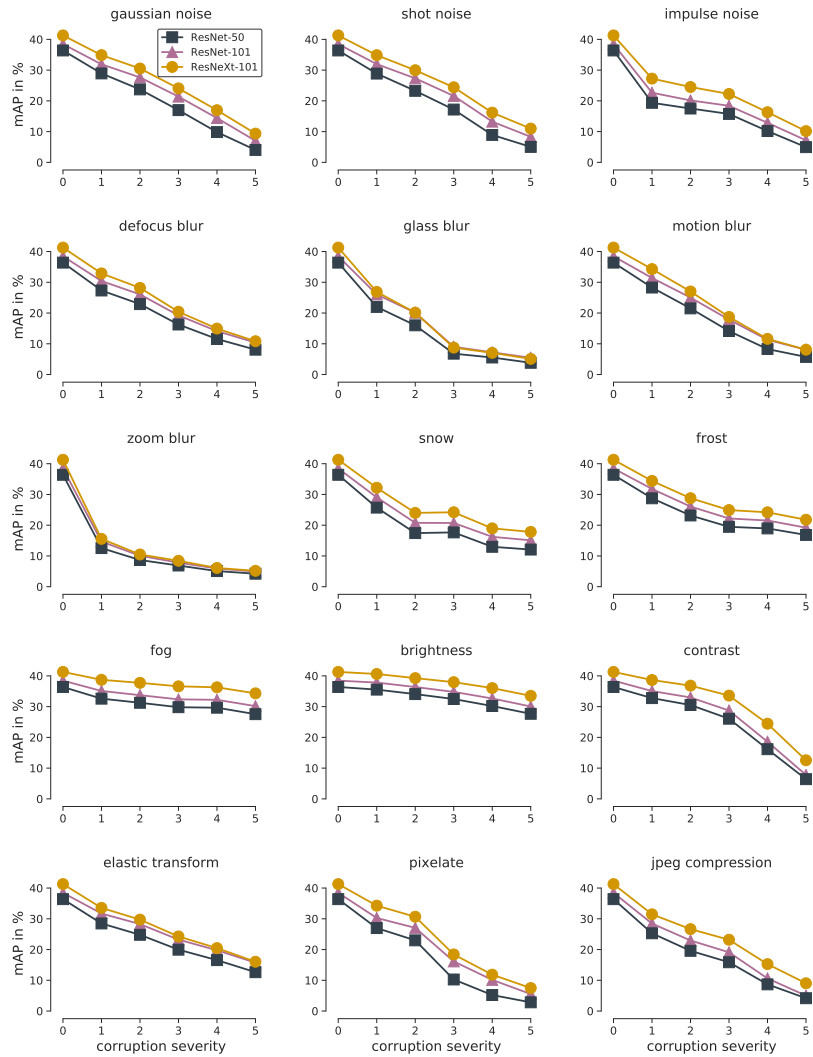


Figure 10: Results for each corruption type using different backbones. Faster R-CNN trained on MS COCO with ResNet-50, ResNet-101 and ResNext-101\_64x4d backbones.

## 2.3 Beyond accuracy: quantifying trial-by-trial behaviour of CNNs and humans by measuring error consistency

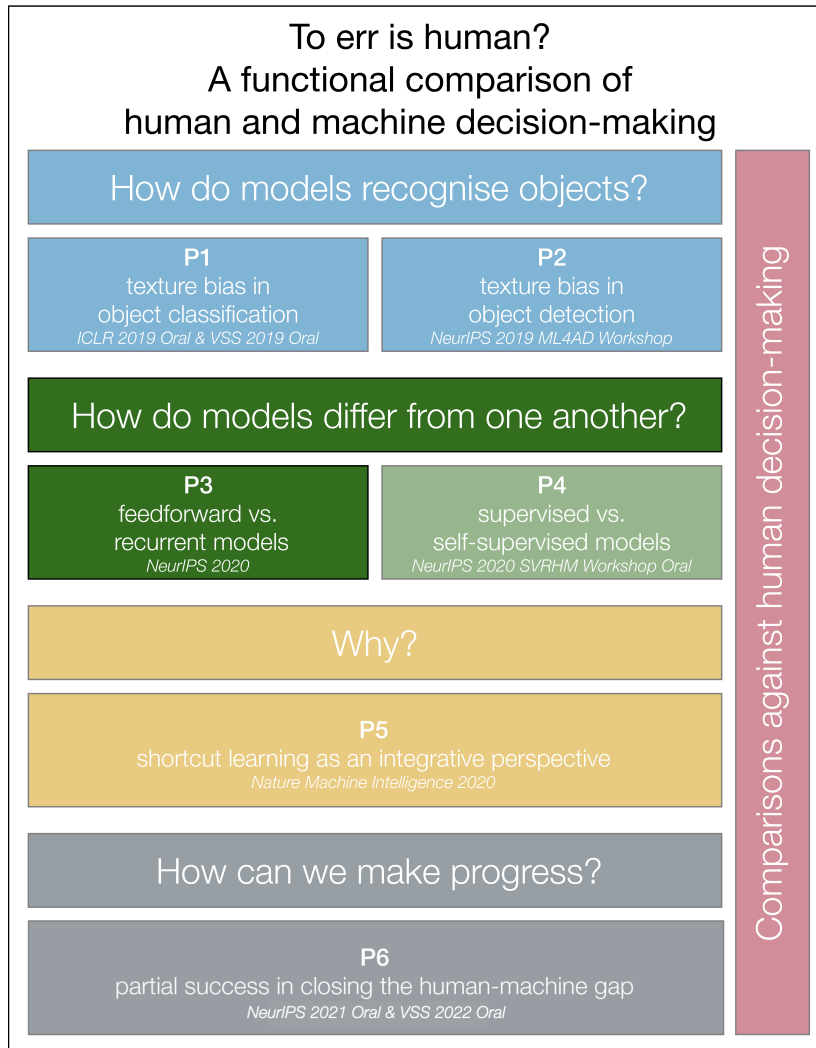


Figure 2.3: Schematic overview of the projects presented in this thesis. Projects P1–P4 ask “How do machines decide”, project P5 presents the concept of shortcut learning as an integrative perspective tackling the question “Why do machines decide the way they do”. Finally, project P6 presents a comprehensive benchmark to measure future progress, and reports first signs of (partial) success. Throughout the thesis, machine decision making will be compared against human decision making.

---

## Beyond accuracy: quantifying trial-by-trial behaviour of CNNs and humans by measuring error consistency

---

**Robert Geirhos\***  
University of Tübingen & IMPRS-IS  
robert.geirhos@uni-tuebingen.de

**Kristof Meding\***  
University of Tübingen  
kristof.meding@uni-tuebingen.de

**Felix A. Wichmann**  
University of Tübingen  
felix.wichmann@uni-tuebingen.de

\* Joint first authors (alphabetical order)

### Abstract

A central problem in cognitive science and behavioural neuroscience as well as in machine learning and artificial intelligence research is to ascertain whether two or more decision makers—be they brains or algorithms—use the same strategy. Accuracy alone cannot distinguish between strategies: two systems may achieve similar accuracy with very different strategies. The need to differentiate beyond accuracy is particularly pressing if two systems are at or near ceiling performance, like Convolutional Neural Networks (CNNs) and humans on visual object recognition. Here we introduce trial-by-trial *error consistency*, a quantitative analysis for measuring whether two decision making systems systematically make errors on the same inputs. Making consistent errors on a trial-by-trial basis is a necessary condition if we want to ascertain similar processing strategies between decision makers. Our analysis is applicable to compare algorithms with algorithms, humans with humans, and algorithms with humans.

When applying error consistency to visual object recognition we obtain three main findings: (1.) Irrespective of architecture, CNNs are remarkably consistent with one another. (2.) The consistency between CNNs and human observers, however, is little above what can be expected by chance alone—indicating that humans and CNNs are likely implementing very different strategies. (3.) CORnet-S, a recurrent model termed the “current best model of the primate ventral visual stream”, fails to capture essential characteristics of human behavioural data and behaves essentially like a standard purely feedforward ResNet-50 in our analysis; highlighting that certain behavioural failure cases are not limited to feedforward models. Taken together, error consistency analysis suggests that the strategies used by human and machine vision are still very different—but we envision our general-purpose error consistency analysis to serve as a fruitful tool for quantifying future progress.

### 1 Introduction<sup>1</sup>

Complex systems are notoriously difficult to understand—be they Convolutional Neural Networks (CNNs) or the human mind or brain. Paradoxically, for CNNs, we have access to every single model parameter, know exactly how the architecture is formed of stacked convolution layers, and

<sup>1</sup>Blog post summary: <https://medium.com/@robertgeirhos/are-all-cnns-created-equal-d13a33b0caf7>

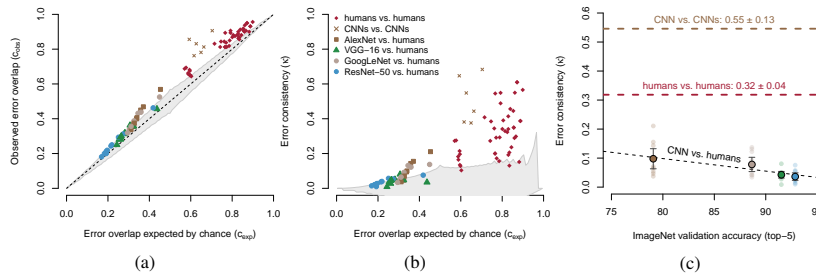


Figure 1: Do humans and CNNs make consistent errors? From left to right three steps for analysing this question are visualised. For a detailed description of these steps please see the intuition (1.1). (a) Observed vs. expected error overlap (errors on the same trials) for a classification experiment where humans and CNNs classified the same images [11]. Values above the diagonal indicate more overlap than expected by chance. (b) Same data as on the left but measured by error consistency ( $\kappa$ ). Higher values indicate greater consistency; shaded areas correspond to a simulated 95% percentile for chance-level consistency. (c) Error consistency vs. ImageNet accuracy.

we can inspect every single pixel of the training data—yet understanding the behaviour emerging from these primitives has proven surprisingly challenging [1], leaving us continually struggling to reconcile the success story of CNNs with their brittleness [2–4].<sup>2</sup> In response to the need to better understand the internal mechanisms, a number of visualisation methods have been developed [6–8]. And while many of them have proven helpful in fuelling intuitions, some have later been found to be misleading [9, 10]; moreover, most visualisation analyses are qualitative at nature. On the other hand, quantitative comparisons of different algorithms like benchmarking model accuracies have led to a lot of progress across deep learning, but reveal little about the internal mechanism: two models may reach similar levels of accuracy with very different internal processing strategies, an aspect that is gaining importance as CNNs are rapidly approaching ceiling performance across tasks and datasets. In order to understand whether two algorithms are implementing a similar or a different strategy, we need analyses that are quantitative *and* allow for drawing conclusions about the internal mechanism.

We here introduce *error consistency*<sup>3</sup>, a quantitative analysis for measuring whether two black-box perceptual systems systematically make errors on the same inputs. Irrespective of any potential differences at Marr’s implementational level [12] (which may be quite large, e.g. between two different neural network architectures or even larger between a CNN and a human observer), one can only conclude that two systems use a similar strategy if these systems make similar errors: not just a similar number of errors (as measured by accuracy), but also errors on the same inputs, i.e. if two systems find the same *individual* stimuli difficult or easy (as measured by error consistency). An agreement can be considered inverse to the Reichenbach-principle [13] of correlation: correlation between variables does not imply a direct causal relationship. However, correlation does imply *at least* an indirect causal link through other variables. For error consistency, zero error consistency implies that two decision makers are not using the same strategy. While error consistency can be applied across fields, tasks and domains (including vision, auditory processing, etc.), we believe it to be of particular relevance at the intersection of deep learning, neuroscience and cognitive science. Both brains and CNNs have, at various points, been described as black-box mechanisms [14–16]. But do the spectacular advances in deep learning shed light on the perceptual and cognitive processes of biological vision? Does similar performance imply similar mechanism or algorithm? Do different CNNs indeed make different errors?<sup>4</sup> We believe that fine-grained analysis techniques like error consistency may serve an important purpose in this debate.

<sup>2</sup>Note again the parallel in neuroscience, even for very simple brains: The nervous system of the nematode *C.elegans* is basically known in its entirety—still it is not fully understood how the (comparatively) complex behaviour of *C.elegans* is brought about by the biological “hardware” [5].

<sup>3</sup>For a discussion of this terminology we refer to Section S.1 in the appendix

<sup>4</sup>[17] found surprising similarities for self-supervised vs. supervised CNNs using error consistency.

**Molecular psychophysics.** Analysing errors for every single input is inspired by the idea of “molecular psychophysics” by David Green [18]. He argued that the goal of psychophysics should be to predict human responses to individual stimuli (trials) and not only aggregated responses (accuracy), let alone only averages across many individuals, as is common in much of the behavioural sciences. Green also predicted that once models of perceptual processes became more advanced, accuracy would cease to be a good criterion to assess and compare them rigorously (see p. 394 in [18]).

**Related work.** Using error consistency we can analyse human and CNN error patterns in a way that has, we believe, not been done before. We obtain *novel findings* but we do not consider error consistency to be an entirely *novel method* by itself. Instead, it builds on, extends and adapts existing methods and ideas developed in three different fields: molecular psychophysics (as described above) as well as causal inference and the social sciences (as described below). Our goal is the systematic analysis of human and CNN error patterns at the trial-by-trial level. Many previous analyses have focused on the aggregated level instead: In machine learning, performance is predominantly measured by accuracy and existing metrics to analyse errors such as comparisons between confusion matrices [19–23] or scores based on KL divergence [24] pool over single trials, thereby losing crucial information—they are not “molecular” but only “molar” in Green’s terminology [18]. [25, 26] went an important step further by comparing errors at an image-by-image level, but consistency was only computed *after* aggregating across participants, and [25] use a metric that automatically leads to higher consistency when comparing two systems with higher accuracy (without discounting for consistency due to chance). Closely related to our analysis is [27], who investigated similarity between models in the context of overfitting. In the context of causal inference, [28] performed a trial-by-trial analysis, plotting expected vs. observed behaviour (a starting point for our analysis). In social sciences, psychology and medicine, comparisons between participants are common, e.g. for problems like “How do people differ when answering a questionnaire?”. In that context, so-called inter-rater agreement is measured by Cohen’s kappa [29]. Here we repurpose and extend Cohen’s kappa ( $\kappa$ ) for the analysis of classification errors by humans and machines, and provide confidence intervals and analytical bounds (limiting possible consistency).

**Terminology.** A *decision maker* is any (living or artificial) entity that implements a decision rule. A *decision rule* is a function that defines a mapping from input to output (see [4] for a taxonomy of decision rules). Note that the same decision rule can result from different strategies. We use the term *strategy* synonymously with the term *algorithm*. For instance, Quicksort ( $X$ ) and Mergesort ( $X$ ) use a different algorithm (strategy), but they implement the same decision rule: the output will always be the same. *Permute* ( $X$ ), on the other hand, will (usually) lead to a different output. Hence, similar output (or similar errors, i.e., high error consistency) is a necessary, but not a sufficient condition for similar strategies.

## 1.1 Intuition

Before going through the mathematical details in Section 2, let us consider a simple example of a psychophysical experiment where human observers and CNNs classified objects from 160 images (line drawing / edge-like stimuli in this case). There are three steps in order to analyse error consistency (visualised in Figure 1). We can start by analysing how many of the decisions (either correct or incorrect) to individual trials are identical (*observed error overlap*). This number only becomes meaningful when plotted against the *error overlap expected by chance* (Figure 1a): for instance, two observers with high accuracies will necessarily agree on many trials by chance alone. However, this visualisation may be hard to interpret since higher values do not simply correspond to higher consistency (instead, above-chance consistency is measured by distance from the diagonal). In a second step, we can therefore normalise the data (Figure 1b) by dividing each datapoint’s distance to the diagonal by the total distance between the diagonal and ceiling (1.0). Now, we can directly compare the error consistency between decision makers: if error consistency is measured by  $\kappa$ , then  $\kappa = 0$  means chance-level consistency (independent processing strategies),  $\kappa > 0$  indicates consistency beyond chance (similar strategies) and  $\kappa < 0$  inconsistency beyond chance (inverse strategies). Lastly, we can analyse the relationship between error consistency ( $\kappa$ ) and an arbitrary other variable, for instance in order to determine whether better ImageNet accuracy leads to higher consistency between a CNN and human observers (Figure 1c), which is not the case here.



## 2 Methods

When comparing two decision makers the most obvious comparison is accuracy. Our goal is to go beyond accuracy per se by assessing the consistency of the responses with respect to individual stimuli. As a prerequisite, all decision makers need to evaluate the exact same stimuli. The order of presentation is irrelevant as long as the responses can be sorted w.r.t. stimuli afterwards.<sup>5</sup> In the following, we show how error consistency can be computed and which bounds and confidence intervals apply for the observed error overlap (2.1) and for  $\kappa$  (2.2). Experimental methods are described in 2.3 and code is available from <https://github.com/wichmann-lab/error-consistency>.

### 2.1 Observed vs. expected error overlap

If two observers  $i$  and  $j$  (be they algorithms, humans or animals) respond to the same  $n$  trials, we can investigate by how much their decisions overlap. For this purpose, we only analyse whether the decisions were correct/incorrect (irrespective of the number of choices). The observed error overlap  $c_{obs}$  is defined as  $c_{obs_{i,j}} = \frac{e_{i,j}}{n}$  where  $e_{i,j}$  is the number of equal responses (either both correct or both incorrect). In order to find out whether this observed overlap is beyond what can be expected by chance, we can compare observers  $i$  and  $j$  to a theoretical model: independent binomial observers (binomial: making either a correct or an incorrect decision; independent: only random consistency). In this case, we can expect only overlap due to chance  $c_{exp_{i,j}}$ :

$$c_{exp_{i,j}} = p_i p_j + (1 - p_i)(1 - p_j). \quad (1)$$

This is the sum of the probabilities that two observers  $i$  and  $j$  with accuracies  $p_i$  and  $p_j$  give the same **correct** and **incorrect** response by chance.<sup>6</sup>

**Confidence intervals.** Unfortunately, the confidence interval of  $c_{obs_{i,j}}$  in the scatter-plot of Figure 1a is not trivial to obtain. [28] used a standard binomial confidence interval. This is, however, only a very rough estimate of the true confidence interval since the position on the x-axis ( $c_{exp}$ ) itself is also estimated from the data and thus influenced by variation. We sample data for the null hypothesis of independent observers and calculate the corresponding 95% percentiles (cf. Figure 2). This process is described in Section S.3 in the appendix.

**Bounds.** Confidence intervals allow to investigate hypotheses. In addition, theoretical bounds might help to assess the degree of the observed consistency not being due to chance: a data point close or at the bound has maximum distance to the diagonal for a given value of  $c_{exp}$ . For this end we have calculated bounds of  $c_{obs}$  as an additional diagnostic tool. The influence of these bounds on the confidence intervals is visualised in Figure 2.

Ideally, we also want to express the bounds of  $c_{obs}$  directly as a function of  $c_{exp}$ . The analytical derivation of the bounds below can be found in the Appendix (S.2) and are visualised in Figure 2.

$$0 \leq c_{obs_{i,j}} \leq 1 - \sqrt{1 - 2c_{exp_{i,j}}} \quad \text{if } c_{exp_{i,j}} \leq 0.5, \quad (2)$$

$$\sqrt{2c_{exp_{i,j}} - 1} \leq c_{obs_{i,j}} \leq 1 \quad \text{if } c_{exp_{i,j}} \geq 0.5. \quad (3)$$

### 2.2 Error consistency measured by Cohen's kappa

$c_{obs_{i,j}}$  described above quantifies the observed error overlap between observers  $i$  and  $j$ . In order to obtain a single behavioural score for error consistency, that is, one disentangled from accuracy<sup>7</sup>, we need to discount for error overlap by chance  $c_{exp_{i,j}}$ . This is solved by Cohen's  $\kappa$  [29] with which we

<sup>5</sup>For human observers the order of presentation can make a (typically small) difference as human observers exhibit serial dependencies and other non-stationarities [18, 30]. Participants, e.g., may make more errors or lapses towards the end of an experiment due to fatigue [31]), and it is thus recommended to randomly shuffle presentation order for each participant to avoid such a "trivial" consistency of errors. Luckily, non-stationarities are usually only problematic if the signal levels are low, i.e. near chance performance.

<sup>6</sup>Note that  $c_{exp} > 0.5 \iff p_1, p_2 > 0.5 \vee p_1, p_2 < 0.5$ , see also Figure SF.2 in the appendix.

<sup>7</sup>In fact, error consistency is an accuracy corrected metric, see S.4 in the appendix

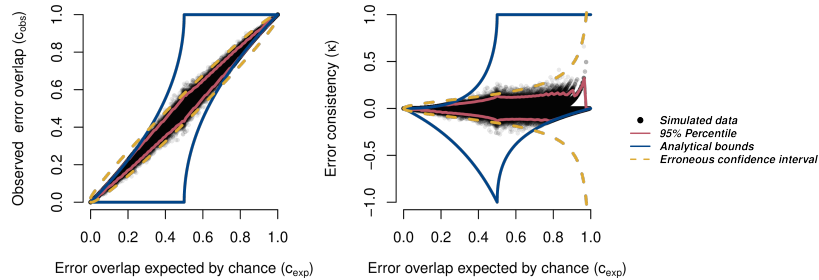


Figure 2: Simulated data of  $c_{exp}$ ,  $c_{obs}$  and  $\kappa$  for 160 trials under the assumption of independent decision makers. Analytical bounds and 95% percentile derived from the simulation of 100,000 experiments do not align with the often reported erroneous confidence interval.

measure error consistency:

$$\kappa_{i,j} = \frac{c_{obs_{i,j}} - c_{exp_{i,j}}}{1 - c_{exp_{i,j}}}. \quad (4)$$

We do not include a comparison of  $\kappa$  to the (Pearson) correlation coefficient since it has been shown that correlation is not a suitable measure of agreement [32, 33].

**Confidence intervals.** Confidence intervals of the average  $\kappa$  of groups, such as the average error consistency of humans vs. humans in Figure 1c, are based on the empirical standard error of the mean and a normal distribution assumption of the average error consistency (a numerical simulation of binomial observers confirmed that this assumption is valid here). Analogous to the observed consistency we use a sampling approach to obtain confidence intervals of  $\kappa$  given  $c_{exp}$ , see S.3 for details. This is necessary since the original confidence approximation interval derived by Cohen [29] (yellow dashes for error consistency in Figure 2) were later shown to be erroneous [34, 35].<sup>8</sup> While a corrected approximate version for individual kappas does exist [34, 38], there is to our knowledge no analytical or approximate confidence interval for  $\kappa$  given  $c_{exp}$ , and hence our sampling approach.

**Bounds.** The following bounds show the limits of  $\kappa$  given a specific value of  $c_{exp}$ , please see Section S.2 for the derivation and Figure 2 for visualisation<sup>9</sup>:

$$\frac{-c_{exp_{i,j}}}{1 - c_{exp_{i,j}}} \leq \kappa_{i,j} \leq \frac{1 - \sqrt{1 - 2c_{exp_{i,j}}} - c_{exp_{i,j}}}{1 - c_{exp_{i,j}}} \quad \text{if } c_{exp_{i,j}} \leq 0.5, \quad (5)$$

$$\frac{\sqrt{2c_{exp_{i,j}} - 1} - c_{exp_{i,j}}}{1 - c_{exp_{i,j}}} \leq \kappa_{i,j} \leq 1 \quad \text{if } c_{exp_{i,j}} \geq 0.5. \quad (6)$$

### 2.3 Experimental methods

**Stimuli: motivation.** Exemplary stimuli are visualised in Figure 3. We tested both “vanilla” images (plain unmodified colour images from ImageNet [40]) and three different types of out-of-distribution (o.o.d.) images. The motivation for using o.o.d. images is the following: Significant progress in neuroscience—e.g., discovering receptive fields of simple and complex cells—was made using “unnatural” bar-like stimuli. In deep learning, adversarial examples and texture bias were discovered by testing models on (unnatural) images different than the training data. Hence, we can learn a lot about the inner workings of a system by probing it with appropriate “artificial” stimuli [41, 42]; [4] even argues that o.o.d. testing is a necessity for drawing reliable inferences about a model’s strategy. Standard ImageNet images (where human and pre-trained CNN accuracies are both very high and similar,  $.960 \pm .036\%$ ) are included as a baseline condition.

<sup>8</sup>This erroneous confidence interval is still used in many publications, including very influential ones [36, 37].

<sup>9</sup>Bounds of kappa depending on  $c_{obs}$  instead of  $c_{exp}$  can be found in [39].

**Stimuli: method details.** [11] tested  $N=10$  human observers in their cue conflict, edge and silhouette experiments. Starting from normal images with a white background, different image manipulations were applied. For *cue conflict* images, the texture of a different image was transferred to this image using neural style transfer [43], creating a texture-shape cue conflict with a total of 1280 trials per observer and network. For *edge* stimuli, a standard edge detector was applied to the original images to obtain line-drawing-like stimuli (160 trials per observer). *Silhouette* stimuli were created by filling the outline of an object with black colour, leaving just the silhouette (160 trials per observer).<sup>10</sup> Lastly, *ImageNet* stimuli were standard coloured ImageNet images; we used the behavioural data ( $N=2$  observers) and stimuli from [44] for this experiment.

**Paradigm.** In order to compare the error consistency of two perceptual systems (e.g. CNNs and humans), those two systems a) need to be evaluated on the exact same stimuli and b) need to be in a regime with neither perfect accuracy nor chance-level performance. We found the publicly available stimuli and data from [11] to be an ideal test case. [11] compared object recognition abilities of humans and algorithms in a carefully designed psychophysical experiment. After a 200 ms presentation of a  $224 \times 224$  pixels image, observers had 16 categories to choose from (e.g. car, dog, chair). For ImageNet-trained networks, categorisation responses for 1,000 fine-grained classes were mapped to those 16 classes using the WordNet hierarchy [45]. In order to obtain the probability of a broad category (e.g. dog), response probabilities of all corresponding fine-grained categories (e.g. all ImageNet dog breeds) were averaged using the arithmetic mean.<sup>11</sup>

**Convolutional Neural Networks.** Human responses were compared against classification decisions of all available CNN models from the PyTorch model zoo (for torchvision version 0.2.2) and against a recurrent model, CORnet-S [46]. All CNNs were trained on ImageNet. Details here: S.5. Additionally, we analysed the relationship between model shape bias (induced by training on Stylized-ImageNet) and error consistency with human observers: S.6.

### 3 Results

If two perceptual systems or decision makers implement the same strategy they can be expected to systematically make errors on the same stimuli. In the following, we show how *error consistency* can be used within visual object recognition to compare algorithms with humans (Section 3.1) and algorithms with algorithms (Section 3.2).

#### 3.1 Comparing algorithms with humans: investigating whether better ImageNet models show higher error consistency with human behavioural data

In deep learning, there is a strong linear relationship between ImageNet accuracy and transfer learning performance [47]; in computational neuroscience, better categorisation accuracy improves the prediction of neural firing patterns [48]. But do better performing ImageNet models also make more human-like errors?

**Error consistency vs. model performance.** In Figure 3, we analyse the error consistency between human observers and sixteen standard ImageNet-trained CNNs. We find that humans to humans show a fair degree of consistency w.r.t. individual stimuli. That is, their agreement on which cats or chairs or cars are easy/hard to categorise is well beyond chance. Interestingly, CNN-to-CNN consistency is even higher than human-to-human consistency in all three experiments. This occurs despite the fact that human accuracies are higher than CNN accuracies across experiments: for instance in the silhouette experiment, the average human accuracy is 0.75 whereas the average CNN accuracy is 0.54 (see Table 1, supplementary information). However, the consistency between CNNs and humans is close to zero for two experiments (cue conflict stimuli and line drawings); a linear model fit indicates no improvement with better ImageNet validation accuracy:  $F(1, 158) = 0.086, p = .769, R^2 = 0.001$  for cue conflict and  $F(1, 158) = 0.478, p = .491, R^2 = 0.003$  for line drawing stimuli. For silhouettes, there is a significant *positive* relationship between ImageNet accuracy and error consistency with  $F(1, 158) = 53.530, p = 1.21 \cdot 10^{-11}, R^2 = 0.253$ ; for ImageNet images, on the other hand, there is a significant *negative* relationship between top-5 accuracy and error consistency with  $F(1, 30) = 8.162, p = .008, R^2 = 0.214$ .

<sup>10</sup>For parametrically distorted images (Appendix, Figure SF.7) we used the stimuli from [44].

<sup>11</sup>This aggregation is optimal. A derivation is included in the appendix of the arXiv version v3 of [44].

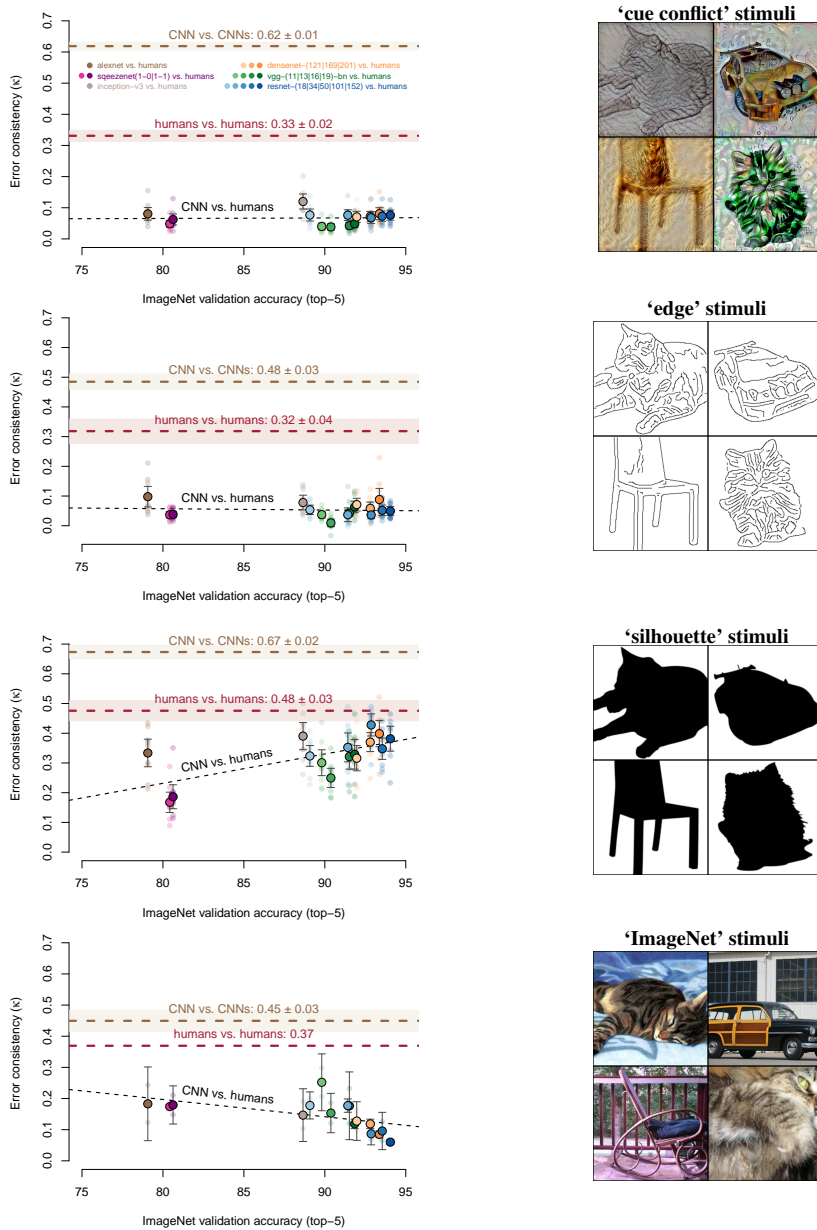


Figure 3: Do better ImageNet models make more human-like errors? Error consistency vs. top-5 ImageNet validation accuracy for four experiments: cue-conflict, edges, silhouettes and standard ImageNet images (exemplary stimuli are visualised on the right). Model colours as in Figure 4a; similar colours indicate same model family. Dashed black lines plot a linear model fit. Whiskers and colored tube show 95% confidence intervals around the mean. Small transparent circles indicate error consistency between a CNN and an individual human observer; mean consistency is shown as a larger saturated circle.

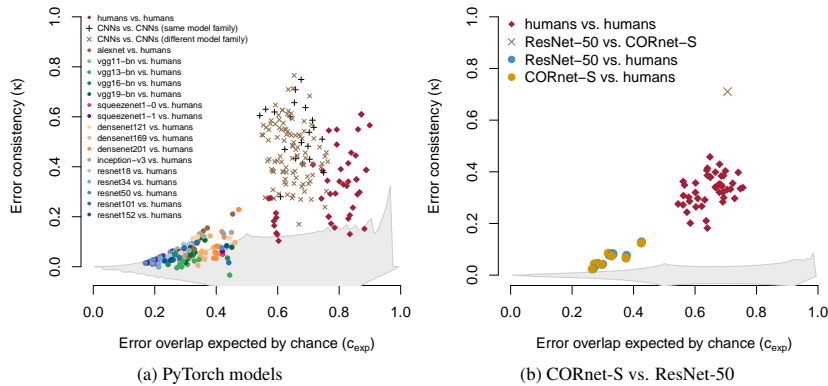


Figure 4: **(a)** How is error consistency influenced by model architecture? PyTorch models tested on edge stimuli (160 trials per observer). **(b)** Recurrent CORnet-S behaves just like a standard feedforward ResNet-50 on cue conflict stimuli (1280 trials). Shaded areas indicate a simulated 95% percentile for consistency by chance.

We conclude that there is a substantial algorithmic difference between human observers and the investigated sixteen CNNs: humans and CNNs are very likely implementing different strategies. This difference is narrowing down for silhouette stimuli, whereas it is as big as ever for cue conflict, line drawing and ImageNet stimuli: AlexNet from 2012 is just as error-consistent as recent models. Our results are in stark contrast to the observation that better ImageNet models appear to be better models of the primate visual cortex, even if they better predict neural activity [48].

**Error consistency vs. model architecture.** We were surprised to see that the consistency between different CNNs is even higher than the consistency between different human observers. In Figure 4a, we investigate the degree to which this CNN-CNN consistency is influenced by similarities in model architecture. When distinguishing between models from the same architecture family (e.g., all ResNet models) and models from a different model family (e.g., ResNet vs. VGG) we observe that even though models from the same family score higher on average, model-to-model consistency is generally very high.<sup>12</sup> In line with these results, [27] also reported extremely high similarity between different models on the ImageNet test set. This might shed some light on the finding that many trained and fitted CNNs predict neural data similarly well, largely irrespective of architecture [49]. Interestingly, the highest observed error consistency ( $\kappa = 0.793$ ) occurs for DenseNet-121 vs. ResNet-18: two models from a different model family with different depth (121 vs. 18 layers) and different connectivity. High error consistency between different CNNs suggests that using CNNs as an ensemble may currently be less effective than desirable, since ensembles benefit from independent (rather than consistent) models. It remains an open question why even multiple instances of a single model (trained with a different random seed) *internally* often differ substantially [50, 51], yet in spite of large architectural differences across models and model families, all CNNs that we investigated seem to be implementing fairly similar strategies.

### 3.2 Comparing algorithms with algorithms: the “current best model of the primate ventral visual stream” behaves like a vanilla ResNet-50 according to error consistency analysis

In order to understand how object recognition is achieved in brains, a necessary—but not sufficient—pre-requisite are quantitative metrics to track improvements and models that improve on those metrics. [46] went an important step in both directions by proposing Brain-Score, a benchmark where models can be ranked according to a number of metrics, for instance how well their activations predict how biological neurons fire when primates see the same images as an ImageNet-trained CNN. Using this benchmark, the authors tested hundreds of architectures to develop CORnet-S, a brain-inspired recurrent neural network. CORnet-S is able to capture recurrent dynamics (so-called object solution times) of monkey behaviour and achieves previously unmatched performance on

<sup>12</sup>Results for two other experiments are plotted in the appendix, Figure SF.3.

**Brain-Score** while retaining good ImageNet performance (73.1% top-1). These results, in the author’s words, “establish CORnet-S, a compact, recurrent ANN, as the current best model of the primate ventral visual stream” performing “brain-like object recognition” [46, p. 1]. Building such a model is an exciting undertaking and, as perhaps indicated by the highly competitive selection as an “Oral” contribution to NeurIPS 2019, an endeavour that sparked considerable excitement at the intersection of the neuroscience and machine learning communities. But how much is behavioural consistency improved in comparison to a baseline model (ResNet-50)? This is exactly the type of question that can be answered with the help of our error consistency analysis.

Figure 4 shows that CORnet-S shares only slightly above-chance error consistency with most human observers—even the highest CORnet-S-to-human error consistency is lower than the lowest human-to-human error consistency. However, there is no improvement whatsoever over a ResNet-50 baseline: Cohen’s  $\kappa$  for CNN-human consistency is very low for both models (.068: ResNet-50; .066: CORnet-S) compared to .331 for human-human consistency. Perhaps worse still, AlexNet from 2012 has higher error consistency than CORnet-S (.080). CNN-CNN consistency between CORnet-S and ResNet-50 is exceptionally high (.711), many datapoints even overlap exactly—a pattern confirmed by additional experiments in the appendix (Figures SF.5, SF.6 and SF.7), where we also perform a more detailed comparison to all six **Brain-Score** metrics (Figures SF.9, SF.10, SF.11 and SF.12 showing, if at all, only a weak relationship between error consistency and **Brain-Score** metrics). This indicates that CORnet-S is likely implementing a very different strategy than the human brain: in our analysis, CORnet-S has more behavioural similarities with a standard feedforward ResNet-50 than with human object recognition.<sup>13</sup> This provides evidence that recurrent computations—often argued to be one of the key missing ingredients in standard CNNs towards a better account of biological vision [46, 52–56]—do not necessarily lead to different behaviour compared to a purely feedforward CNN. It is still an open question to determine the conditions under which recurrence provides advantages over feedforward networks. Recent evidence seems to indicate that recurrence may be especially useful for difficult images [57–59].

Overall, the observed discrepancy between the leading score of CORnet-S on **Brain-Score** and its similarity to a standard ResNet-50 according to error consistency analysis points to the decisive importance of metrics: CORnet-S was mainly built for neural predictivity and while it scores very well on a number of other benchmarks, such as capturing object solution times and even a previously reported behavioural error analysis [26], it performs poorly on the behavioural metric reported here, *trial-by-trial error consistency*. New metrics to scrutinise models will hopefully lead to an improved generation of models, which in turn might inspire ever-more challenging analyses. An ideal model of biological object recognition would score well on multiple metrics (both neural and behavioural data, an important idea behind **Brain-Score**), including on metrics that the model was not directly optimised for.

#### 4 Conclusion

Error consistency is a quantitative analysis for comparing strategies/methods of black-box decision makers—be they brains or algorithms. Accuracy alone is insufficient for distinguishing between strategies: two decision makers may achieve similar accuracy with very different strategies. In contrast to aggregated metrics (averaging across trials/stimuli and observers/networks), error consistency measures behavioural errors on a fine-grained level following the idea of “molecular psychophysics” [18]. Using error consistency we find:

- Irrespective of architecture, CNNs are remarkably consistent with one another
- The consistency between humans and CNNs, however, is little beyond what can be expected by chance alone, indicating that CNNs still employ very different perceptual mechanisms and “brain-like machine learning” may be still but a distant dream (cf. [60])
- Recurrent CORnet-S, termed the “current best model of the primate ventral visual stream”, fails to capture essential characteristics of human behavioural data and instead behaves effectively like a standard feedforward ResNet-50 in our analysis.

Taken together, error consistency analysis suggests that the strategies used by human and machine vision are still very different—but we envision that error consistency will be a useful analysis in the quest to understand complex systems, be they CNNs or the human mind and brain.

<sup>13</sup>Interestingly, CORnet-S and ResNet-50 also score fairly similarly on a few metrics of **Brain-Score**.

## Broader Impact

*Error consistency* is a statistical analysis for measuring whether two or more decision makers make similar errors. Like any statistical analysis, it can be used for better or worse. For instance, as a very simple example, calculating the *mean* of a number of observations can be used to quantify a world-wide temperature increase caused by human carbon emissions [61, 62] (positive impact). However, calculating the mean could just as well be utilised by authoritarian governments to obtain an aggregated credit score of “social”—i.e., conformist—behaviour (negative impact) [63]. Concerning error consistency, we could envisage the following broader impact.

**Potential positive impact.** Quantifying differences between decision making strategies can contribute to a better understanding of algorithmic decisions. This improves model interpretability, which is a scientific goal by itself but also closely linked to societal requirements like accountability of algorithmic decision making and the “right to explanation” in the European Union [64]. Furthermore, calculating the error consistency between humans and CNNs can be used for fact-checking overly hyped “human-like AI” statements, e.g. by startups. We argue that human-level accuracy does not imply human-like decision making, which might contribute to increased rigour in model evaluation.

**Potential negative impact.** While not intended to cause any harm, quantifying differences between individuals can be used to identify group-conform and outlier behaviour. Furthermore, measuring error consistency between machines and humans might be used to quantify progress towards building machines that mimic human decision making on certain tasks. While this might sound exciting to a scientist, it very likely sounds a lot more frightening from the perspective of someone losing their job because a machine would then be capable of doing the same work more cheaply. Depending on the complexity of the task, this may not be a problem in the near future but, given current trends in the use of machine learning for automation, perhaps in the distant future.

## Acknowledgments & funding disclosure

Funding was provided, in part, by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – project number 276693517 – SFB 1233, TP 4 Causal inference strategies in human vision (K.M. and F.A.W.). The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting R.G; and the German Research Foundation through the Cluster of Excellence “Machine Learning—New Perspectives for Science”, EXC 2064/1, project number 390727645, for supporting F.A.W. The authors declare no competing interests.

We would like to thank Silke Gramer and Leila Masri for administrative and Uli Wannek for technical support; and David-Elias Künstle, Bernhard Lang, Maximus Mutschler as well as Uli Wannek for helpful comments. We thank Kubilius et al. [46] for making their implementation of CORnet-S publicly available. Furthermore, we would like to thank Jonas Kubilius and Martin Schrimpf for feedback and many valuable suggestions.

## Author contributions

Based on ideas from [65] and [28], R.G. first applied trial-by-trial analysis ideas to CNNs. Thereafter, all three authors jointly initiated the project. R.G. and K.M. jointly led the project. K.M. derived the bounds, performed and visualised the simulations and acquired the Brain-Score data (with input from R.G.). The CNN experiments were performed, analysed and visualised by R.G. (with input from K.M.). F.A.W. provided guidance, feedback, and pointed out the link to molecular psychophysics. All three authors planned and structured the manuscript. R.G. and K.M. wrote the paper with active input from F.A.W.

## References

- [1] T. P. Lillicrap and K. P. Kording. What does it mean to understand a neural network? *arXiv preprint arXiv:1907.06374*, 2019.
- [2] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.
- [3] A. Ilyas, S. Santurkar, D. Tsipras, L. Engstrom, B. Tran, and A. Madry. Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems*, pages 125–136, 2019.

- [4] R. Geirhos, J.-H. Jacobsen, C. Michaelis, R. Zemel, W. Brendel, M. Bethge, and F. A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, in press, 2020.
- [5] R. Mausfeld. No Psychology In - No Psychology Out. *Psychologische Rundschau*, 54(3):185–191, 2003.
- [6] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.
- [7] L. M. Zintgraf, T. S. Cohen, T. Adel, and M. Welling. Visualizing deep neural network decisions: Prediction difference analysis. *arXiv preprint arXiv:1702.04595*, 2017.
- [8] C. Olah, N. Cammarata, L. Schubert, G. Goh, M. Petrov, and S. Carter. Zoom in: An introduction to circuits. *Distill*, 5(3):e00024–001, 2020.
- [9] W. Nie, Y. Zhang, and A. Patel. A theoretical explanation for perplexing behaviors of backpropagation-based visualizations. *arXiv preprint arXiv:1805.07039*, 2018.
- [10] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems*, pages 9505–9515, 2018.
- [11] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, Felix A. Wichmann, and W. Brendel. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2019.
- [12] D. Marr. *Vision: A computational investigation into the human representation and processing of visual information*. W.H.Freeman & Co Ltd, San Francisco, 1982.
- [13] Hans Reichenbach. *The direction of time*. Univ of California Press, 1956.
- [14] D. Castelvecchi. Can we open the black box of AI? *Nature News*, 538:20–23, 2016.
- [15] R. Shwartz-Ziv and N. Tishby. Opening the black box of deep neural networks via information. *arXiv preprint arXiv:1703.00810*, 2017.
- [16] T. C. Kietzmann, P. McClure, and N. Kriegeskorte. Deep neural networks in computational neuroscience. *BioRxiv*, 2018.
- [17] Robert Geirhos, Kantharaju Narayanappa, Benjamin Mitzkus, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. On the surprising similarities between supervised and self-supervised models. *arXiv preprint arXiv:2010.08377*, 2020.
- [18] D. M. Green. Consistency of auditory detection judgments. *Psychological Review*, 71(5):392–407, 1964.
- [19] M. Ghodrati, A. Farzmaidi, K. Rajaei, R. Ebrahimpour, and S.-M. Khaligh-Razavi. Feedforward object-vision models only tolerate small image variations compared to human. *Frontiers in Computational Neuroscience*, 8:74, 2014.
- [20] R. Rajalingham, K. Schmidt, and J. J. DiCarlo. Comparison of object recognition behavior in human and monkey. *Journal of Neuroscience*, 35(35):12127–12136, 2015.
- [21] S. R. Kheradpisheh, M. Ghodrati, M. Ganjtabesh, and T. Masquelier. Deep networks can resemble human feed-forward vision in invariant object recognition. *Scientific Reports*, 6:32672, 2016.
- [22] S. R. Kheradpisheh, M. Ghodrati, M. Ganjtabesh, and T. Masquelier. Humans and deep networks largely agree on which kinds of variation make object recognition harder. *Frontiers in Computational Neuroscience*, 10:92, 2016.
- [23] R. Geirhos, D. H.J. Janssen, H. H. Schütt, J. Rauber, M. Bethge, and F. A. Wichmann. Comparing deep neural networks against humans: object recognition when the signal gets weaker. *arXiv preprint arXiv:1706.06969*, 2017.
- [24] W. J. Ma and B. Peters. A neural network walks into a lab: towards using deep nets as models for human behavior. *arXiv preprint arXiv:2005.02181*, 2020.
- [25] J. Kubilius, S. Bracci, and H. P. Op de Beeck. Deep neural networks as a computational model for human shape sensitivity. *PLoS Computational Biology*, 12(4):e1004896, 2016.
- [26] R. Rajalingham, E. B. Issa, P. Bashivan, K. Kar, K. Schmidt, and J. J. DiCarlo. Large-scale, high-resolution comparison of the core visual object recognition behavior of humans, monkeys, and state-of-the-art deep artificial neural networks. *Journal of Neuroscience*, 38(33):7255–7269, 2018.
- [27] Horia Mania, John Miller, Ludwig Schmidt, Moritz Hardt, and Benjamin Recht. Model similarity mitigates test set overuse. In *Advances in Neural Information Processing Systems*, pages 9993–10002, 2019.
- [28] K. Meding, D. Janzing, B. Schölkopf, and F. A. Wichmann. Perceiving the arrow of time in autoregressive motion. In *Advances in Neural Information Processing Systems*, pages 2303–2314, 2019.
- [29] J. Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.



- [30] I. Fründ, F. A. Wichmann, and J. H. Macke. Quantifying the effect of intertrial dependence on perceptual decisions. *Journal of Vision*, 14(7), 2014.
- [31] F. A. Wichmann and N. J. Hill. The psychometric function: I. Fitting, sampling, and goodness of fit. *Perception & Psychophysics*, 63(8):1293–1313, 2001.
- [32] R.J. Hunt. Percent agreement, pearson’s correlation, and kappa as measures of inter-examiner reliability. *Journal of Dental Research*, 65(2):128–130, 1986.
- [33] PF Watson and A Petrie. Method agreement analysis: a review of correct methodology. *Theriogenology*, 73(9):1167–1179, 2010.
- [34] J. L. Fleiss, J. Cohen, and B. S. Everitt. Large sample standard errors of kappa and weighted kappa. *Psychological Bulletin*, 72(5):323–327, 1969.
- [35] W. D. Hudson and C. W. Ramm. Correct formulation of the kappa coefficient of agreement. *Photogrammetric engineering and remote sensing*, 53(4):421–422, 1987.
- [36] M. L. McHugh. Interrater reliability: the kappa statistic. *Biochemia medica: Biochemia medica*, 22(3): 276–282, 2012.
- [37] M. Bland. *An introduction to medical statistics*. Oxford University Press (UK), 2015.
- [38] J. L. Fleiss, B. Levin, and M. C. Paik. *Statistical methods for rates and proportions*. J. Wiley, Hoboken, N.J, 3rd ed edition, 2003. ISBN 978-0-471-52629-2.
- [39] U. N. Umesh, R. A. Peterson, and M. H. Sauber. Interjudge agreement and the maximum value of kappa. *Educational and Psychological Measurement*, 49(4):835–850, 1989.
- [40] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.
- [41] Nicole C Rust and J Anthony Movshon. In praise of artifice. *Nature Neuroscience*, 8(12):1647–1650, 2005.
- [42] Marina Martinez-Garcia, Marcelo Bertalmío, and Jesús Malo. In praise of artifice reloaded: caution with natural image databases in modeling vision. *Frontiers in Neuroscience*, 13:8, 2019.
- [43] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2414–2423, 2016.
- [44] R. Geirhos, C. R. M. Temme, J. Rauber, H. H. Schütt, M. Bethge, and F. A. Wichmann. Generalisation in humans and deep neural networks. In *Advances in Neural Information Processing Systems*, pages 7538–7550, 2018.
- [45] G. A. Miller. WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41, 1995.
- [46] J. Kubilius, M. Schrimpf, K. Kar, R. Rajalingham, H. Hong, N. Majaj, E. Issa, P. Bashivan, J. Prescott-Roy, K. Schmidt, et al. Brain-like object recognition with high-performing shallow recurrent ANNs. In *Advances in Neural Information Processing Systems*, pages 12785–12796, 2019.
- [47] S. Kornblith, J. Shlens, and Q. V. Le. Do better imagenet models transfer better? In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2661–2671, 2019.
- [48] D. L. K. Yamins, H. Hong, C. F. Cadieu, E. A. Solomon, D. Seibert, and J. J. DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, 111(23):8619–8624, 2014.
- [49] Katherine R Storrs, Tim C Kietzmann, Alexander Walther, Johannes Mehrer, and Nikolaus Kriegeskorte. Diverse deep neural networks all predict human it well, after training and fitting. *bioRxiv*, 2020.
- [50] Arash Akbarinia and Karl R Gegenfurtner. Paradox in deep neural networks: Similar yet different while different yet similar. *arXiv preprint arXiv:1903.04772*, 2019.
- [51] Johannes Mehrer, Courtney J Spoerer, Nikolaus Kriegeskorte, and Tim C Kietzmann. Individual differences among deep neural network models. *bioRxiv*, 2020.
- [52] N. Kriegeskorte. Deep neural networks: a new framework for modeling biological vision and brain information processing. *Annual review of vision science*, 1:417–446, 2015.
- [53] C. J. Spoerer, P. McClure, and N. Kriegeskorte. Recurrent convolutional neural networks: a better model of biological object recognition. *Frontiers in psychology*, 8:1551, 2017.
- [54] T. C. Kietzmann, C. J. Spoerer, L. K. A. Sörensen, R. M. Cichy, O. Hauk, and N. Kriegeskorte. Recurrence is required to capture the representational dynamics of the human visual system. *Proceedings of the National Academy of Sciences*, 116(43):21854–21863, 2019.

- [55] T. Serre. Deep learning: the good, the bad, and the ugly. *Annual Review of Vision Science*, 5:399–426, 2019.
- [56] R. S. van Bergen and N. Kriegeskorte. Going in circles is the way forward: the role of recurrence in visual inference. *arXiv preprint arXiv:2003.12128*, 2020.
- [57] D. Linsley, J. Kim, V. Veerabadrán, C. Windolf, and T. Serre. Learning long-range spatial dependencies with horizontal gated recurrent units. In *Advances in Neural Information Processing Systems*, pages 152–164, 2018.
- [58] H. Tang, M. Schrimpf, W. Lotter, C. Moerman, A. Paredes, J. O. Caro, W. Hardesty, D. Cox, and G. Kreiman. Recurrent computations for visual pattern completion. *Proceedings of the National Academy of Sciences*, 115(35):8835–8840, 2018.
- [59] K. Kar, J. Kubilius, K. Schmidt, E. B. Issa, and J. J. DiCarlo. Evidence that recurrent circuits are critical to the ventral stream’s execution of core object recognition behavior. *Nature Neuroscience*, 22(6):974–983, 2019.
- [60] B. M. Lake, T. D. Ullman, J. B. Tenenbaum, and S. J. Gershman. Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40, 2017.
- [61] G. Hagedorn, P. Kalmus, M. Mann, S. Vicca, J. Van den Berge, J.-P. van Ypersele, D. Bourg, J. Rotmans, R. Kaaronen, S. Rahmstorf, et al. Concerns of young protesters are justified. *Science*, 364:139–140, 2019.
- [62] G. Hagedorn, T. Loew, S. I. Seneviratne, W. Lucht, M.-L. Beck, J. Hesse, R. Knutti, V. Quaschnig, J.-H. Schleimer, L. Mattauca, et al. The concerns of the young protesters are justified: A statement by scientists for future concerning the protests for more climate protection. *GAIA-Ecological Perspectives for Science and Society*, 28(2):79–87, 2019.
- [63] R. Creemers. China’s social credit system: an evolving practice of control. *Available at SSRN 3175792*, 2018.
- [64] A. D. Selbst and J. Powles. Meaningful information and the right to explanation. *International Data Privacy Law*, 7(4):233–242, 2017.
- [65] Vinzenz H. Schönfelder and Felix A. Wichmann. Identification of stimulus cues in narrow-band tone-in-noise detection using sparse observer models. *The Journal of the Acoustical Society of America*, 134(1):447–463, 2013.
- [66] R. J. Hyndman and Y. Fan. Sample quantiles in statistical packages. *The American Statistician*, 50(4):361–365, 1996.
- [67] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.
- [68] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2015.
- [69] Katherine L Hermann and Simon Kornblith. Exploring the origins and prevalence of texture bias in convolutional neural networks. *arXiv preprint arXiv:1911.09071*, 2019.

## Supplementary Material

Code and data to reproduce results and figures are available from <https://github.com/wichmann-lab/error-consistency>.

The supplementary material is structured as follows. We start with terminology in Section S.1, afterwards we derive bounds of  $c_{obs}$  and kappa in Section S.2 (limiting possible consistency), followed by a description of how we simulated the confidence intervals for  $c_{exp}$  and kappa under the null hypothesis of independent observers in Section S.3. Finally, we provide method details for Brain-Score and the evaluated CNNs in Section S.5 and report accuracies across experiments in Table 1.

In addition to method details, we provide extended experimental results in Figure SF.3 (error consistency of all PyTorch models for cue conflict and edge stimuli) as well as Figures SF.5, SF.6, SF.7, SF.8 (detailed analyses of CORnet-S vs. ResNet-50). Figures SF.9, SF.10 and SF.11 and SF.12 (investigating the relationship between Brain-Score metrics and error consistency).

Furthermore, Figure SF.4 visualises qualitative error differences by plotting which stimuli were particularly easy for humans and CNNs, respectively.

### S.1 Terminology: “error consistency”

We would like to briefly clarify the name *error consistency*. Our analysis helps to compare the consistency of two decision makers. Two decision makers necessarily show some degree of consistency due to chance agreement. Error consistency helps to examine whether the two decision makers show significantly more consistency than expected by chance by analysing behavioural error patterns. However, this analysis takes into account not only the consistency of errors but also the consistency of correctly answered trials, hence ‘error consistency’ may sound imprecise at first. Nonetheless, we believe that the term captures the most crucial aspect of this analysis: Humans and CNNs—which are particularly well suited for our analysis—are often close to ceiling performance or at least have high accuracies. Thus trials where the decision makers agree do not provide much evidence for distinguishing between processing strategies. In contrast, the (few) errors of the decision makers are the most informative trials in this respect: Hence the name error consistency.

### S.2 Derivation of bounds for $c_{obs}$ and kappa given $c_{exp}$

How much observed consistency can we expect at most for a given expected consistency? We assume two independent observers  $i$  and  $j$  with accuracies  $p_i$  and  $p_j$ . For given  $p_i, p_j$  only a certain range of  $c_{obs}$  is possible:

$$c_{obs_{max}} = 1 - |p_i - p_j| \text{ and } c_{obs_{min}} = |p_j + p_i - 1|. \quad (7)$$

Ideally, we also want to express the bounds of  $c_{obs}$  directly as a function of  $c_{exp}$ . We obtain the following bounds:

$$\begin{aligned} 0 \leq c_{obs_{i,j}} \leq 1 - \sqrt{1 - 2c_{exp_{i,j}}} & \quad \text{if } c_{exp_{i,j}} < 0.5, & (8) \\ \sqrt{2c_{exp_{i,j}} - 1} \leq c_{obs_{i,j}} \leq 1 & \quad \text{if } c_{exp_{i,j}} \geq 0.5. & (9) \end{aligned}$$

These bounds are visualised in Figure 2.

The derivation is as follows. We distinguish between two cases.

**Case 1:**  $p_i \leq 0.5 \ \& \ p_j \leq 0.5$  or  $p_i \geq 0.5 \ \& \ p_j \geq 0.5 \iff c_{exp_{i,j}} \geq 0.5$

The expected consistency then lies in the interval of  $[0.5, 1]$ , see Figure SF.2. First we calculate the upper bound  $b_{obs_{max}}$  given  $c_{exp_{i,j}}$ . Please note that a specific  $c_{exp_{i,j}}$  can be obtained by multiple combinations of values for  $p_i$  and  $p_j$ . For a given  $c_{exp_{i,j}}$  we choose  $p_j = p_i$ . We can calculate the exact value of  $p_i$  in this case with eq. (1). However since  $p_j = p_i$  we get with eq. (7) that  $b_{obs_{max}} = 1$ . Thus we directly obtain from eq. (7) that the upper bound of  $c_{obs_{i,j}}$  is always 1 for all  $c_{exp_{i,j}}$  in the interval  $[0.5, 1]$ .

It is a bit more challenging to derive the lower bound  $b_{obs_{min}}$  given  $c_{exp_{i,j}}$ . Using equation (7) and (1) we obtain

$$b_{obs_{min}} = p_i + \frac{c_{exp_{i,j}} + p_i - 1}{2p_i - 1} - 1. \quad (10)$$

Setting  $\frac{\partial b_{obs_{min}}}{\partial p_i} = 0$  to find the minimum results in

$$p_{i_{min}} = \frac{1}{2} \pm \sqrt{\frac{1}{4} - \frac{-2c_{exp_{i,j}} + 2}{4}}. \quad (11)$$

We only take the positive term in eq. (11) since  $p_i > 0.5$  by definition. Checking the second order derivative confirms a minimum. Finally using equation eq. (11) with eq. (10) we calculate

$$b_{obs_{min}} = \sqrt{2c_{exp_{i,j}} - 1}, \text{ thus} \quad (12)$$

$$\sqrt{2c_{exp_{i,j}} - 1} \leq c_{obs_{i,j}} \leq 1. \quad (13)$$

**Case 2:**  $p_i > 0.5 \ \& \ p_j < 0.5$  or  $p_i < 0.5 \ \& \ p_j > 0.5 \iff c_{exp_{i,j}} < 0.5$

The expected consistency then lies in the interval of  $[0, 0.5]$ , see Figure SF.2. This case is point symmetric to the right part. Thus we obtain for the bounds of the left part

$$b_{obs_{max2}} = 1 - b_{obs_{min}}(1 - c_{exp_{i,j}}), \quad (14)$$

$$b_{obs_{min2}} = 0 \text{ and finally} \quad (15)$$

$$0 \leq c_{obs_{i,j}} \leq 1 - \sqrt{1 - 2c_{exp_{i,j}}}. \quad (16)$$

**Bounds for kappa** If we plug in the bounds of  $c_{obs_{i,j}}$  into the equation of kappa, we obtain the following bounds for kappa:

$$\frac{-c_{exp_{i,j}}}{1 - c_{exp_{i,j}}} \leq \kappa_{i,j} \leq \frac{1 - \sqrt{1 - 2c_{exp_{i,j}}} - c_{exp_{i,j}}}{1 - c_{exp_{i,j}}} \quad \text{if } c_{exp_{i,j}} < 0.5, \quad (17)$$

$$\frac{\sqrt{2c_{exp_{i,j}} - 1} - c_{exp_{i,j}}}{1 - c_{exp_{i,j}}} \leq \kappa_{i,j} \leq 1 \quad \text{if } c_{exp_{i,j}} \geq 0.5. \quad (18)$$

### S.3 Calculating 95% percentiles of observed overlap and kappa for the null hypothesis of independent observers given an expected consistency

Here we describe the procedure to calculate 95% percentiles of  $\kappa$  and  $c_{obs}$ .

Our null hypothesis is that two decision makers are independent. Assuming independence, we can easily simulate these two observers. Based on  $p_i, p_j$  (the accuracies of decision makers  $i$  and  $j$ ) we sample  $n$  trials and calculate  $c_{exp_{i,j}}, c_{obs_{i,j}}$ , and  $\kappa_{i,j}$  accordingly based on these simulated values. This process is repeated systematically for different  $p_i$  and  $p_j$ . For this purpose we sample a grid of 4200 x 4200 points in the range  $[[0, 1], [0, 1]]$ . For each individual combination of  $p_i$  and  $p_j$ , the sampling is repeated five times, thus in total we simulate  $4200 \times 4200 \times 5 = 88,200,000$  values.<sup>14</sup>

The grid is not divided equally. 66% of  $p_i$  and  $p_j$  are located in the upper and lower 15% of the domain. This is important because kappa diverges for large values of  $c_{exp}$  (small and large values of  $p_i$  and  $p_j$ ); thus a dense sampling is necessary there.

Based on these simulated data we obtain 95% percentiles for  $c_{obs}$  and  $\kappa$ . We binned the data in 1% steps and used the standard quantile-function of R (type 7, see [66]). It is important to note that we have only a small number of trials (160 or 1280).<sup>15</sup> Therefore  $c_{obs}$  can take a maximum number of 161 or 1281 values respectively. The range of uniquely observed values is very small for a given  $c_{exp}$ . This implies that the accuracy of our percentiles is limited for data points that are very close to the quantiles. However, this does not influence our findings.

Please note that the denominator of kappa gets very small for high values of  $c_{exp}$ . Thus we see some instability of kappa towards high expected consistencies. Figure SF.1 shows diagnostic plots for both cases.

### S.4 Disentangling of Error consistency and Accuracy

Our argument for the disentanglement between kappa and accuracy is as follows. For independent observer no correlation between accuracy and kappa is observed, e.g. In Figure 2b,  $\kappa$  and  $c_{exp}$ <sup>16</sup> are not correlated ( $r = -0.00015, p > 0.05$ ). As expressed by the bounds in Figure 2,  $\kappa$  is limited by accuracy. If two observers have an accuracy for 90%, only certain levels of (dis-)agreement are possible. Error consistency (measured by  $\kappa$ ) aims to correct for accuracy and thus in our experiments different kinds of correlations between error consistency and overall accuracy occur. We observe zero correlation in (Figures 3a, 3b) and positive correlation in Figure 3c. In Figure 3d we observe a negative correlation between accuracy and error consistency. We conclude that there is

<sup>14</sup>The more values are simulated, the better: we chose the maximum number of samples feasible to simulate on our hardware within reasonable time.

<sup>15</sup>Percentiles for a different number of trials can also be computed with the code that we provide.

<sup>16</sup>Accuracy and  $c_{exp}$  are linked as one can see in figure SF.2

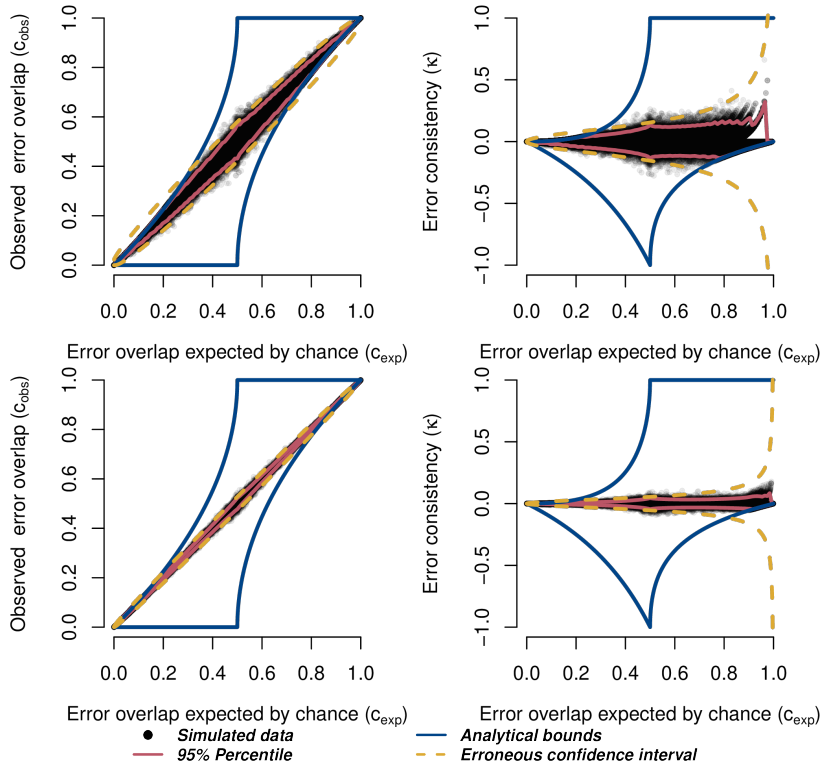


Figure SF.1: Simulated data of  $c_{exp}$ ,  $c_{obs}$  and  $\kappa$  for 160 (top) and 1280 (bottom) trials per block. Black dots show 100.000 randomly drawn blocks from our simulation. Blue lines show analytical bounds. Red lines show the 95% percentiles. Orange dashed lines show the wrong binomial confidence interval (left) and the erroneous confidence interval for  $\kappa$  (right) reported in many papers.

no correlation between consistency ( $\kappa$ ) and accuracy for independent observers whilst for dependent (consistent) observers correlations are possible. Kappa corrects for accuracy but is not independent from it.

### S.5 Method details for Brain-Score and CNNs

Human responses were compared against classification decisions of all available CNN models from the PyTorch model zoo (for torchvision version 0.2.2) [67], namely alexnet, vgg11-bn, vgg13-bn, vgg16-bn, vgg19-bn, squeezenet1-0, squeezenet1-1, densenet121, densenet169, densenet201, inception-v3, resnet18, resnet34, resnet50, resnet101, resnet152. For the VGG model family [68], we used the implementation with batch norm. CORnet-S, an additional recurrent model [46] analysed in Section 3.2, was obtained from the author's github implementation.<sup>17</sup> The comparison to Brain-Score in Figures SF.9, SF.10, SF.11 and SF.12 uses Brain-Score values obtained from the Brain-Score website (date of download: April 17, 2020) and error consistency values obtained by us. Note that the model implementations differ slightly: we consistently used PyTorch models whereas Brain-Score tested models from a few different frameworks (the full list can be seen here). Namely, squeezenet1-0, squeezenet1-1, resnet18, resnet-34 are identical (PyTorch); the VGG models use Keras instead (without batch norm) and so do the Brain-Score DenseNet models; inception\_v3, resnet50\_v1, resnet101\_v1, resnet152\_v1 are TF Slim models. Since model implementations usually differ slightly

<sup>17</sup><https://github.com/dicarlolab/CORnet>

across frameworks, a small variation in the results can be expected depending on the chosen model and framework.

### S.6 Error consistency of shape-biased models

We analyzed three CNNs with different degrees of stylized training data from [11]. Model shape bias predicts human-CNN error consistency for cue conflict stimuli, indicating that networks basing their decisions on object shape (rather than texture) make more human-like errors:

model shape bias (%)	20.5	21.4	34.7	81.4
human-CNN consistency ( $\kappa$ )	.066	.068	.098	.195

	observer / model	cue conflict	edge	silhouette
1	subject-01	0.69	0.89	0.80
2	subject-02	0.76	0.94	0.66
3	subject-03	0.84	0.93	0.80
4	subject-04	0.62	0.84	0.78
5	subject-05	0.85	0.89	0.77
6	subject-06	0.82	0.93	0.72
7	subject-07	0.76	0.81	0.76
8	subject-08	0.78	0.96	0.64
9	subject-09	0.86	0.61	0.76
10	subject-10	0.77	0.92	0.85
11	alexnet	0.19	0.29	0.43
12	vgg11-bn	0.12	0.14	0.46
13	vgg13-bn	0.12	0.25	0.36
14	vgg16-bn	0.14	0.22	0.47
15	vgg19-bn	0.15	0.28	0.46
16	squeezenet1-0	0.14	0.15	0.24
17	squeezenet1-1	0.17	0.14	0.29
18	densenet121	0.19	0.24	0.42
19	densenet169	0.21	0.33	0.53
20	densenet201	0.21	0.38	0.51
21	inception-v3	0.27	0.28	0.54
22	resnet18	0.19	0.20	0.47
23	resnet34	0.19	0.16	0.45
24	resnet50	0.18	0.14	0.54
25	resnet101	0.20	0.24	0.49
26	resnet152	0.21	0.21	0.56
27	cornet-s	0.18	0.25	0.46

Table 1: Accuracies for human observers and CNNs for all three experiments. In the cue conflict experiment case, an answer is counted as correct in this table if this answer corresponds to the correct shape category (other choices are possible).

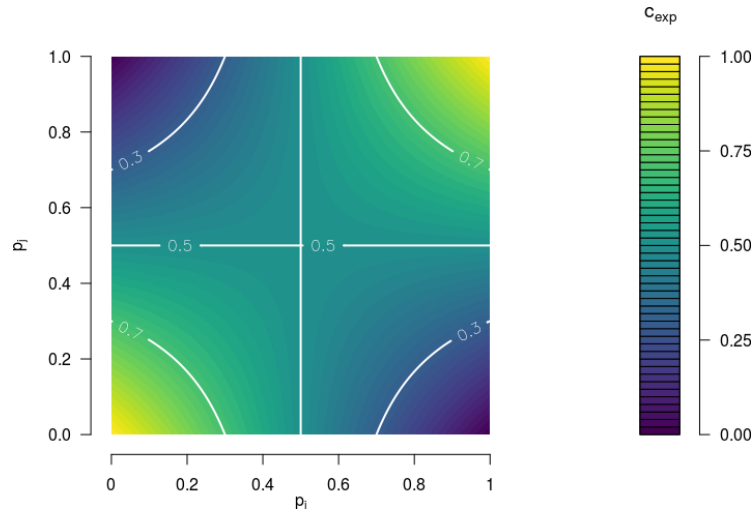
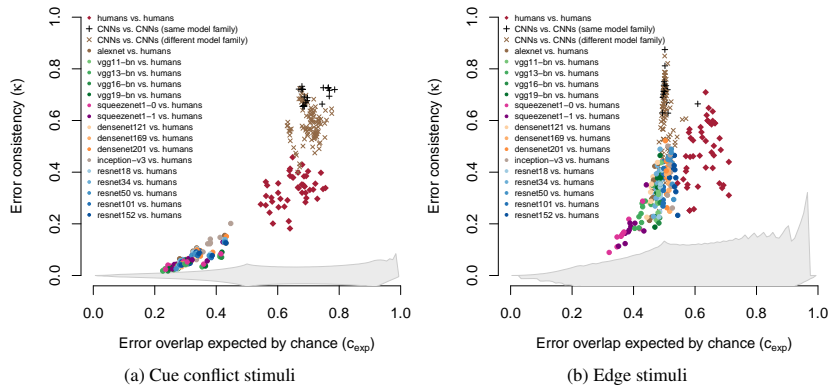


Figure SF.2: Values that  $c_{exp}$  can take depending on  $p_i$  and  $p_j$  for two independent observers.



(a) Cue conflict stimuli

(b) Edge stimuli

Figure SF.3: Error consistens vs. expected error overlap for all PyTorch models.

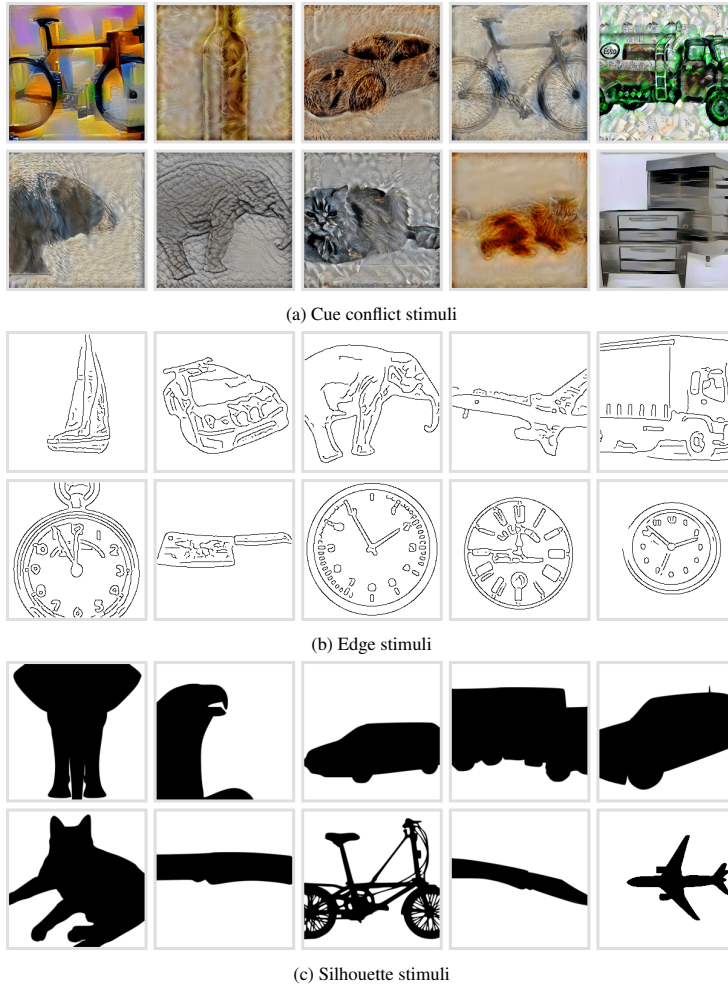


Figure SF.4: “Easy” stimuli for humans and CNNs. For each experiment, the images in the top row were those that most humans correctly classified. In the bottom row: stimuli that most CNNs correctly classified. If there were more than five images where humans were very accurate on, we here selected those where CNNs were the least accurate, and vice versa. ImageNet stimuli are not visualised due to image permission reasons.



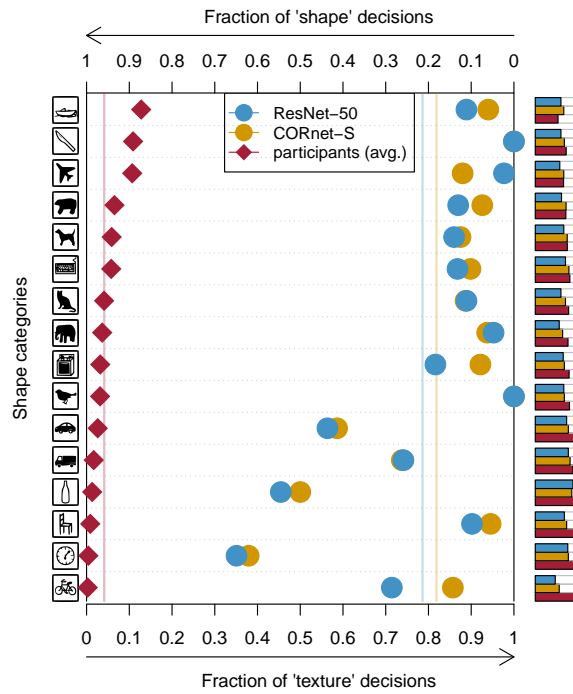


Figure SF.5: Shape bias of CORnet-S and ResNet-50 in comparison to human observers. Human observers categorise objects by shape rather than texture [11], which differentiates them from standard ImageNet-trained CNNs like ResNet-50 (categorising predominantly by texture). In this experiment, CORnet-S again behaves similarly to ResNet-50 but does not show a human-like shape bias as would be expected for an accurate model of human object recognition. Small bar plots on the right indicate accuracy (answer corresponds to either correct texture category or correct shape category). This pattern was also observed by Hermann and Kornblith [69], who performed a detailed investigation of the factors that influence model shape bias.

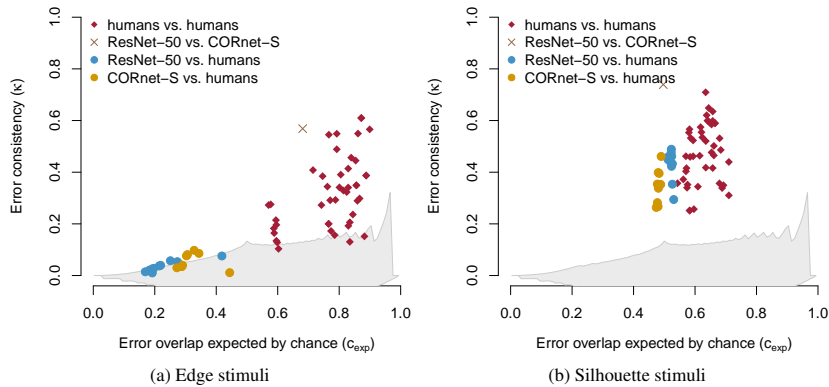


Figure SF.6: Error consistency of CORnet-S vs. ResNet-50 for edge and silhouette stimuli.

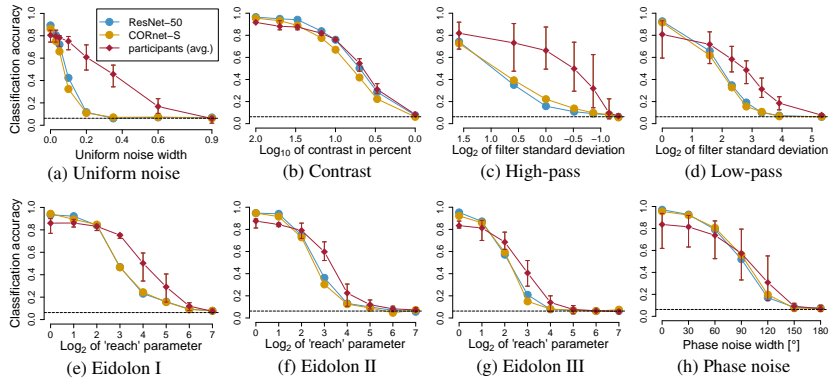


Figure SF.7: Classification accuracy on parametrically distorted images for ResNet-50, CORnet-S and human observers. Again, CORnet-S behaves like a ResNet-50 rather than like human observers.

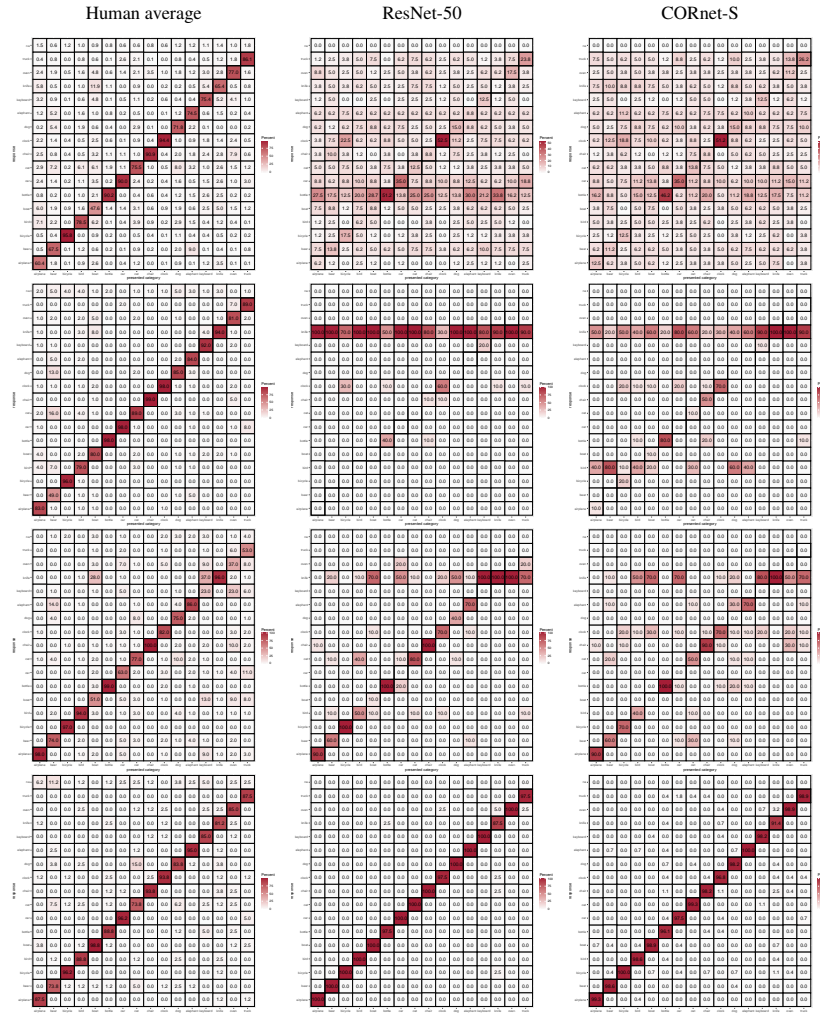


Figure SF.8: Confusion matrices for humans, ResNet-50 and CORnet-S. Different rows correspond to different experiments. Top row: cue conflict stimuli, second row: edge stimuli, third row: silhouette stimuli, last row: ImageNet stimuli.

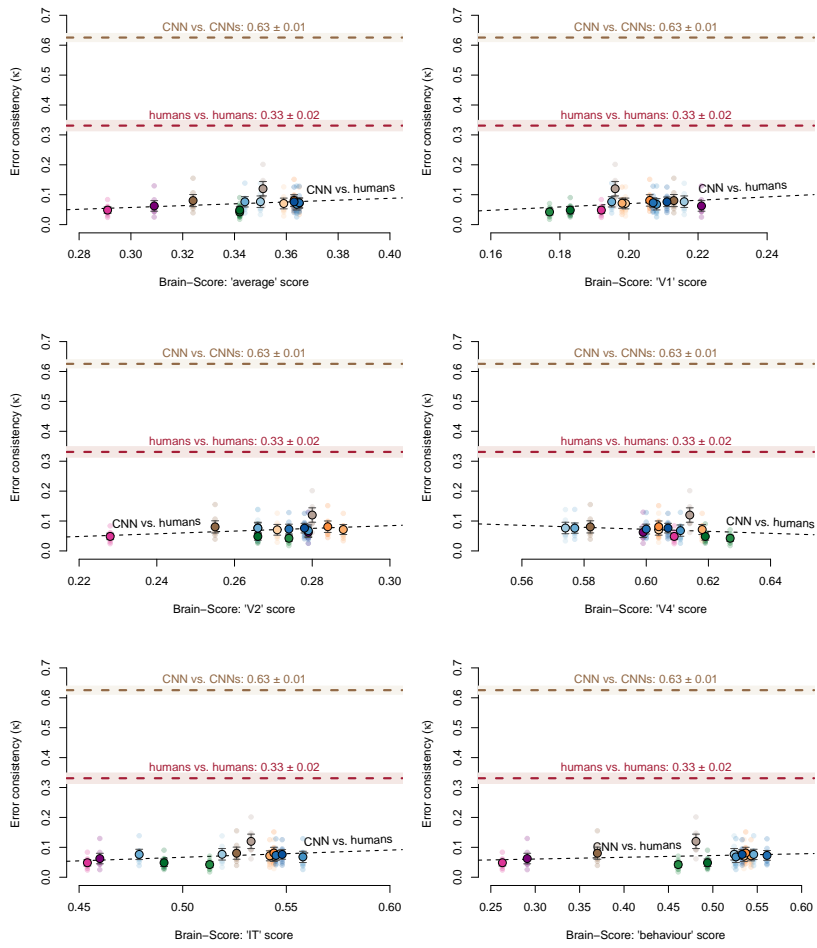


Figure SF9: Error consistency vs. Brain-Score metrics for PyTorch models, “cue conflict” stimuli.

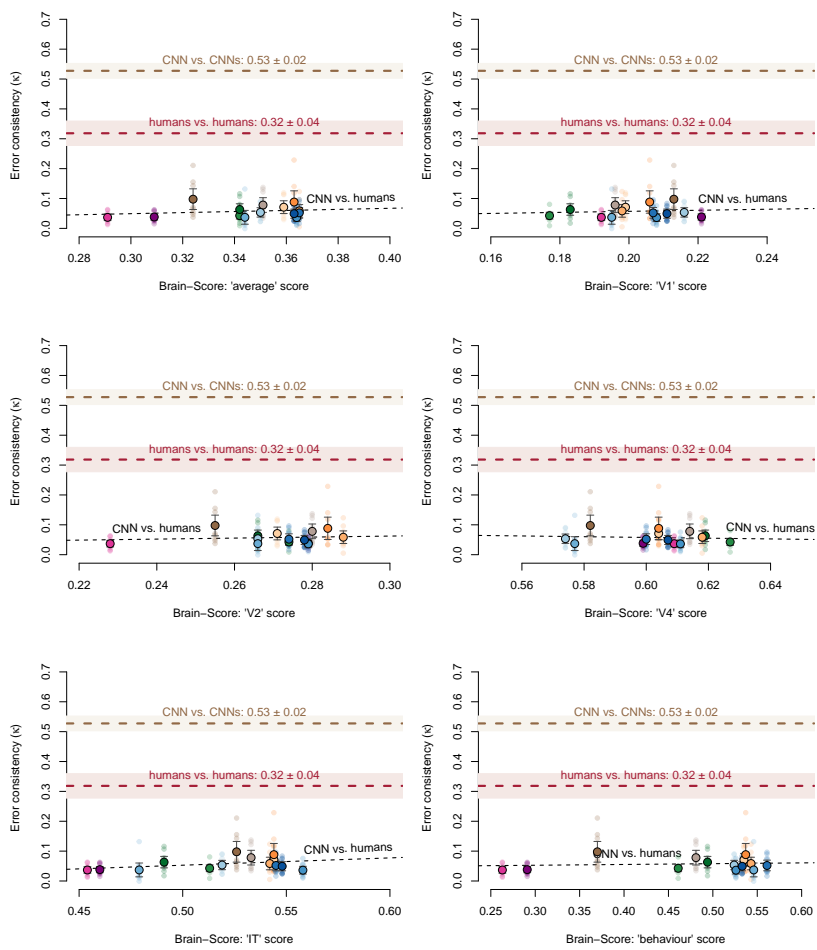


Figure SF.10: Error consistency vs. Brain-Score metrics for PyTorch models, “edge” stimuli.

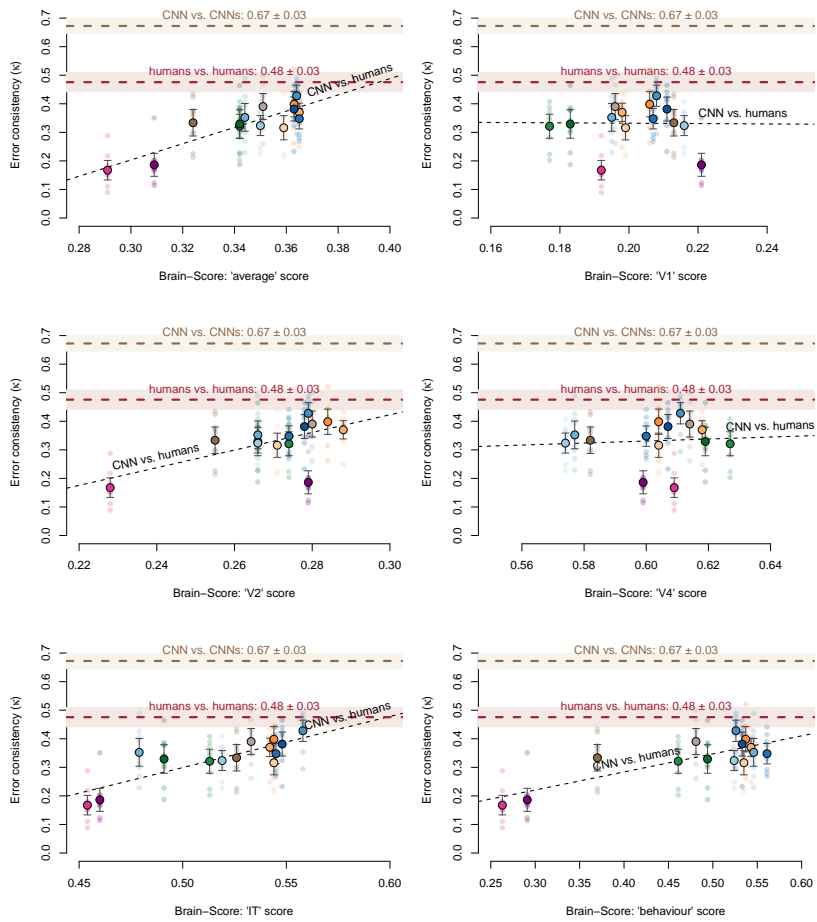


Figure SF.11: Error consistency vs. Brain-Score metrics for PyTorch models, “silhouette” stimuli.

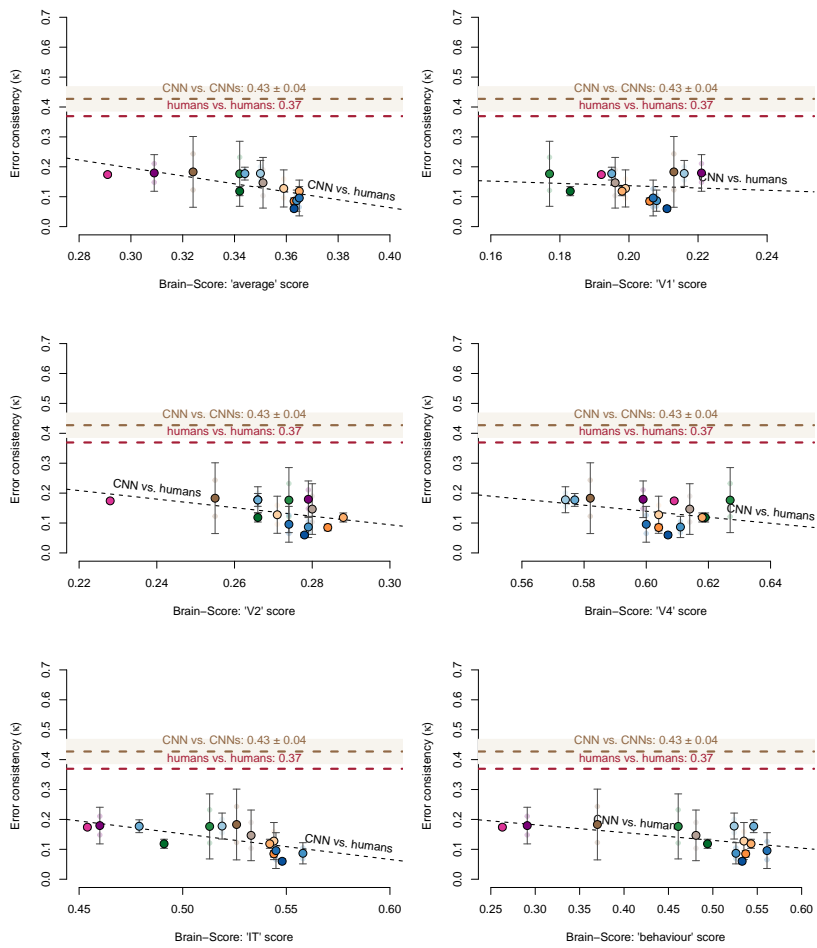


Figure SF.12: Error consistency vs. Brain-Score metrics for PyTorch models, “ImageNet” stimuli.

## 2.4 On the surprising similarities between supervised and self-supervised models

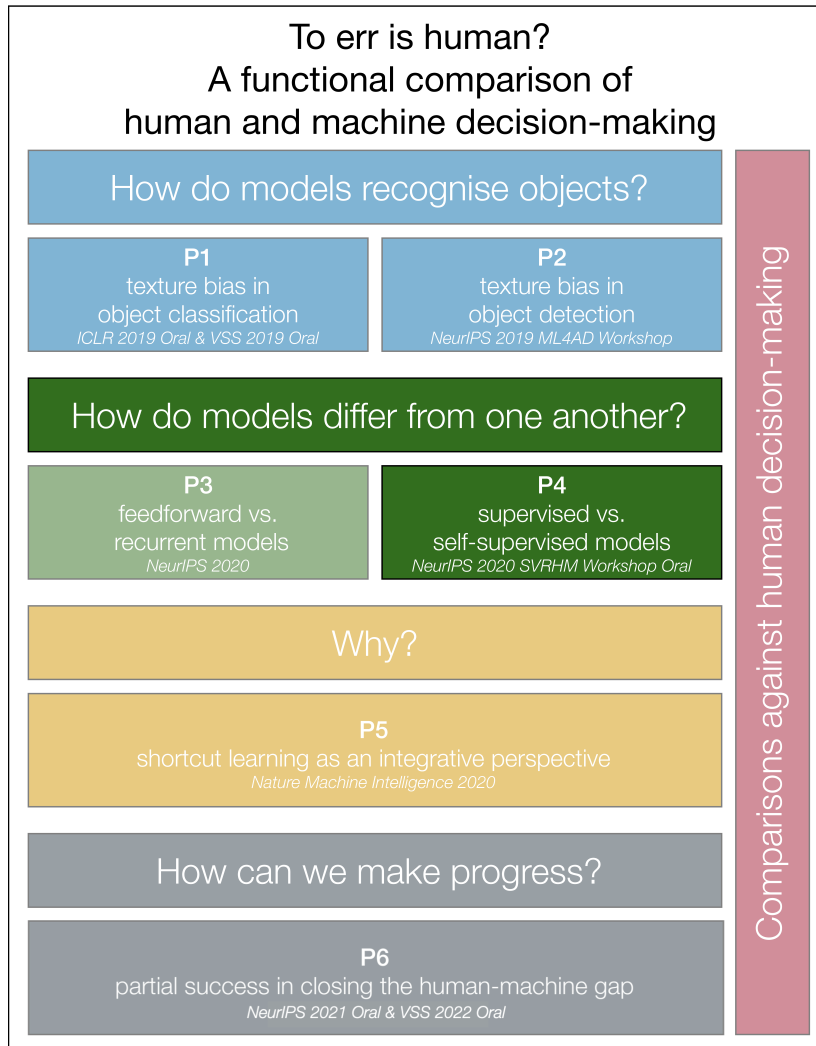


Figure 2.4: Schematic overview of the projects presented in this thesis. Projects P1–P4 ask “How do machines decide”, project P5 presents the concept of shortcut learning as an integrative perspective tackling the question “Why do machines decide the way they do”. Finally, project P6 presents a comprehensive benchmark to measure future progress, and reports first signs of (partial) success. Throughout the thesis, machine decision making will be compared against human decision making.



---

# On the surprising similarities between supervised and self-supervised models

---

**Robert Geirhos**<sup>§</sup>      **Kantharaju Narayanappa**      **Benjamin Mitzkus**  
 University of Tübingen & IMPRS-IS      University of Tübingen      University of Tübingen

**Matthias Bethge**\*      **Felix A. Wichmann**\*      **Wieland Brendel**\*  
 University of Tübingen      University of Tübingen      University of Tübingen

\*Joint senior authors

<sup>§</sup>robert.geirhos@uni-tuebingen.de

## Abstract

How do humans learn to acquire a powerful, flexible and robust representation of objects? While much of this process remains unknown, it is clear that humans do not require millions of object labels. Excitingly, recent algorithmic advancements in self-supervised learning now enable convolutional neural networks (CNNs) to learn useful visual object representations without supervised labels, too. In the light of this recent breakthrough, we here compare self-supervised networks to supervised models and human behaviour.

We tested models on 15 generalisation datasets for which large-scale human behavioural data is available (130K highly controlled psychophysical trials). Surprisingly, current self-supervised CNNs share four key characteristics of their supervised counterparts: (1.) relatively poor noise robustness (with the notable exception of SimCLR), (2.) non-human category-level error patterns, (3.) non-human image-level error patterns (yet high similarity to supervised model errors) and (4.) a bias towards texture. Taken together, these results suggest that the strategies learned through today’s supervised and self-supervised training objectives end up being surprisingly similar, but distant from human-like behaviour. That being said, we are clearly just at the beginning of what could be called a self-supervised revolution of machine vision, and we are hopeful that future self-supervised models behave differently from supervised ones, and—perhaps—more similar to robust human object recognition.

## 1 Introduction

*“If intelligence is a cake, the bulk of the cake is unsupervised learning, the icing on the cake is supervised learning and the cherry on the cake is reinforcement learning”, Yann LeCun famously said [1]. Four years later, the entire cake is finally on the table—the representations learned via self-supervised learning now compete with supervised methods on ImageNet [2] and outperform supervised pre-training for object detection [3]. But given this fundamentally different learning mechanism, how do recent self-supervised models differ from their supervised counterparts in terms of their behaviour?*

We here attempt to shed light on this question by comparing eight flavours of “cake” (PIRL, MoCo, MoCoV2, InfoMin, InsDis, SimCLR-x1, SimCLR-x2, SimCLR-x4) with 24 common variants of

2nd Workshop on Shared Visual Representations in Human and Machine Intelligence (SVRHM), NeurIPS 2020.

“icing” (from the AlexNet, VGG, Squeezenet, DenseNet, Inception, ResNet, ShuffleNet, MobileNet, ResNeXt, WideResNet and MNASNet star cuisines). Specifically, our culinary test buffet aims to investigate:

1. Are self-supervised models more robust towards distortions?
2. Do self-supervised models make similar errors as either humans or supervised models?
3. Do self-supervised models recognise objects by texture or shape?

For all of these questions, we compare supervised and self-supervised<sup>1</sup> models against a comprehensive set of openly available human psychophysical data totalling over 130,000 trials [4, 5]. This is motivated on one hand by the fact that humans, too, rapidly learn to recognise new objects without requiring hundreds of labels per instance; and on the other hand by a number of fascinating studies reporting increased similarities between self-supervised models and human perception. For instance, Lotter et al. [6] train a model for self-supervised next frame prediction on videos, which leads to phenomena known from human vision, including perceptual illusions. Orhan et al. [7] train a self-supervised model on infant video data, finding good categorisation accuracies on some (small) datasets. Furthermore, Konkle and Alvarez [8] and Zhuang et al. [9] report an improved match with neural data; Zhuang et al. [9] also find more human-like error patterns for semi-supervised models and Storrs and Fleming [10] observe that a self-supervised network accounts for behavioural patterns of human gloss perception. While methods and models differ substantially across these studies, they jointly provide evidence for the intriguing hypothesis that self-supervised machine learning models may better approximate human vision.

## 2 Methods

**Models.** InsDis [11], MoCo [12], MoCoV2 [13], PIRL [3] and InfoMin [14] were obtained as pre-trained models from the PyContrast model zoo. We trained one linear classifier per model on top of the self-supervised representation. A PyTorch [15] implementation of SimCLR [2] was obtained via simclr-converter. All self-supervised models use a ResNet-50 architecture and a different training approach within the framework of contrastive learning [e.g. 16]. For supervised models, we used all 24 available pre-trained models from the PyTorch model zoo version 1.4.0 (VGG: with batch norm).

**Linear classifier training procedure.** The PyContrast repository by Yonglong Tian contains a Pytorch implementation of unsupervised representation learning methods, including pre-trained representation weights. The repository provides training and evaluation pipelines, but it supports only multi-node distributed training and does not (currently) provide weights for the classifier. We have used the repository’s linear classifier evaluation pipeline to train classifiers for InsDis [11], MoCo [12], MoCoV2 [13], PIRL [3] and InfoMin [14] on ImageNet. Pre-trained weights of the model representations (without classifier) were taken from the provided Dropbox link and we then ran the training pipeline on a NVIDIA TESLA P100 using the default parameters configured in the pipeline. Detailed documentation about running the pipeline and parameters can be found in the PyContrast repository (commit #3541b82).

**Datasets.** Models were tested on 12 different image degradations from [4], as well as on texture-vs-shape datasets from [5]. Plotting conventions follow these papers (unless indicated otherwise).

## 3 Results

We here investigate four behavioural characteristics of self-supervised networks, comparing them to their supervised counterparts on the one hand and to human observers on the other hand: out-of-distribution generalisation (3.1), category-level error patterns (3.2), image-level error patterns (3.3), and texture/shape biases (3.4).

<sup>1</sup>“Unsupervised learning” and “self-supervised learning” are sometimes used interchangeably. We use the term “self-supervised learning” since the methods use (label-free) supervision.

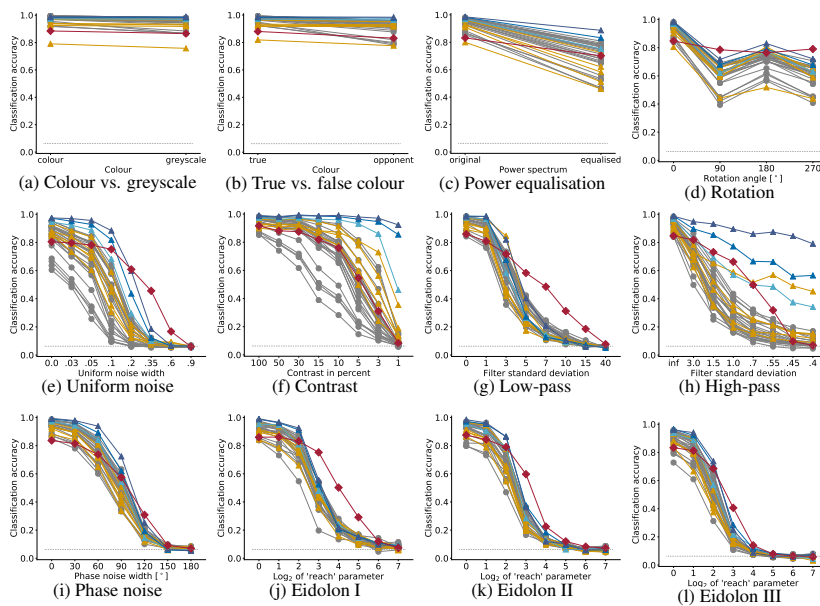


Figure 1: Noise generalisation results for humans (red diamonds) vs. supervised models (grey circles) vs. self-supervised models (orange triangles). Self-supervised SimCLR variants: blue triangles.

### 3.1 With the exception of SimCLR, supervised and self-supervised models show similar (non-human) out-of-distribution generalisation

**Motivation.** Given sufficient quantities of labelled training data, CNNs can learn to identify objects when the input images are noisy. However, supervised CNNs typically generalise poorly to novel distortion types not seen during training, so-called out-of-distribution images [4]. In contrast, human perception is remarkably robust when dealing with previously unseen types of noise. Given that recent self-supervised networks are trained to identify objects under a variety of transformations (like scaling, cropping and colour shifts), have they learned a more robust, human-like representation of objects, where high-level semantic content is unimpaired by low-level noise?

**Results.** In Figure 1, we compare self-supervised and supervised networks on twelve different types of image distortions. Human observers were tested on the exact same distortions by [4]. Across distortion types, self-supervised networks are well within the range of their poorly generalising supervised counterparts. However, there is one exception: SimCLR shows strong generalisation improvements on uniform noise, low contrast and high-pass images—quite remarkable given that the network was trained using other augmentations (random crop with flip and resize, colour distortion, and Gaussian blur). Apart from SimCLR, however, we do not find benefits of self-supervised training for distortion robustness. These results for ImageNet models contrast with [17] who observed some robustness improvements for a self-supervised model trained on the CIFAR-10 dataset.

### 3.2 Self-supervised models make non-human category-level errors

**Motivation.** On clean images, CNNs now recognise objects as well as humans. But do they also confuse similar categories with each other (which can be investigated using confusion matrices)?

**Results.** In Figure 4 (moved to appendix for space reasons), we compare category-level errors of humans against a standard supervised CNN (ResNet-50) and three self-supervised CNNs. We chose

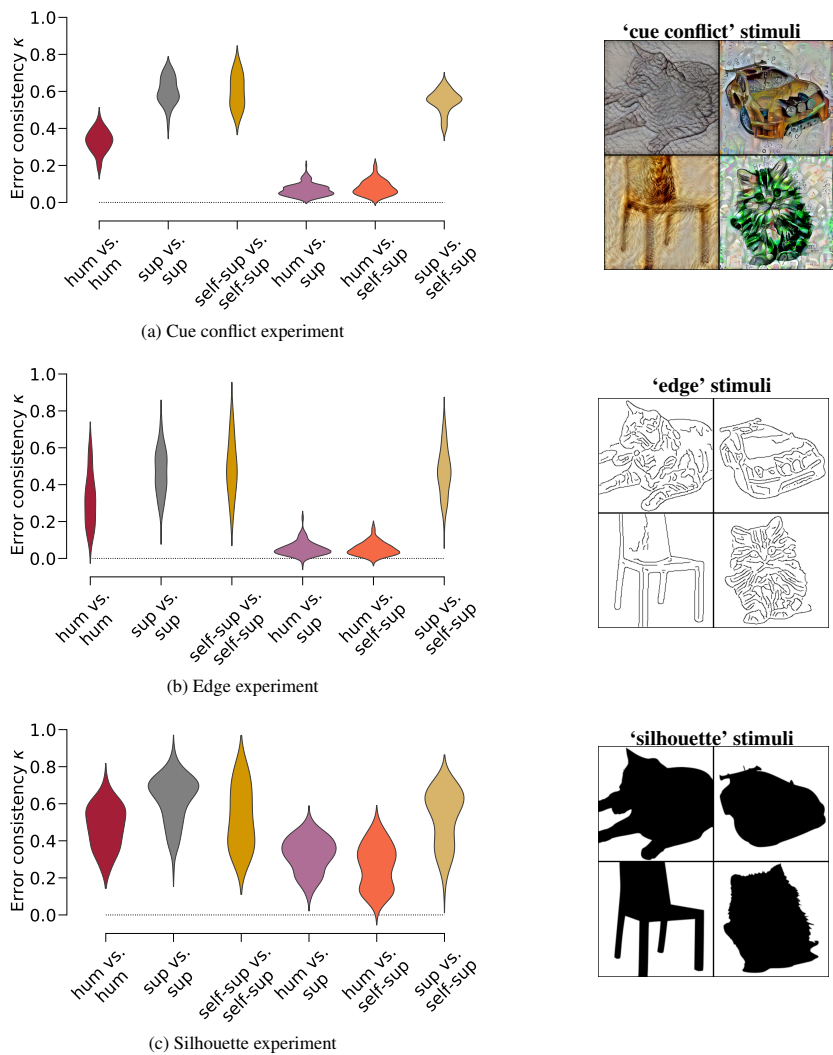


Figure 2: Self-supervised models make errors on the same images as supervised models. Error consistency (high  $\kappa$  = consistent errors) between all combinations of the following three groups: humans (hum), supervised networks (sup) and self-supervised networks (self-sup). Stimuli from [5]: (a) cue conflict, (b) edges and (c) silhouettes (visualisation by [18]). For all three experiments, consistency between networks is much higher than between networks and humans: CNNs make errors on the same images as other CNNs, whether these are supervised or self-supervised.

uniform noise for this comparison since this is one of the noise types where SimCLR shows strong improvements, nearing human-level accuracies. Looking at the confusion matrices, both humans and CNNs start with a dominant diagonal indicating correct categorisation. With increasing noise level, however, all CNNs develop a strong tendency to predict one category and only one, such as “knife” for ResNet-50. Human observers, on the other hand, more or less evenly distribute their errors across classes. For supervised networks, this pattern of errors was already observed by [4]; we here find that this peculiar idiosyncrasy is shared by self-supervised networks, indicating that discriminative supervised training is not the underlying reason for this non-human behaviour.

### 3.3 Self-supervised models make non-human image-level errors (error consistency)

**Motivation.** Achieving human-level accuracies on a dataset does not necessarily imply using a human-like strategy: different strategies can lead to similar accuracies. Therefore, it is essential to investigate *on which stimuli* errors occur. If two decision makers—for instance, a human observer and a CNN—use a similar strategy, we can expect them to consistently make errors on the same individual images. This intuition is captured by *error consistency* ( $\kappa$ ), a metric to measure the degree to which decision makers make the same image-level errors [18].  $\kappa > 0$  means that two decision makers systematically make errors on the same images;  $\kappa = 0$  indicates no more error overlap than what could be expected by chance alone.

**Results.** Figure 2 plots the consistency of errors (measured by  $\kappa$ ). Humans make highly similar errors as other humans (mean  $\kappa = 0.32$ ), but neither supervised nor self-supervised models make human-like errors. Instead, error consistency *between* model groups (self-supervised vs. supervised) is just as high as consistency *within* model groups: self-supervised models make errors on the same images as supervised models, an indicator for highly similar strategies.

### 3.4 Self-supervised models are biased towards texture

**Motivation.** Standard supervised networks recognise objects by relying on local texture statistics, largely ignoring global object shape [5, 19, 20]. This striking difference to human visual perception has been attributed to the fact that texture is a shortcut sufficient to discriminate among objects [21]—but is texture also sufficient to solve self-supervised training objectives?

**Results.** We tested a broad range of CNNs on the texture-shape cue conflict dataset from [5]. This dataset consists of images where the shape belongs to one category (e.g. cat) and the texture belongs to a different category (e.g. elephant). When plotting whether CNNs prefer texture or shape (Figure 3), we observe that most self-supervised models have a strong texture bias known from traditional supervised models. This texture bias is less prominent for SimCLR (58.3–61.2% texture decisions), which is still on par with supervised model Inception-V3 (60.7% texture decisions). These findings are in line with [22], who observed that the influence of training data augmentations on shape bias is stronger than the role of architecture or training objective. Neither supervised nor self-supervised models have the strong shape bias that is so characteristic for human observers, indicating fundamentally different decision making processes between humans and CNNs.

## 4 Discussion

Comparing self-supervised networks to supervised models and human observers, we here investigated four key behavioural characteristics: out-of-distribution generalisation, category-level error patterns, image-level error patterns, and texture bias. Overall, we find that self-supervised models resemble their supervised counterparts much more closely than what could have been expected given fundamentally different training objectives. While standard models are notoriously non-robust [4, 21, 23–25], SimCLR represents a notable exception in some of our experiments as it is less biased towards texture and much more robust towards some types of distortions. It is an open question whether these benefits arise from the specific set of data augmentations used during SimCLR model training.

Perhaps surprisingly, error consistency analysis suggests that the images on which supervised and self-supervised models make errors overlap strongly, much more than what could have been expected by chance alone. This provides evidence for similar processing mechanisms: It seems that switching label-based supervision for a contrastive learning scheme does not have a strong effect on the inductive

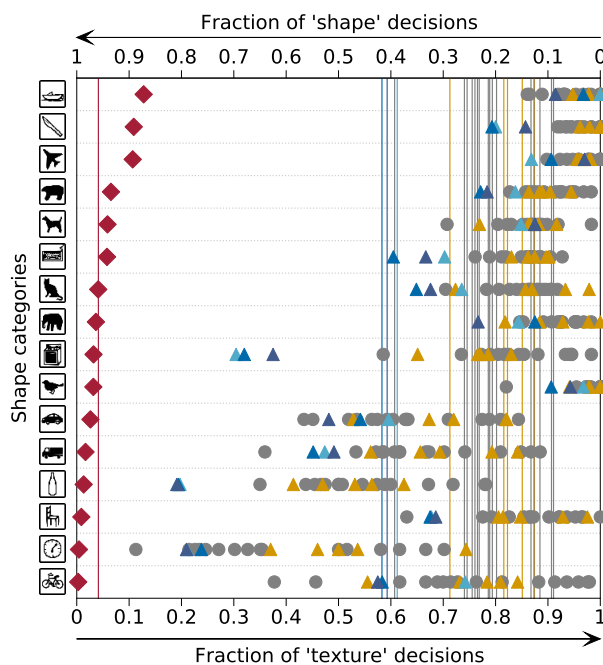


Figure 3: Self-supervised models are biased towards texture. Vertical lines indicate the average across categories for a certain model. **Humans:** red diamonds, supervised models: grey circles, **self-supervised models:** orange triangles, self-supervised SimCLR variants: blue triangles.

bias of the resulting model—at least for the currently used contrastive approaches. Furthermore, we find little evidence for human-like behaviour in the investigated self-supervised models. While this investigation focused on state-of-the-art contrastive learning methods, other self-supervised methods might lead to different results.

We are clearly just witnessing the beginning of what could be called a self-supervised revolution of machine vision, and we expect future self-supervised models to behave significantly differently from supervised ones. What we are showing is that the current self-supervised CNNs are not yet more human-like in their strategies and internal representations than plain-vanilla supervised CNNs. We hope, however, that analyses like ours may facilitate the tracking of emerging similarities and differences, whether between different types of models or between models and human perception.

#### Acknowledgement

We thank Santiago Cadena for sharing a PyTorch implementation of SimCLR, and Yonglong Tian for providing pre-trained self-supervised models on github. Furthermore, we are grateful to the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting R.G.; the Collaborative Research Center (Projektnummer 276693517—SFB 1233: Robust Vision) for supporting M.B. and F.A.W.; the German Federal Ministry of Education and Research through the Tübingen AI Center (FKZ 01IS18039A) for supporting W.B. and M.B.; and the German Research Foundation through the Cluster of Excellence “Machine Learning—New Perspectives for Science”, EXC 2064/1, project number 390727645 for supporting F.A.W.

#### Author contributions

Project idea: R.G. and W.B.; project lead: R.G.; implementing and training self-supervised models: K.N.; model evaluation pipeline: R.G., K.N. with input from W.B.; data visualisation: R.G. and B.M. with input from M.B., F.A.W. and W.B.; guidance, feedback, infrastructure & funding acquisition: M.B., F.A.W. and W.B.; paper writing: R.G. with input from all other authors.

## References

- [1] Yann LeCun. Predictive learning, 2016. URL <https://www.youtube.com/watch?v=0unt2Y4qxQo>.
- [2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.
- [3] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6707–6717, 2020.
- [4] Robert Geirhos, Carlos RM Temme, Jonas Rauber, Heiko H Schütt, Matthias Bethge, and Felix A Wichmann. Generalisation in humans and deep neural networks. In *Advances in Neural Information Processing Systems*, 2018.
- [5] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2019.
- [6] William Lotter, Gabriel Kreiman, and David Cox. A neural network trained for prediction mimics diverse features of biological neurons and perception. *Nature Machine Intelligence*, 2(4):210–219, 2020.
- [7] A Emin Orhan, Vaibhav V Gupta, and Brenden M Lake. Self-supervised learning through the eyes of a child. *arXiv preprint arXiv:2007.16189*, 2020.
- [8] Talia Konkle and George A Alvarez. Instance-level contrastive learning yields human brain-like representation without category-supervision. *bioRxiv*, 2020.
- [9] Chengxu Zhuang, Siming Yan, Aran Nayebi, Martin Schrimpf, Michael Frank, James DiCarlo, and Daniel Yamins. Unsupervised neural network models of the ventral visual stream. *bioRxiv*, 2020.
- [10] Katherine R Storrs and Roland W Fleming. Unsupervised learning predicts human perception and misperception of specular surface reflectance. *bioRxiv*, 2020.
- [11] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3733–3742, 2018.
- [12] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.
- [13] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020.
- [14] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning. *arXiv preprint arXiv:2005.10243*, 2020.
- [15] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8026–8037, 2019.
- [16] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv:1807.03748*, 2018.
- [17] Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. Using self-supervised learning can improve model robustness and uncertainty. In *Advances in Neural Information Processing Systems*, pages 15663–15674, 2019.

- [18] Robert Geirhos, Kristof Meding, and Felix A Wichmann. Beyond accuracy: quantifying trial-by-trial behaviour of CNNs and humans by measuring error consistency. *Advances in Neural Information Processing Systems*, 33, 2020.
- [19] Nicholas Baker, Hongjing Lu, Gennady Erlikhman, and Philip J Kellman. Deep convolutional networks do not classify based on global object shape. *PLoS Computational Biology*, 14(12): e1006613, 2018.
- [20] Wieland Brendel and Matthias Bethge. Approximating CNNs with bag-of-local-features models works surprisingly well on ImageNet. In *International Conference on Learning Representations*, 2019.
- [21] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2:665–673, 2020.
- [22] Katherine Hermann, Ting Chen, and Simon Kornblith. The origins and prevalence of texture bias in convolutional neural networks. *Advances in Neural Information Processing Systems*, 33, 2020.
- [23] Samuel Dodge and Lina Karam. Understanding how image quality affects deep neural networks. *International Conference on Quality of Multimedia Experience*, 2016.
- [24] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019.
- [25] Arturo Deza and Talia Konkle. Emergent properties of foveated perceptual systems. *arXiv preprint arXiv:2006.07991*, 2020.

## Appendix

Figure 4 shows confusion matrices for uniform noise, Figure 5 for low-pass filtering.



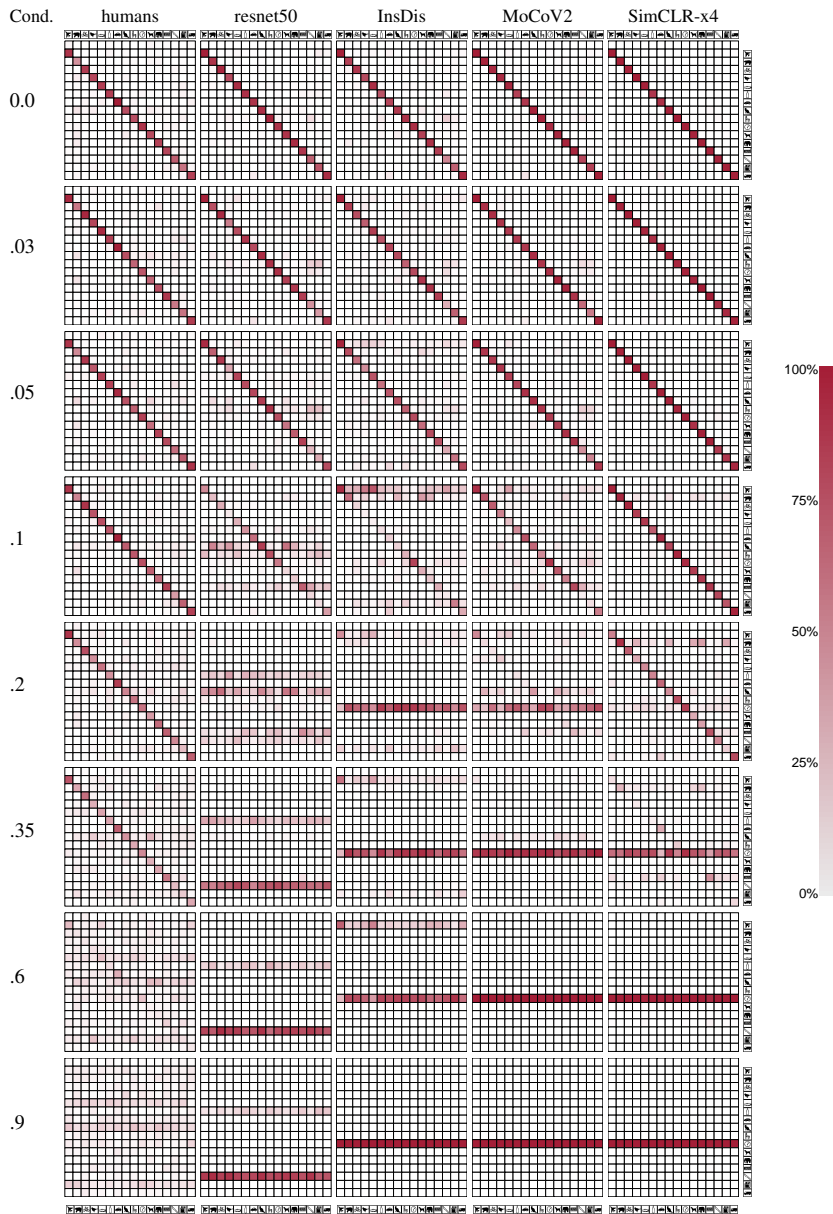


Figure 4: Confusion matrices for different conditions (“Cond.”) of the uniform noise experiment. Columns show ground truth object categories, rows indicate predicted categories. Supervised ResNet-50 and self-supervised networks InsDis, MoCoV2 & SimCLR-x4 all preferentially select a single category with increasing noise level.

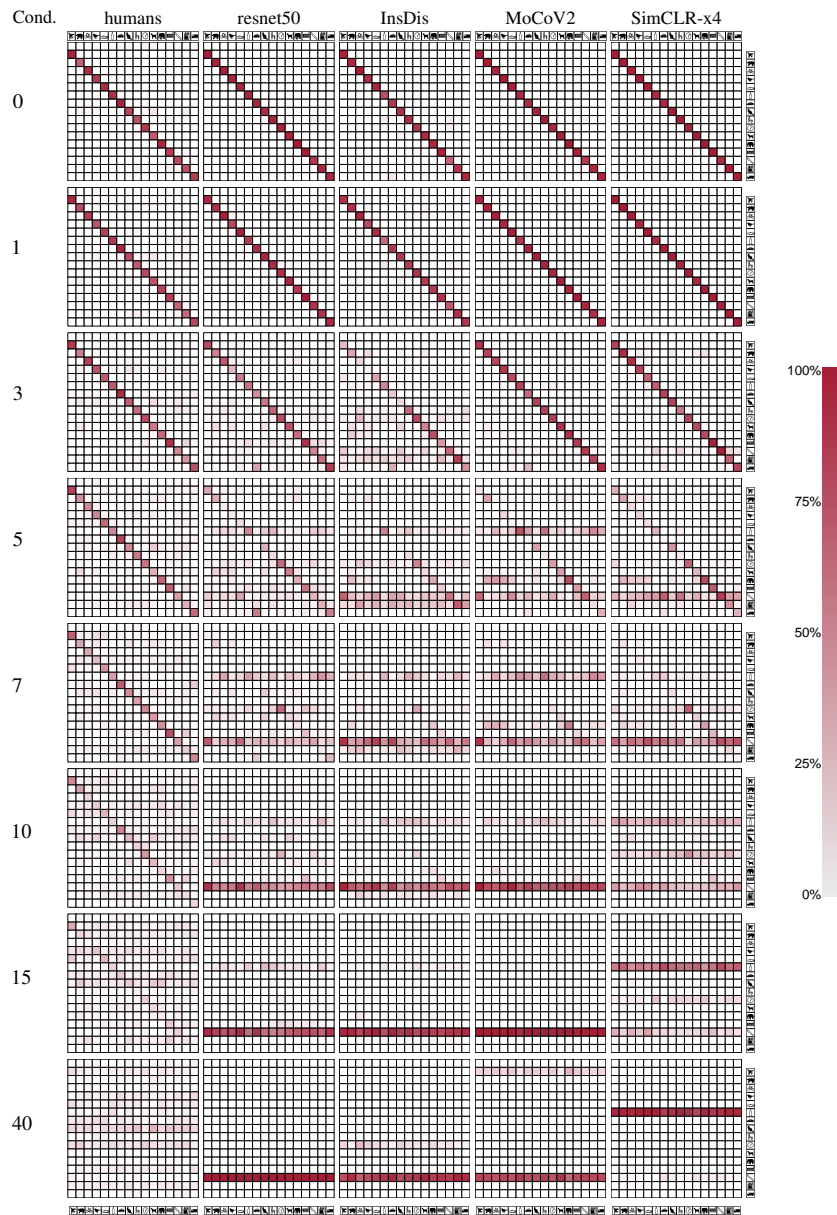


Figure 5: Confusion matrices for low-pass filtering. Again, CNNs develop a preference for a certain category as the distortion strength increases.

## 2.5 Shortcut Learning in Deep Neural Networks

*Publication & version notice* First published in Nature Machine Intelligence, volume 2(11), pages 665–673, 2020 by Springer Nature. The final published version is available from <https://doi.org/10.1038/s42256-020-00257-z>. Due to the publisher’s copyright assignment, reprinting it in the final formatted and published version is not possible; therefore, the preprint version ([arXiv version v4](#)) is included here.

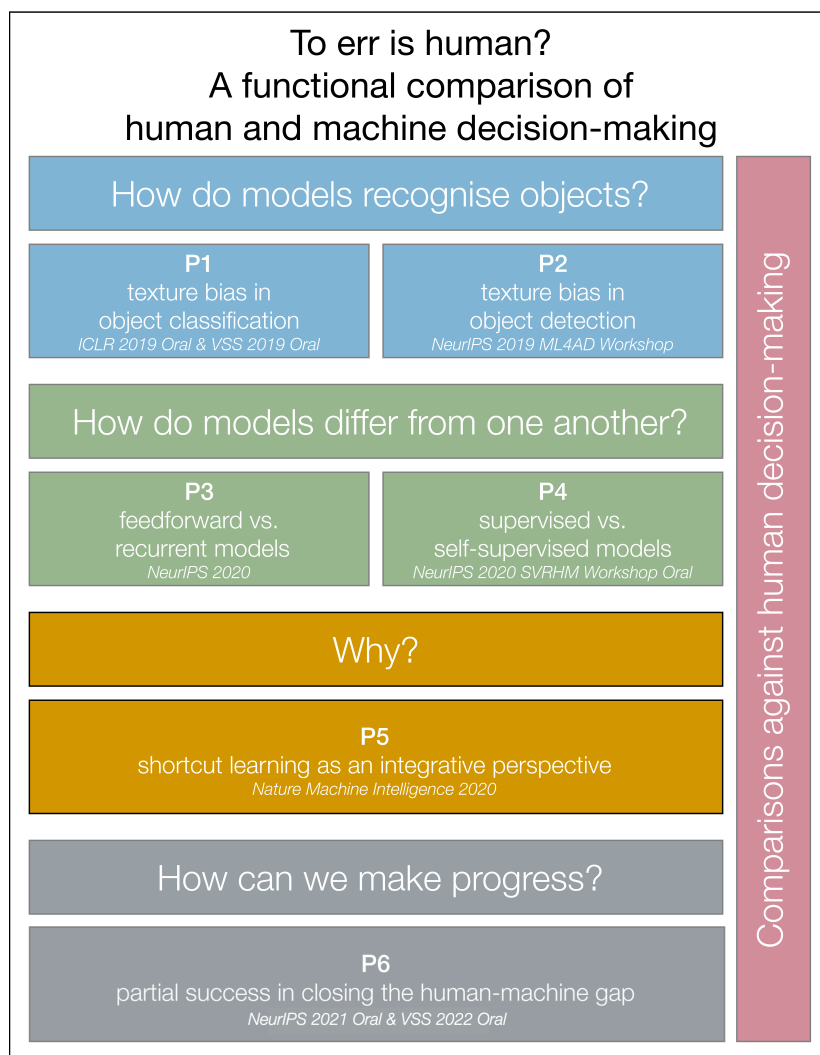


Figure 2.5: Schematic overview of the projects presented in this thesis. Projects P1–P4 ask “How do machines decide”, project P5 presents the concept of shortcut learning as an integrative perspective tackling the question “Why do machines decide the way they do”. Finally, project P6 presents a comprehensive benchmark to measure future progress, and reports first signs of (partial) success. Throughout the thesis, machine decision making will be compared against human decision making.

# Shortcut Learning in Deep Neural Networks

Robert Geirhos<sup>1,2,\*</sup>, Jörn-Henrik Jacobsen<sup>3,\*</sup>, Claudio Michaelis<sup>1,2,\*</sup>,  
Richard Zemel<sup>†,3</sup>, Wieland Brendel<sup>†,1</sup>, Matthias Bethge<sup>†,1</sup> & Felix A. Wichmann<sup>†,1</sup>

<sup>1</sup>University of Tübingen, Germany

<sup>2</sup>International Max Planck Research School for Intelligent Systems, Germany

<sup>3</sup>University of Toronto, Vector Institute, Canada

\*Joint first / † joint senior authors

<sup>§</sup>To whom correspondence should be addressed: robert.geirhos@w Wichmannlab.org

## Abstract

Deep learning has triggered the current rise of artificial intelligence and is the workhorse of today’s machine intelligence. Numerous success stories have rapidly spread all over science, industry and society, but its limitations have only recently come into focus. In this perspective we seek to distil how many of deep learning’s problem can be seen as different symptoms of the same underlying problem: *shortcut learning*. Shortcuts are decision rules that perform well on standard benchmarks but fail to transfer to more challenging testing conditions, such as real-world scenarios. Related issues are known in Comparative Psychology, Education and Linguistics, suggesting that shortcut learning may be a common characteristic of learning systems, biological and artificial alike. Based on these observations, we develop a set of recommendations for model interpretation and benchmarking, highlighting recent advances in machine learning to improve robustness and transferability from the lab to real-world applications.

## 1 Introduction

If science was a journey, then its destination would be the discovery of simple explanations to complex phenomena. There was a time when the existence of tides, the planet’s orbit around the sun, and the observation that “things fall down” were all largely considered to be independent phenomena—until 1687, when Isaac Newton formulated his law of gravitation that provided an elegantly simple explanation to all of these (and many more). Physics has made tremendous progress over the last few centuries, but the thriving field of deep learning is still very much at the beginning of its journey—often lacking a detailed understanding of the underlying principles.

For some time, the tremendous success of deep learning has perhaps overshadowed the need to thoroughly understand the behaviour of Deep Neural Networks (DNNs). In an ever-increasing pace, DNNs were reported as having achieved human-level object classification performance [1], beating world-class human Go, Poker, and Starcraft players [2, 3],

---

This is the preprint version of an article that has been published by Nature Machine Intelligence (<https://doi.org/10.1038/s42256-020-00257-z>).

Task for DNN	Caption image	Recognise object	Recognise pneumonia	Answer question
Problem	Describes green hillsides as grazing sheep	Hallucinates teapot if certain patterns are present	Fails on scans from new hospitals	Changes answer if irrelevant information is added
Shortcut	Uses background to recognise primary object	Uses features irrecognisable to humans	Looks at hospital token, not lung	Only looks at last sentence and ignores context

**Figure 1.** Deep neural networks often solve problems by taking shortcuts instead of learning the intended solution, leading to a lack of generalisation and unintuitive failures. This pattern can be observed in many real-world applications.

detecting cancer from X-ray scans [4], translating text across languages [5], helping combat climate change [6], and accelerating the pace of scientific progress itself [7]. Because of these successes, deep learning has gained a strong influence on our lives and society. At the same time, however, researchers are unsatisfied about the lack of a deeper understanding of the underlying principles and limitations. Different from the past, tackling this lack of understanding is not a purely scientific endeavour anymore but has become an urgent necessity due to the growing societal impact of machine learning applications. If we are to trust algorithms with our lives by being driven in an autonomous vehicle, if our job applications are to be evaluated by neural networks, if our cancer screening results are to be assessed with the help of deep learning—then we indeed need to understand thoroughly: When does deep learning work? When does it fail, and why?

In terms of understanding the limitations of deep learning, we are currently observing a large number of failure cases, some of which are visualised in Figure 1. DNNs achieve super-human performance recognising objects, but even small invisible changes [8] or a different background context [9, 10] can completely derail predictions. DNNs can generate a plausible caption for an image, but—worryingly—they can do so without ever looking at that image [11]. DNNs can accurately recognise faces, but they show high error rates for faces from minority groups [12]. DNNs can predict hiring decisions on the basis of résumés, but the algorithm’s decisions are biased towards selecting men [13].

How can this discrepancy between super-human performance on one hand and astonishing failures on the other hand be reconciled? One central observation is that many failure cases are not independent phenomena, but are instead connected in the sense that DNNs follow unintended “shortcut” strategies. While superficially successful, these strategies typically fail under slightly different circumstances. For instance, a DNN may appear to classify cows perfectly well—but fails when tested on pictures where cows appear outside the typical grass landscape, revealing “grass” as an unintended (shortcut) predictor for “cow” [9]. Likewise, a language model may appear to have learned to reason—but drops to chance performance when superficial correlations are removed from the dataset [14]. Worse yet, a machine classifier successfully detected pneumonia from X-ray scans of a number of hospitals, but its performance was surprisingly low for scans from novel hospitals: The model had unexpectedly learned to identify particular hospital systems with near-perfect accuracy (e.g. by detecting a hospital-specific metal token on the scan, see Figure 1). Together with the hospital’s pneumonia prevalence rate it was able to achieve a

reasonably good prediction—without learning much about pneumonia at all [15].

At a principal level, shortcut learning is not a novel phenomenon. The field of machine learning with its strong mathematical underpinnings has long aspired to develop a formal understanding of shortcut learning which has led to a variety of mathematical concepts and an increasing amount of work under different terms such as *learning under covariate shift* [16], *anti-causal learning* [17], *dataset bias* [18], the *tank legend* [19] and the *Clever Hans effect* [20]. This perspective aims to present a unifying view of the various phenomena that can be collectively termed shortcuts, to describe common themes underlying them, and lay out the approaches that are being taken to address them both in theory and in practice.

The structure of this perspective is as follows. Starting from an intuitive level, we introduce shortcut learning across biological neural networks (Section 2) and then approach a more systematic level by introducing a taxonomy (Section 3) and by investigating the origins of shortcuts (Section 4). In Section 5, we highlight how these characteristics affect different areas of deep learning (Computer Vision, Natural Language Processing, Agent-based Learning, Fairness). The remainder of this perspective identifies actionable strategies towards diagnosing and understanding shortcut learning (Section 6) as well as current research directions attempting to overcome shortcut learning (Section 7). Overall, our selection of examples is biased towards Computer Vision since this is one of the areas where deep learning has had its biggest successes, and an area where examples are particularly easy to visualise. We hope that this perspective facilitates the awareness for shortcut learning and motivates new research to tackle this fundamental challenge we currently face in machine learning.

## 2 Shortcut learning in biological neural networks

Shortcut learning typically reveals itself by a strong discrepancy between intended and actual learning strategy, causing an unexpected failure. Interestingly, machine learning is not alone with this issue: From the way students learn to the unintended strategies rats use in behavioural experiments—variants of shortcut learning are also common for biological neural networks. We here point out two examples of unintended learning strategies by natural systems in the hope that this may provide an interesting frame of reference for thinking about shortcut learning within and beyond artificial systems.

### 2.1 Shortcut learning in Comparative Psychology: unintended cue learning

*Rats learned to navigate a complex maze apparently based on subtle colour differences—very surprising given that the rat retina has only rudimentary machinery to support at best somewhat crude colour vision. Intensive investigation into this curious finding revealed that the rats had tricked the researchers: They did not use their visual system at all in the experiment and instead simply discriminated the colours by the odour of the colour paint used on the walls of the maze. Once smell was controlled for, the remarkable colour discrimination ability disappeared ...<sup>1</sup>*

Animals are no strangers to finding simple, unintended solutions that fail unexpectedly: They are prone to *unintended cue learning*, as shortcut learning is called in Comparative

<sup>1</sup>Nicholas Rawlins, personal communication with F.A.W. some time in the early 1990s, confirmed via email on 12.11.2019.

Psychology and the Behavioural Neurosciences. When discovering cases of unintended cue learning, one typically has to acknowledge that there was a crucial difference between performance in a given experimental paradigm (e.g. rewarding rats to identify different colours) and the investigated mental ability one is actually interested in (e.g. visual colour discrimination). In analogy to machine learning, we have a striking discrepancy between intended and actual learning outcome.

## 2.2 Shortcut learning in Education: surface learning

*Alice loves history. Always has, probably always will. At this very moment, however, she is cursing the subject: After spending weeks immersing herself in the world of Hannibal and his exploits in the Roman Empire, she is now faced with a number of exam questions that are (in her opinion) to equal parts dull and difficult. “How many elephants did Hannibal employ in his army—19, 34 or 40?” ... Alice notices that Bob, sitting in front of her, seems to be doing very well. Bob of all people, who had just boasted how he had learned the whole book chapter by rote last night ...*

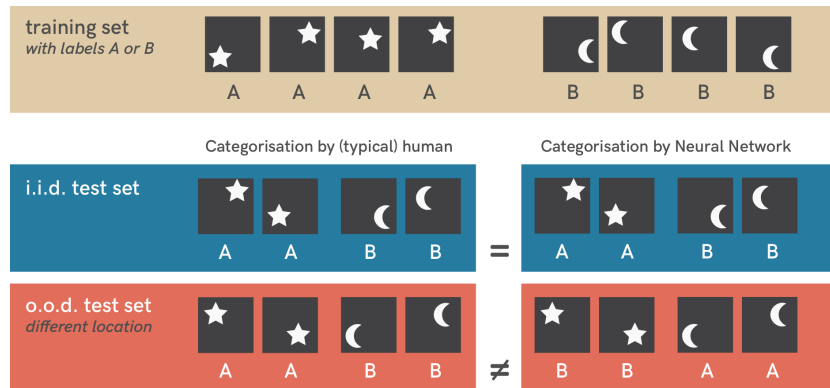
In educational research, Bob’s reproductive learning strategy would be considered *surface learning*, an approach that relies on narrow testing conditions where simple discriminative generalisation strategies can be highly successful. This fulfils the characteristics of shortcut learning by giving the appearance of good performance but failing immediately under more general test settings. Worryingly, surface learning helps rather than hurts test performance on typical multiple-choice exams [21]: Bob is likely to receive a good grade, and judging from grades alone Bob would appear to be a much better student than Alice in spite of her focus on understanding. Thus, in analogy to machine learning we again have a striking discrepancy between intended and actual learning outcome.

## 3 Shortcuts defined: a taxonomy of decision rules

With examples of biological shortcut learning in mind (examples which we will return to in Section 6), what does shortcut learning in artificial neural networks look like? Figure 2 shows a simple classification problem that a neural network is trained on (distinguishing a star from a moon).<sup>2</sup> When testing the model on similar data (blue) the network does very well—or so it may seem. Very much like the smart rats that tricked the experimenter, the network uses a shortcut to solve the classification problem by relying on the location of stars and moons. When location is controlled for, network performance deteriorates to random guessing (red). In this case (as is typical for object recognition), classification based on object shape would have been the intended solution, even though the difference between intended and shortcut solution is not something a neural network can possibly infer from the training data.

On a general level, any neural network (or machine learning algorithm) implements a decision rule which defines a relationship between input and output—in this example assigning a category to every input image. Shortcuts, the focus of this article, are one particular group of decision rules. In order to distinguish them from other decision rules, we here introduce a taxonomy of decision rules (visualised in Figure 3), starting from a very general rule and subsequently adding more constraints until we approach the intended solution.

<sup>2</sup>Code is available from <https://github.com/rgeirhos/shortcut-perspective>.



**Figure 2.** Toy example of shortcut learning in neural networks. When trained on a simple dataset of stars and moons (top row), a standard neural network (three layers, fully connected) can easily categorise novel similar exemplars (mathematically termed i.i.d. test set, defined later in Section 3). However, testing it on a slightly different dataset (o.o.d. test set, bottom row) reveals a shortcut strategy: The network has learned to associate object location with a category. During training, stars were always shown in the top right or bottom left of an image; moons in the top left or bottom right. This pattern is still present in samples from the i.i.d. test set (middle row) but not in o.o.d. test images (bottom row), exposing the shortcut.

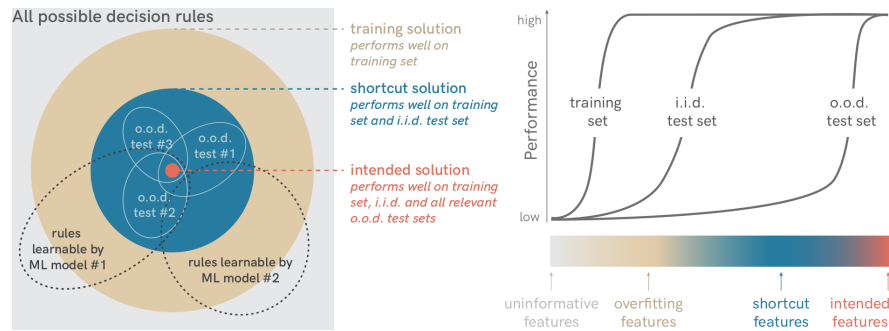
### (1) all possible decision rules, including non-solutions

Imagine a model that tries to solve the problem of separating stars and moons by predicting “star” every time it detects a white pixel in the image. This model uses an *uninformative feature* (the grey area in Figure 3) and does not reach good performance on the data it was trained on, since it implements a poor decision rule (both moon and star images contain white pixels). Typically, interesting problems have an abundant amount of non-solutions.

### (2) training solutions, including overfitting solutions

In machine learning it is common practice to split the available data randomly into a training and a test set. The training set is used to guide the model in its selection of a (hopefully useful) decision rule, and the test set is used to check whether the model achieves good performance on similar data it has not seen before. Mathematically, the notion of similarity between training and test set commonly referred to in machine learning is the assumption that the samples in both sets are drawn from the same distribution. This is the case if both the data generation mechanism and the sampling mechanism are identical. In practice this is achieved by randomising the split between training and test set. The test set is then called independent and identically distributed (i.i.d.) with regard to the training set. In order to achieve high average performance on the test set, a model needs to learn a function that is approximately correct within a subset of the input domain which covers most of the probability of the distribution. If a function is learned that yields the correct output on the training images but not on the i.i.d. test images, the learning machine uses *overfitting features* (the blue area in Figure 3).





**Figure 3.** Taxonomy of decision rules. Among the set of all possible rules, only some solve the training data. Among the solutions that solve the training data, only some generalise to an i.i.d. test set. Among those solutions, shortcuts fail to generalise to different data (o.o.d. test sets), but the intended solution does generalise.

### (3) i.i.d. test solutions, including shortcuts

Decision rules that solve both the training and i.i.d. test set typically score high on standard benchmark leaderboards. However, even the simple toy example can be solved through at least three different decision rules: (a) by shape, (b) by counting the number of white pixels (moons are smaller than stars) or (c) by location (which was correlated with object category in the training and i.i.d. test sets). As long as tests are performed only on i.i.d. data, it is impossible to distinguish between these. However, one can instead test models on datasets that are systematically different from the i.i.d. training and test data (also called *out-of-distribution* or *o.o.d.* data). For example, an o.o.d. test set with randomised object size will instantly invalidate a rule that counts white pixels. Which decision rule is the *intended solution* is clearly in the eye of the beholder, but humans often have clear expectations. In our toy example, humans typically classify by shape. A standard fully connected neural network<sup>3</sup> trained on this dataset, however, learns a location-based rule (see Figure 2). In this case, the network has used a *shortcut feature* (the blue area in Figure 3): a feature that helps to perform well on i.i.d. test data but fails in o.o.d. generalisation tests.

### (4) intended solution

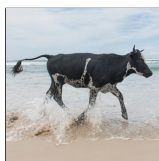
Decision rules that use the *intended features* (the red area in Figure 3) work well not only on an i.i.d. test set but also perform as intended on o.o.d. tests, where shortcut solutions fail. In the toy example, a decision rule based on object shape (the intended feature) would generalise to objects at a different location or with a different size. Humans typically have a strong intuition for what the intended solution should be capable of. Yet, for complex problems, intended solutions are mostly impossible to formalise, so machine learning is needed to estimate these solutions from examples. Therefore the choice of examples, among other aspects, influence how closely the intended solution can be approximated.

<sup>3</sup>A convolutional (rather than fully connected) network would be prevented from taking this shortcut by design.

## 4 Shortcuts: where do they come from?

Following this taxonomy, shortcuts are decision rules that perform well on i.i.d. test data but fail on o.o.d. tests, revealing a mismatch between intended and learned solution. It is clear that shortcut learning is to be avoided, but where do shortcuts come from, and what are the defining real-world characteristics of shortcuts that one needs to look out for when assessing a model or task through the lens of shortcut learning? There are two different aspects that one needs to take into account. First, shortcut opportunities (or shortcut features) in the data: possibilities for solving a problem differently than intended (Section 4.1). Second, feature combination: how different features are combined to form a decision rule (Section 4.2). Together, these aspects determine how a model generalises (Section 4.3).

### 4.1 Dataset: shortcut opportunities



What makes a cow a cow? To DNNs, a familiar background can be as important for recognition as the object itself, and sometimes even more important: A cow at an unexpected location (such as a beach rather than grassland) is not classified correctly [9]. Conversely, a lush hilly landscape without any animal at all might be labelled as a “herd of grazing sheep” by a DNN [22].

This example highlights how a systematic relationship between object and background or context can easily create a shortcut opportunity. If cows happen to be on grassland for most of the training data, detecting grass instead of cows becomes a successful strategy for solving a classification problem in an unintended way; and indeed many models base their predictions on context [23, 24, 25, 26, 9, 27, 10]. Many shortcut opportunities are a consequence of natural relationships, since grazing cows are typically surrounded by grassland rather than water. These so-called *dataset biases* have long been known to be problematic for machine learning algorithms [18]. Humans, too, are influenced by contextual biases (as evident from faster reaction times when objects appear in the expected context), but their predictions are much less affected when context is missing [28, 29, 30, 31]. In addition to shortcut opportunities that are fairly easy to recognise, deep learning has led to the discovery of much more subtle shortcut features, including high-frequency patterns that are almost invisible to the human eye [32, 33]. Whether easy to recognise or hard to detect, it is becoming more and more evident that shortcut opportunities are by no means disappearing when the size of a dataset is simply scaled up by some orders of magnitude (in the hope that this is sufficient to sample the diverse world that we live in [34]). Systematic biases are still present even in “Big Data” with large volume and variety, and consequently even large real-world datasets usually contain numerous shortcut opportunities. Overall, it is quite clear that data alone rarely constrains a model sufficiently, and that data cannot replace making assumptions [35]. The totality of all assumptions that a model incorporates (such as, e.g., the choice of architecture) is called the *inductive bias* of a model and will be discussed in more detail in Section 6.3.

## 4.2 Decision rule: shortcuts from discriminative learning



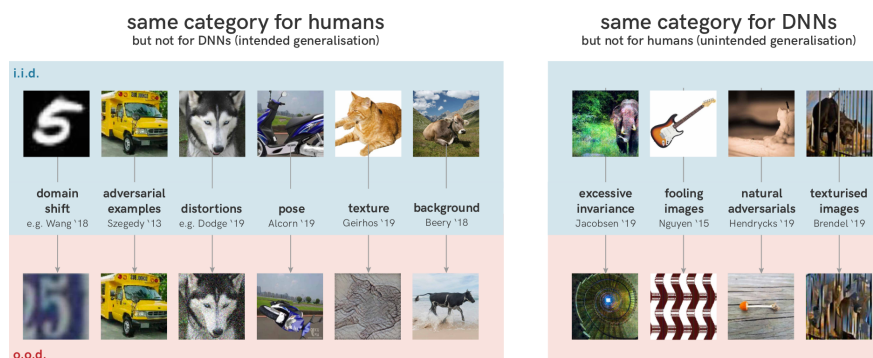
What makes a cat a cat? To standard DNNs, the example image on the left clearly shows an elephant, not a cat. Object textures and other local structures in images are highly useful for object classification in standard datasets [36], and DNNs strongly rely on texture cues for object classification, largely ignoring global object shape [37, 38].

In many cases, relying on object textures can be sufficient to solve an object categorisation task. Obviously, however, texture is only one of many attributes that define an object. Discriminative learning differs from generative modeling by picking any feature that is sufficient to reliably discriminate on a given dataset but the learning machine has no notion of how realistic examples typically look like and how the features used for discrimination are combined with other features that define an object. In our example, using textures for object classification becomes problematic if other intended attributes (like shape) are ignored entirely. This exemplifies the importance of feature combination: the definition of an object relies on a (potentially highly non-linear) combination of information from different sources or attributes that influence a decision rule.<sup>4</sup> In the example of the cat with elephant texture above, a shape-agnostic decision rule that merely relies on texture properties clearly fails to capture the task of object recognition as it is understood for human vision. While the model uses an important attribute (texture) it tends to equate it with the definition of the object missing out other important attributes such as shape. Of course, being aligned with the human decision rule does not always conform to our intention. In medical or safety-critical applications, for instance, we may instead seek an improvement over human performance.

Inferring human-interpretable object attributes like shape or texture from an image requires specific nonlinear computations. In typical end-to-end discriminative learning, this again may be prone to shortcut learning. Standard DNNs do not impose any human-interpretable requirements on intermediate image representations and thus might be severely biased to the extraction of overly simplistic features which only generalise under the specific design of the particular dataset used but easily fail otherwise. Discriminative feature learning goes as far that some decision rules only depend on a single predictive pixel [39, 40, 41] while all other evidence is ignored.<sup>5</sup> In principle, ignoring some evidence can be beneficial. In object recognition, for example, we want the decision rule to be invariant to an object shift. However, undesirable invariance (sometimes called *excessive invariance*) is harmful and modern machine learning models can be invariant to almost all features that humans would rely on when classifying an image [41].

<sup>4</sup>In Cognitive Science, this process is called *cue combination*.

<sup>5</sup>In models of animal learning, the *blocking effect* is a related phenomenon. Once a predictive cue/feature (say, a light flash) has been associated with an outcome (e.g. food), animals sometimes fail to associate a new, equally predictive cues with the same outcome [42, 43, 44].



**Figure 4.** Both human and machine vision generalise, but they generalise very differently. Left: image pairs that belong to the same category for humans, but not for DNNs. Right: images pairs assigned to the same category by a variety of DNNs, but not by humans.

### 4.3 Generalisation: how shortcuts can be revealed



What makes a guitar a guitar? When tested on this pattern never seen before, standard DNNs predict “guitar” with high certainty [45]. Exposed by the generalisation test, it seems that DNNs learned to detect certain patterns (curved guitar body? strings?) instead of guitars: a successful strategy on training and i.i.d. test data that leads to unintended generalisation on o.o.d. data.

This exemplifies the inherent link between shortcut learning and generalisation. By itself, generalisation is not a part of shortcut learning—but more often than not, shortcut learning is discovered through cases of unintended generalisation, revealing a mismatch between human-intended and model-learned solution. Interestingly, DNNs do not suffer from a general lack of o.o.d. generalisation (Figure 4) [45, 36, 46, 41]. DNNs recognise guitars even if only some abstract pattern is left—however, this remarkable generalisation performance is undesired, at least in this case. In fact, the set of images that DNNs classify as “guitar” with high certainty is incredibly big. To humans only some of these look like guitars, others like patterns (interpretable or abstract) and many more resemble white noise or even look like airplanes, cats or food [8, 45, 41]. Figure 4 on the right, for example, highlights a variety of image pairs that have hardly anything in common for humans but belong to the same category for DNNs. Conversely, to the human eye an image’s category is not altered by innocuous *distribution shifts* like rotating objects or adding a bit of noise, but if these changes interact with the shortcut features that DNNs are sensitive to, they completely derail neural network predictions [8, 47, 9, 48, 49, 50, 38]. This highlights that generalisation failures are neither a failure to learn nor a failure to generalise at all, but instead a failure to generalise in the intended direction—generalisation and robustness can be considered the flip side of shortcut learning. Using a certain set of features creates insensitivity towards other features. Only if the selected features are still present after a distribution shift, a model generalises o.o.d.

## 5 Shortcut learning across deep learning

Taken together, we have seen how shortcuts are based on dataset shortcut opportunities and discriminative feature learning that result in a failure to generalise as intended. We will now turn to specific application areas, and discover how this general pattern appears across Computer Vision, Natural Language Processing, Agent-based (Reinforcement) Learning and Fairness / algorithmic decision-making. While shortcut learning is certainly not limited to these areas, they might be the most prominent ones where the problem has been observed.

**Computer Vision** To humans, for example, a photograph of a car still shows the same car even when the image is slightly transformed. To DNNs, in contrast, innocuous transformations can completely change predictions. This has been reported in various cases such as shifting the image by a few pixels [47], rotating the object [49], adding a bit of random noise or blur [51, 50, 52, 53] or (as discussed earlier) by changing background [9] or texture while keeping the shape intact [38] (see Figure 4 for examples). Some key problems in Computer Vision are linked to shortcut learning. For example, transferring model performance across datasets (*domain transfer*) is challenging because models often use domain-specific shortcut features, and shortcuts limit the usefulness of unsupervised representations [54]. Furthermore, *adversarial examples* are particularly tiny changes to an input image that completely derail model predictions [8] (an example is shown in Figure 4). Invisible to the human eye, those changes modify highly predictive patterns that DNNs use to classify objects [33]. In this sense, adversarial examples—one of the most severe failure cases of neural networks—can at least partly be interpreted as a consequence of shortcut learning.

**Natural Language Processing** The widely used language model BERT has been found to rely on superficial cue words. For instance, it learned that within a dataset of natural language arguments, detecting the presence of “not” was sufficient to perform above chance in finding the correct line of argumentation. This strategy turned out to be very useful for drawing a conclusion without understanding the content of a sentence [14]. Natural Language Processing suffers from very similar problems as Computer Vision and other fields. Shortcut learning starts from various dataset biases such as annotation artefacts [55, 56, 57, 58]. Feature combination crucially depends on shortcut features like word length [59, 60, 14, 61], and consequently leads to a severe lack of robustness such as an inability to generalise to more challenging test conditions [62, 63, 64, 65]. Attempts like incorporating a certain degree of unsupervised training as employed in prominent language models like BERT [5] and GPT-2 [66] did not resolve the problem of shortcut learning [14].

**Agent-based (Reinforcement) Learning** Instead of learning how to play Tetris, an algorithm simply learned to pause the game to evade losing [67]. Systems of Agent-based Learning are usually trained using Reinforcement Learning and related approaches such as evolutionary algorithms. In both cases, designing a good reward function is crucial, since a reward function measures how close a system is to solving the problem. However, they all too often contain unexpected shortcuts that allow for so-called *reward hacking* [68]. The existence of loopholes exploited by machines that follow the letter (and not the spirit) of the reward function highlight how difficult it is to design a shortcut-free reward function [69]. Reinforcement Learning is also a widely used method in Robotics, where there is a commonly observed *generalisation* or *reality gap* between simulated training

environment and real-world use case. This can be thought of as a consequence of narrow shortcut learning by adapting to specific details of the simulation. Introducing additional variation in colour, size, texture, lighting, etc. helps a lot in closing this gap [70, 71].

**Fairness & algorithmic decision-making** Tasked to predict strong candidates on the basis of their résumés, a hiring tool developed by Amazon was found to be biased towards preferring men. The model, trained on previous human decisions, found gender to be such a strong predictor that even removing applicant names would not help: The model always found a way around, for instance by inferring gender from all-woman college names [13]. This exemplifies how some—but not all—problems of (un)fair algorithmic decision-making are linked to shortcut learning: Once a predictive feature is found by a model, even if it is just an artifact of the dataset, the model’s decision rule may depend entirely on the shortcut feature. When human biases are not only replicated, but worsened by a machine, this is referred to as *bias amplification* [72]. Other shortcut strategies include focusing on the majority group in a dataset while accepting high error rates for underrepresented groups [12, 73], which can amplify existing societal disparities and even create new ones over time [74]. In the dynamical setting a related problem is called *disparity amplification* [74], where sequential feedback loops may amplify a model’s reliance on a majority group. It should be emphasised, however, that fairness is an active research area of machine learning closely related to invariance learning that might be useful to quantify and overcome biases of both machine and human decision making.

## 6 Diagnosing and understanding shortcut learning

Shortcut learning currently occurs across deep learning, causing machines to fail unexpectedly. Many individual elements of shortcut learning have been identified long ago by parts of the machine learning community and some have already seen substantial progress, but currently a variety of approaches are explored without a commonly accepted strategy. We here outline three actionable steps towards diagnosing and analysing shortcut learning.

### 6.1 Interpreting results carefully

**Distinguishing datasets and underlying abilities** Shortcut learning is most deceptive when gone unnoticed. The most popular benchmarks in machine learning still rely on i.i.d. testing which drags attention away from the need to verify how closely this test performance measures the *underlying ability* one is actually interested in. For example, the ImageNet dataset [75] was intended to measure the ability “object recognition”, but DNNs seem to rely mostly on “counting texture patches” [36]. Likewise, instead of performing “natural language inference”, some language models perform well on datasets by simply detecting correlated key words [56]. Whenever there is a discrepancy between the simplicity with which a dataset (e.g. ImageNet, SQuAD) can be solved and the complexity evoked by the high-level description of the underlying ability (e.g. object recognition, scene understanding, argument comprehension), it is important to bear in mind that a dataset is useful only for as long as it is a good proxy for the ability one is actually interested in [56, 76]. We would hardly be intrigued by reproducing human-defined labels on datasets per se (a lookup table would do just as well in this case)—it is the underlying generalisation ability that we truly intend to measure, and ultimately improve upon.

**Morgan’s Canon for machine learning** Recall the cautionary tale of rats sniffing rather than seeing colour, described in Section 2.1. Animals often trick experimenters by solving an experimental paradigm (i.e., dataset) in an unintended way without using the underlying ability one is actually interested in. This highlights how incredibly difficult it can be for humans to imagine solving a tough challenge in any other way than the human way: Surely, at Marr’s implementational level [77] there may be differences between rat and human colour discrimination. But at the algorithmic level there is often a tacit assumption that human-like performance implies human-like strategy (or algorithm) [78]. This *same strategy assumption* is paralleled by deep learning: Surely, DNN units are different from biological neurons—but if DNNs successfully recognise objects, it seems natural to assume that they are using object shape like humans do [37, 36, 38].

Comparative Psychology with its long history of comparing mental abilities across species has coined a term for the fallacy to confuse human-centered interpretations of an observed behaviour and the actual behaviour at hand (which often has a much simpler explanation): *anthropomorphism*, “the tendency of humans to attribute human-like psychological characteristics to nonhumans on the basis of insufficient empirical evidence” [79, p. 5]. As a reaction to the widespread occurrence of this fallacy, psychologist Lloyd Morgan developed a conservative guideline for interpreting non-human behaviour as early as 1903. It later became known as Morgan’s Canon: “In no case is an animal activity to be interpreted in terms of higher psychological processes if it can be fairly interpreted in terms of processes which stand lower on the scale of psychological evolution and development” [80, p. 59]. Picking up on a simple correlation, for example, would be considered a process that stands low on this psychological scale whereas “understanding a scene” would be considered much higher. It has been argued that Morgan’s Canon can and should be applied to interpreting machine learning results [79], and we consider it to be especially relevant in the context of shortcut learning. Accordingly, it might be worth acquiring the habit to confront machine learning models with a “Morgan’s Canon for machine learning”<sup>6</sup>: *Never attribute to high-level abilities that which can be adequately explained by shortcut learning.*

**Testing (surprisingly) strong baselines** In order to find out whether a result may also be explained by shortcut learning, it can be helpful to test whether a baseline model exceeds expectations even though it does not use intended features. Examples include using nearest neighbours for scene completion and estimating geolocation [81, 82], object recognition with local features only [36], reasoning based on single cue words [59, 14] or answering questions about a movie without ever showing the movie to a model [83]. Importantly, this is not meant to imply that DNNs cannot acquire high-level abilities. They certainly do have the potential to solve complex challenges and serve as scientific models for prediction, explanation and exploration [84]—however, we must not confuse performance on a *dataset* with the acquisition of an *underlying ability*.

## 6.2 Detecting shortcuts: towards o.o.d. generalisation tests

**Making o.o.d. generalisation tests a standard practice** Currently, measuring model performance by assessing validation performance on an i.i.d. test set is at the very heart of the vast majority of machine learning benchmarks. Unfortunately, in real-world settings

<sup>6</sup>Our formulation is adapted from Hanlon’s razor, “Never attribute to malice that which can be adequately explained by stupidity”.

the i.i.d. assumption is rarely justified; in fact, this assumption has been called “the big lie in machine learning” [85]. While any metric is typically only an approximation of what we truly intend to measure, the i.i.d. performance metric may not be a good approximation as it can often be misleading, giving a false sense of security. In Section 2.2 we described how Bob gets a good grade on a multiple-choice exam through rote learning. Bob’s reproductive approach gives the superficial appearance of excellent performance, but it would not generalise to a more challenging test. Worse yet, as long as Bob continues to receive good grades through surface learning, he is unlikely to change his learning strategy.

Within the field of Education, what is the best practice to avoid surface learning? It has been argued that changing the type of examination from multiple-choice tests to essay questions discourages surface learning, and indeed surface approaches typically fail on these kinds of exams [21]. Essay questions, on the other hand, encourage so-called *deep* or *transformational* learning strategies [86, 87], like Alice’s focus on understanding. This in turn enables transferring the learned content to *novel* problems and consequently achieves a much better overlap between the educational objectives of the teacher and what the students actually learn [88]. We can easily see the connection to machine learning—transferring knowledge to novel problems corresponds to testing generalisation beyond the narrowly learned setting [89, 90, 91]. If model performance is assessed only on i.i.d. test data, then we are unable to discover whether the model is actually acquiring the ability we think it is, since exploiting shortcuts often leads to deceptively good results on standard metrics [92]. We, among many others [93, 78, 94, 95, 96], have explored a variety of o.o.d. tests and we hope it will be possible to identify a sufficiently simple and effective test procedure that could replace i.i.d. testing as a new standard method for benchmarking machine learning models in the future.

**Designing good o.o.d. tests** While a distribution shift (between i.i.d. and o.o.d. data) has a clear mathematical definition, it can be hard to detect in practice [101, 102]. In these cases, training a classifier to distinguish samples in dataset A from samples in dataset A’ can reveal a distribution shift. We believe that good o.o.d. tests should fulfill at least the following three conditions: First, per definition there needs to be a *clear distribution shift*, a shift that may or may not be distinguishable by humans. Second, it should have a *well-defined intended solution*. Training on natural images while testing on white noise would technically constitute an o.o.d. test but lacks a solution. Third, a good o.o.d. test is a test where the majority of *current models struggle*. Typically, the space of all conceivable o.o.d. tests includes numerous uninteresting tests. Thus given limited time and resources, one might want to focus on challenging test cases. As models evolve, generalisation benchmarks will need to evolve as well, which is exemplified by the Winograd Schema Challenge [103]. Initially designed to overcome shortcut opportunities caused by the open-ended nature of the Turing test, this common-sense reasoning benchmark was scrutinised after modern language models started to perform suspiciously well—and it indeed contained more shortcut opportunities than originally envisioned [104], highlighting the need for revised tests. Fortunately, stronger generalisation tests are beginning to gain traction across deep learning. While o.o.d. tests will likely need to evolve alongside the models they aim to evaluate, a few current encouraging examples are listed in Box I. In summary, rigorous generalisation benchmarks are crucial when distinguishing between the intended and a shortcut solution, and it would be extremely useful if a strong generally applicable testing procedure will emerge from this range of approaches.



**Box I. EXAMPLES OF INTERESTING O.O.D. BENCHMARKS**

We here list a few selected, encouraging examples of o.o.d. benchmarks.

**Adversarial attacks** can be seen as testing on model-specific worst-case o.o.d. data, which makes it an interesting diagnostic tool. If a successful adversarial attack [8] can change model predictions without changing semantic content, this is an indication that something akin to shortcut learning may be occurring [33, 97].

**ARCT with removed shortcuts** is a language argument comprehension dataset that follows the idea of removing known shortcut opportunities from the data itself in order to create harder test cases [14].

**Cue conflict stimuli** like images with conflicting texture and shape information pitch features/cues against each other, such as an intended against an unintended cue [38]. This approach can easily be compared to human responses.

**ImageNet-A** is a collection of natural images that several state-of-the-art models consistently classify wrongly. It thus benchmarks models on worst-case natural images [46].

**ImageNet-C** applies 15 different image corruptions to standard test images, an approach we find appealing for its variety and usability [52].

**ObjectNet** introduces the idea of scientific controls into o.o.d. benchmarking, allowing to disentangle the influence of background, rotation and viewpoint [98].

**PACS** and other domain generalisation datasets require extrapolation beyond i.i.d. data per design by testing on a domain different from training data (e.g. cartoon images) [99].

**Shift-MNIST / biased CelebA / unfair dSprites** are controlled toy datasets that introduce correlations in the training data (e.g. class-predictive pixels or image quality) and record the accuracy drop on clean test data as a way of finding out how prone a given architecture and loss function are to picking up on shortcuts [39, 40, 100, 41].

### 6.3 Shortcuts: why are they learned?

**The “Principle of Least Effort”** Why are machines so prone to learning shortcuts, detecting grass instead of cows [9] or a metal token instead of pneumonia [15]? Exploiting those shortcuts seems much *easier* for DNNs than learning the intended solution. But what determines whether a solution is easy to learn? In Linguistics, a related phenomenon is called the “Principle of Least Effort” [119], the observation that language speakers generally try to minimise the amount of effort involved in communication. For example, the use of “plane” is becoming more common than “airplane”, and in pronouncing “cupboard”, “p” and “b” are merged into a single sound [120, 121]. Interestingly, whether a language change makes it easier for the speaker doesn’t always simply depend on objective measures like word length. On the contrary, this process is shaped by a variety of different factors, including the anatomy (architecture) of the human speech organs and previous language experience (training data).

**Box II. SHORTCUT LEARNING & INDUCTIVE BIASES**

The four components listed below determine the *inductive bias* of a model and dataset: the set of assumptions that influence which solutions are learnable, and how readily they can be learned. Although in theory DNNs can approximate any function (given potentially infinite capacity) [105], their inductive bias plays an important role for the types of patterns they prefer to learn given finite capacity and data.

- **Structure: architecture.** Convolutions make it harder for a model to use location—a prior [106] that is so powerful for natural images that even untrained networks can be used for tasks like image inpainting and denoising [107]. In Natural Language Processing, transformer architectures [108] use *attention layers* to understand the context by modelling relationships between words. In most cases, however, it is hard to understand the implicit priors in a DNN and even standard elements like ReLU activations can lead to unexpected effects like unwarranted confidence [109].
- **Experience: training data.** As discussed in Section 4.1, shortcut opportunities are present in most data and rarely disappear by adding more data [32, 69, 56, 38, 33]. Modifying the training data to block specific shortcuts has been demonstrated to work for reducing adversarial vulnerability [110] and texture bias [38].
- **Goal: loss function.** The most commonly used loss function for classification, *cross-entropy*, encourages DNNs to stop learning once a simple predictor is found; a modification can force neural networks to use all available information [41]. Regularisation terms that use additional information about the training data have been used to disentangle intended features from shortcut features [39, 111].
- **Learning: optimisation.** Stochastic gradient descent and its variants bias DNNs towards learning simple functions [112, 113, 114, 115]. The learning rate influences which patterns networks focus on: Large learning rates lead to learning simple patterns that are shared across examples, while small learning rates facilitate complex pattern learning and memorisation [116, 117]. The complex interactions between training method and architecture are poorly understood so far; strong claims can only be made for simple cases [118].

**Understanding the influence of inductive biases** In a similar vein, whether a solution is easy to learn for machines does not simply depend on the data but on all of the four components of a machine learning algorithm: architecture, training data, loss function, and optimisation. Often, the training process starts with feeding training data to the model with a fixed architecture and randomly initialised parameters. When the model's prediction is compared to ground truth, the loss function measures the prediction's quality. This supervision signal is used by an optimiser for adapting the model's internal parameters such that the model makes a better prediction the next time. Taken together, these four components (which determine the *inductive bias* of a model) influence how certain solutions are much easier to learn than others, and thus ultimately determine whether a shortcut is learned instead of the intended solution [122]. Box II provides an overview of the connections between shortcut learning and inductive biases.

## 7 Beyond shortcut learning

A lack of out-of-distribution generalisation can be observed all across machine learning. Consequently, a significant fraction of machine learning research is concerned with overcoming shortcut learning, albeit not necessarily as a concerted effort. Here we highlight connections between different research areas. Note that an exhaustive list would be out of the scope for this work. Instead, we cover a diverse set of approaches we find promising, each providing a unique perspective on learning beyond shortcut learning.

**Domain-specific prior knowledge** Avoiding reliance on unintended cues can be achieved by designing architectures and data-augmentation strategies that discourage learning shortcut features. If the orientation of an object does not matter for its category, either data-augmentation or hard-coded rotation invariance [123] can be applied. This strategy can be applied to almost any well-understood transformation of the inputs and finds its probably most general form in auto-augment as an augmentation strategy [124]. Extreme data-augmentation strategies are also the core ingredient of the most successful semi-supervised [125] and self-supervised learning approaches to date [126, 127].

**Adversarial examples and robustness** Adversarial attacks are a powerful analysis tool for worst-case generalisation [8]. Adversarial examples can be understood as counterfactual explanations, since they are the smallest change to an input that produces a certain output. Achieving counterfactual explanations of predictions aligned with human intention makes the ultimate goals of adversarial robustness tightly coupled to causality research in machine learning [128]. Adversarially robust models are somewhat more aligned with humans and show promising generalisation abilities [129, 130]. While adversarial attacks test model performance on model-dependent worst-case noise, a related line of research focuses on model-independent noise like image corruptions [51, 52].

**Domain adaptation, -generalisation and -randomisation** These areas are explicitly concerned with out-of-distribution generalisation. Usually, multiple distributions are observed during training time and the model is supposed to generalise to a new distribution at test time. Under certain assumptions the intended (or even causal) solution can be learned from multiple domains and environments [131, 39, 111]. In robotics, domain randomisation (setting certain simulation parameters randomly during training) is a very successful approach for learning policies that generalise to similar situations in the real-world [70].

**Fairness** Fairness research aims at making machine decisions “fair” according to a certain definition [132]. Individual fairness aims at treating similar individuals similarly while group fairness aims at treating subgroups no different than the rest of the population [133, 134]. Fairness is closely linked to generalisation and causality [135]. Sensitive group membership can be viewed as a domain indicator: Just like machine decisions should not typically be influenced by changing the domain of the data, they also should not be biased against minority groups.

**Meta-learning** Meta-learning seeks to learn how to learn. An intermediate goal is to learn representations that can adapt quickly to new conditions [136, 137, 138]. This ability is connected to the identification of causal graphs [139] since learning causal features allows for small changes when changing environments.

**Generative modelling and disentanglement** Learning to generate the observed data forces a neural network to model every variation in the training data. By itself, however, this does not necessarily lead to representations useful for downstream tasks [140], let alone out-of-distribution generalisation. Research on disentanglement addresses this shortcoming by learning generative models with well-structured latent representations [141]. The goal is to recover the true generating factors of the data distribution from observations [142] by identifying independent causal mechanisms [128].

## 8 Conclusion

*“The road reaches every place, the short cut only one”*

— James Richardson [143]

Science aims for understanding. While deep learning as an engineering discipline has seen tremendous progress over the last few years, deep learning as a scientific discipline is still lagging behind in terms of understanding the principles and limitations that govern how machines learn to extract patterns from data. A deeper understanding of how to overcome shortcut learning is of relevance beyond the current application domains of machine learning and there might be interesting future opportunities for cross-fertilisation with other disciplines such as Economics (designing management incentives that do not jeopardise long-term success by rewarding unintended “shortcut” behaviour) or Law (creating laws without “loophole” shortcut opportunities). Until the problem is solved, however, we offer the following four recommendations:

### (1) Connecting the dots: shortcut learning is ubiquitous

Shortcut learning appears to be a ubiquitous characteristic of learning systems, biological and artificial alike. Many of deep learning’s problems are connected through shortcut learning—models exploit dataset shortcut opportunities, select only a few predictive features instead of taking all evidence into account, and consequently suffer from unexpected generalisation failures. “Connecting the dots” between affected areas is likely to facilitate progress, and making progress can generate highly valuable impact across various applications domains.

### (2) Interpreting results carefully

Discovering a shortcut often reveals the existence of an easy solution to a seemingly complex dataset. We argue that we will need to exercise great care before attributing high-level abilities like “object recognition” or “language understanding” to machines, since there is often a much simpler explanation.

### (3) Testing o.o.d. generalisation

Assessing model performance on i.i.d. test data (as the majority of current benchmarks do) is insufficient to distinguish between intended and unintended (shortcut) solutions. Consequently, o.o.d. generalisation tests will need to become the rule rather than the exception.

### (4) Understanding what makes a solution easy to learn

DNNs always learn the easiest possible solution to a problem, but understanding which solutions are easy (and thus likely to be learned) requires disentangling the influence of

structure (architecture), experience (training data), goal (loss function) and learning (optimisation), as well as a thorough understanding of the interactions between these factors.

Shortcut learning is one of the key roadblocks towards fair, robust, deployable and trustworthy machine learning. While overcoming shortcut learning in its entirety may potentially be impossible, any progress towards mitigating it will lead to a better alignment between learned and intended solutions. This holds the promise that machines behave much more reliably in our complex and ever-changing world, even in situations far away from their training experience. Furthermore, machine decisions would become more transparent, enabling us to detect and remove biases more easily. Currently, the research on shortcut learning is still fragmented into various communities. With this perspective we hope to fuel discussions across these different communities and to initiate a movement that pushes for a new standard paradigm of generalisation that is able to replace the current i.i.d. tests.

### Acknowledgement

The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting R.G. and C.M.; the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) for supporting C.M. via grant EC 479/1-1; the Collaborative Research Center (Projektnummer 276693517—SFB 1233: Robust Vision) for supporting M.B. and F.A.W.; the German Federal Ministry of Education and Research through the Tübingen AI Center (FKZ 01IS18039A) for supporting W.B. and M.B.; as well as the Natural Sciences and Engineering Research Council of Canada and the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/Interior Business Center (DoI/IBC) contract number D16PC00003 for supporting J.J.

The authors would like to thank Judy Borowski, Max Burg, Santiago Cadena, Alexander S. Ecker, Lisa Eisenberg, Roland Fleming, Ingo Fründ, Samuel Greiner, Florian Griebner, Shaiyan Keshvari, Ruth Kessler, David Klindt, Matthias Kümmerer, Benjamin Mitzkus, Hendrikje Nienborg, Jonas Rauber, Evgenia Rusak, Steffen Schneider, Lukas Schott, Tino Sering, Yash Sharma, Matthias Tangemann, Roland Zimmermann and Tom Wallis for helpful discussions.

### Author contributions

The project was initiated by R.G. and C.M. and led by R.G. with support from C.M. and J.J.; M.B. and W.B. reshaped the initial thrust of the perspective and together with R.Z. supervised the machine learning components. The toy experiment was conducted by J.J. with input from R.G. and C.M. Most figures were designed by R.G. and W.B. with input from all other authors. Figure 2 (left) was conceived by M.B. The first draft was written by R.G., J.J. and C.M. with input from F.A.W. All authors contributed to the final version and provided critical revisions from different perspectives.

### References

- [1] He, K., Zhang, X., Ren, S. & Sun, J. Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. In *Proceedings of the IEEE International Conference on Computer Vision*, 1026–1034 (2015).
- [2] Silver, D. *et al.* Mastering the game of Go with deep neural networks and tree search. *Nature* **529**, 484 (2016).
- [3] Moravčík, M. *et al.* Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. *Science* **356**, 508–513 (2017).

- [4] Rajpurkar, P. *et al.* Chexnet: Radiologist-level pneumonia detection on chest X-rays with deep learning. *arXiv:1711.05225* (2017).
- [5] Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv:1810.04805* (2018).
- [6] Rolnick, D. *et al.* Tackling climate change with machine learning. *arXiv preprint arXiv:1906.05433* (2019).
- [7] Reichstein, M. *et al.* Deep learning and process understanding for data-driven earth system science. *Nature* **566**, 195 (2019).
- [8] Szegedy, C. *et al.* Intriguing properties of neural networks. *arXiv:1312.6199* (2013).
- [9] Beery, S., Van Horn, G. & Perona, P. Recognition in terra incognita. In *Proceedings of the European Conference on Computer Vision*, 456–473 (2018).
- [10] Rosenfeld, A., Zemel, R. & Tsotsos, J. K. The elephant in the room. *arXiv preprint arXiv:1808.03305* (2018).
- [11] Heuer, H., Monz, C. & Smeulders, A. W. Generating captions without looking beyond objects. *arXiv preprint arXiv:1610.03708* (2016).
- [12] Buolamwini, J. & Gebru, T. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on Fairness, Accountability and Transparency*, 77–91 (2018).
- [13] Dastin, J. Amazon scraps secret AI recruiting tool that showed bias against women. <https://reut.rs/20d9fPr> (2018).
- [14] Niven, T. & Kao, H.-Y. Probing neural network comprehension of natural language arguments. *arXiv preprint arXiv:1907.07355* (2019).
- [15] Zech, J. R. *et al.* Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study. *PLoS Medicine* **15**, e1002683 (2018).

**This study highlights the importance of testing model generalisation in the medical context.**

- [16] Bickel, S., Brückner, M. & Scheffer, T. Discriminative learning under covariate shift. *Journal of Machine Learning Research* **10**, 2137–2155 (2009).
- [17] Schölkopf, B. *et al.* On causal and anticausal learning. In *International Conference on Machine Learning*, 1255–1262 ([SI: sn], 2012).
- [18] Torralba, A. & Efros, A. A. Unbiased look at dataset bias. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2011).

**This study provides a comprehensive overview of dataset biases in computer vision.**

- [19] Branwen, G. The neural net tank urban legend. <https://www.gwern.net/Tanks> (2011).
- [20] Pfungst, O. *Clever Hans: (the horse of Mr. Von Osten.) a contribution to experimental animal and human psychology* (Holt, Rinehart and Winston, 1911).
- [21] Scouller, K. The influence of assessment method on students' learning approaches: Multiple choice question examination versus assignment essay. *Higher Education* **35**, 453–472 (1998).

- [22] Shane, J. Do neural nets dream of electric sheep? (2018). URL <https://aiweirdness.com/post/171451900302/do-neural-nets-dream-of-electric-sheep>.
- [23] Wichmann, F. A., Drewes, J., Rosas, P. & Gegenfurtner, K. R. Animal detection in natural scenes: Critical features revisited. *Journal of Vision* **10**, 6–6 (2010).
- [24] Ribeiro, M. T., Singh, S. & Guestrin, C. Why should I trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1135–1144 (ACM, 2016).
- [25] Zhu, Z., Xie, L. & Yuille, A. L. Object recognition with and without objects. *arXiv preprint arXiv:1611.06596* (2016).
- [26] Wang, J. *et al.* Visual concepts and compositional voting. *arXiv preprint arXiv:1711.04451* (2017).
- [27] Dawson, M., Zisserman, A. & Nellåker, C. From same photo: Cheating on visual kinship challenges. In *Asian Conference on Computer Vision*, 654–668 (Springer, 2018).
- [28] Biederman, I. *On the semantics of a glance at a scene* (Hillsdale, NJ: Erlbaum, 1981).
- [29] Biederman, I., Mezzanotte, R. J. & Rabinowitz, J. C. Scene perception: Detecting and judging objects undergoing relational violations. *Cognitive Psychology* **14**, 143–177 (1982).
- [30] Oliva, A. & Torralba, A. The role of context in object recognition. *Trends in Cognitive Sciences* **11**, 520–527 (2007).
- [31] Castelhana, M. S. & Heaven, C. Scene context influences without scene gist: Eye movements guided by spatial associations in visual search. *Psychonomic Bulletin & Review* **18**, 890–896 (2011).
- [32] Jo, J. & Bengio, Y. Measuring the tendency of CNNs to learn surface statistical regularities. *arXiv preprint arXiv:1711.11561* (2017).
- [33] Ilyas, A. *et al.* Adversarial examples are not bugs, they are features. *arXiv preprint arXiv:1905.02175* (2019).
- This study shows how learning imperceptible predictive features leads to adversarial vulnerability.**
- [34] Halevy, A., Norvig, P. & Pereira, F. The unreasonable effectiveness of data. *Intelligent Systems* (2009).
- [35] Wolpert, D. H. & Macready, W. G. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation* **1**, 67–82 (1997).
- [36] Brendel, W. & Bethge, M. Approximating CNNs with bag-of-local-features models works surprisingly well on ImageNet. In *International Conference on Learning Representations* (2019).
- [37] Baker, N., Lu, H., Erlikhman, G. & Kellman, P. J. Deep convolutional networks do not classify based on global object shape. *PLoS Computational Biology* **14**, e1006613 (2018).

- [38] Geirhos, R. *et al.* ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations* (2019).

**This article shows how shortcut feature combination strategies are linked to distortion robustness.**

- [39] Heinze-Deml, C. & Meinshausen, N. Conditional variance penalties and domain shift robustness. *arXiv:1710.11469* (2017).
- [40] Malhotra, G. & Bowers, J. What a difference a pixel makes: An empirical examination of features used by CNNs for categorisation. In *International Conference on Learning Representations* (2019).
- [41] Jacobsen, J.-H., Behrmann, J., Zemel, R. & Bethge, M. Excessive invariance causes adversarial vulnerability. In *International Conference on Learning Representations* (2019).
- [42] Kamin, L. J. Predictability, surprise, attention, and conditioning. *Punishment and aversive behavior* 279–96 (1969).
- [43] Dickinson, A. *Contemporary animal learning theory*, vol. 1 (CUP Archive, 1980).
- [44] Bouton, M. E. *Learning and behavior: A contemporary synthesis*. (Sinauer Associates, 2007).
- [45] Nguyen, A., Yosinski, J. & Clune, J. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 427–436 (IEEE, 2015).
- [46] Hendrycks, D., Zhao, K., Basart, S., Steinhardt, J. & Song, D. Natural adversarial examples. *arXiv preprint arXiv:1907.07174* (2019).
- [47] Azulay, A. & Weiss, Y. Why do deep convolutional networks generalize so poorly to small image transformations? *arXiv:1805.12177* (2018).
- [48] Wang, M. & Deng, W. Deep visual domain adaptation: A survey. *Neurocomputing* **312**, 135–153 (2018).
- [49] Alcorn, M. A. *et al.* Strike (with) a pose: Neural networks are easily fooled by strange poses of familiar objects. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2019).
- [50] Dodge, S. & Karam, L. Human and DNN classification performance on images with quality distortions: A comparative study. *ACM Transactions on Applied Perception (TAP)* **16**, 7 (2019).
- [51] Geirhos, R. *et al.* Generalisation in humans and deep neural networks. In *Advances in Neural Information Processing Systems* (2018).
- [52] Hendrycks, D. & Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations* (2019).
- [53] Michaelis, C. *et al.* Benchmarking robustness in object detection: Autonomous driving when winter is coming. *arXiv preprint arXiv:1907.07484* (2019).
- [54] Minderer, M., Bachem, O., Houlsby, N. & Tschannen, M. Automatic shortcut removal for self-supervised representation learning. *arXiv preprint arXiv:2002.08822* (2020).



- [55] Goyal, Y., Khot, T., Summers-Stay, D., Batra, D. & Parikh, D. Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6904–6913 (2017).
- [56] Gururangan, S. *et al.* Annotation artifacts in Natural Language Inference data. *arXiv preprint arXiv:1803.02324* (2018).

**This article highlights how Natural Language Inference models learn heuristics that exploit superficial cues.**

- [57] Kaushik, D. & Lipton, Z. C. How much reading does reading comprehension require? A critical investigation of popular benchmarks. *arXiv preprint arXiv:1808.04926* (2018).
- [58] Geva, M., Goldberg, Y. & Berant, J. Are we modeling the task or the annotator? An investigation of annotator bias in natural language understanding datasets. *arXiv preprint arXiv:1908.07898* (2019).
- [59] Poliak, A., Naradowsky, J., Haldar, A., Rudinger, R. & Van Durme, B. Hypothesis only baselines in Natural Language Inference. *arXiv preprint arXiv:1805.01042* (2018).
- [60] Kavumba, P. *et al.* When choosing plausible alternatives, Clever Hans can be clever. *arXiv preprint arXiv:1911.00225* (2019).
- [61] McCoy, R. T., Pavlick, E. & Linzen, T. Right for the wrong reasons: Diagnosing syntactic heuristics in Natural Language Inference. *arXiv preprint arXiv:1902.01007* (2019).
- [62] Agrawal, A., Batra, D. & Parikh, D. Analyzing the behavior of visual question answering models. *arXiv preprint arXiv:1606.07356* (2016).
- [63] Belinkov, Y. & Bisk, Y. Synthetic and natural noise both break neural machine translation. *arXiv preprint arXiv:1711.02173* (2017).
- [64] Jia, R. & Liang, P. Adversarial examples for evaluating reading comprehension systems. *arXiv preprint arXiv:1707.07328* (2017).
- [65] Glockner, M., Shwartz, V. & Goldberg, Y. Breaking NLI systems with sentences that require simple lexical inferences. *arXiv preprint arXiv:1805.02266* (2018).
- [66] Radford, A. *et al.* Language models are unsupervised multitask learners. *OpenAI Blog* **1** (2019).
- [67] Murphy VII, T. The first level of Super Mario Bros. is easy with lexicographic orderings and time travel. *The Association for Computational Heresy (SIGBOVIK) 2013* 112 (2013).
- [68] Amodei, D. *et al.* Concrete problems in AI safety. *arXiv preprint arXiv:1606.06565* (2016).
- [69] Lehman, J. *et al.* The surprising creativity of digital evolution: A collection of anecdotes from the evolutionary computation and artificial life research communities. *arXiv preprint arXiv:1803.03453* (2018).

**This paper provides a comprehensive collection of anecdotes about shortcut learning / reward hacking in Reinforcement Learning and beyond.**

- [70] Tobin, J. *et al.* Domain randomization for transferring deep neural networks from simulation to the real world. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 23–30 (IEEE, 2017).
- [71] Akkaya, I. *et al.* Solving Rubik’s Cube with a robot hand. *arXiv:1910.07113* (2019).
- [72] Zhao, J., Wang, T., Yatskar, M., Ordonez, V. & Chang, K.-W. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. *arXiv preprint arXiv:1707.09457* (2017).

**This study shows how algorithms amplify social biases to boost performance.**

- [73] Rich, A. S. & Gureckis, T. M. Lessons for artificial intelligence from the study of natural stupidity. *Nature Machine Intelligence* **1**, 174 (2019).
- [74] Hashimoto, T. B., Srivastava, M., Namkoong, H. & Liang, P. Fairness without demographics in repeated loss minimization. *arXiv preprint arXiv:1806.08010* (2018).
- [75] Russakovsky, O. *et al.* ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision* **115**, 211–252 (2015).
- [76] Zellers, R., Holtzman, A., Bisk, Y., Farhadi, A. & Choi, Y. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830* (2019).
- [77] Marr, D. *Vision: A computational investigation into the human representation and processing of visual information* (W.H. Freeman and Company, San Francisco, 1982).
- [78] Borowski, J. *et al.* The notorious difficulty of comparing human and machine perception. In *NeurIPS Shared Visual Representations in Human and Machine Intelligence Workshop* (2019).

**The case studies presented in this article highlight the difficulty of interpreting machine behaviour in the presence of shortcut learning.**

- [79] Buckner, C. The Comparative Psychology of Artificial Intelligences (2019). URL <http://philsci-archive.pitt.edu/16034/>.

**This opinionated article points out important caveats when comparing human to machine intelligence.**

- [80] Morgan, C. L. Introduction to Comparative Psychology. (rev. ed.). *New York: Scribner* (1903).
- [81] Hays, J. & Efros, A. A. Scene completion using millions of photographs. *ACM Transactions on Graphics (TOG)* **26**, 4 (2007).
- [82] Hays, J. & Efros, A. A. IM2GPS: estimating geographic information from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1–8 (IEEE, 2008).
- [83] Jasani, B., Girdhar, R. & Ramanan, D. Are we asking the right questions in MovieQA? In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 0–0 (2019).
- [84] Cichy, R. M. & Kaiser, D. Deep neural networks as scientific models. *Trends in Cognitive Sciences* (2019).
- [85] Ghahramani, Z. Panel of workshop on advances in Approximate Bayesian Inference (AABI) 2017 (2017). URL <https://www.youtube.com/watch?v=x1UByHT60mQ&feature=youtu.be&t=37m44s>.

- [86] Marton, F. & Säljö, R. On qualitative differences in learning—II Outcome as a function of the learner’s conception of the task. *British Journal of Educational Psychology* **46**, 115–127 (1976).
- [87] Biggs, J. Individual differences in study processes and the quality of learning outcomes. *Higher Education* **8**, 381–394 (1979).
- [88] Chin, C. & Brown, D. E. Learning in science: A comparison of deep and surface approaches. *Journal of Research in Science Teaching* **37**, 109–138 (2000).
- This article from the field of Education reflects upon ways to achieve a better overlap between educational objectives and the way students learn.**
- [89] Marcus, G. F. Rethinking eliminative connectionism. *Cognitive Psychology* **37**, 243–282 (1998).
- [90] Kilbertus, N., Parascandolo, G. & Schölkopf, B. Generalization in anti-causal learning. *arXiv preprint arXiv:1812.00524* (2018).
- [91] Marcus, G. Deep learning: A critical appraisal. *arXiv preprint arXiv:1801.00631* (2018).
- [92] Lapuschkin, S. *et al.* Unmasking Clever Hans predictors and assessing what machines really learn. *Nature Communications* **10**, 1096 (2019).
- This study highlights how shortcut learning can lead to deceptively good results on standard metrics.**
- [93] Lake, B. M., Ullman, T. D., Tenenbaum, J. B. & Gershman, S. J. Building machines that learn and think like people. *arXiv preprint arXiv:1604.00289* (2016).
- [94] Chollet, F. The measure of intelligence. *arXiv preprint arXiv:1911.01547* (2019).
- [95] Crosby, M., Beyret, B. & Halina, M. The Animal-AI Olympics. *Nature Machine Intelligence* **1**, 257–257 (2019).
- [96] Juliani, A. *et al.* Obstacle tower: A generalization challenge in vision, control, and planning. *arXiv preprint arXiv:1902.01378* (2019).
- [97] Engstrom, L. *et al.* A discussion of ‘adversarial examples are not bugs, they are features’. *Distill* (2019).
- [98] Barbu, A. *et al.* ObjectNet: a large-scale bias-controlled dataset for pushing the limits of object recognition models. In *Advances in Neural Information Processing Systems*, 9448–9458 (2019).
- [99] Li, D., Yang, Y., Song, Y.-Z. & Hospedales, T. M. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, 5542–5550 (2017).
- [100] Creager, E. *et al.* Flexibly fair representation learning by disentanglement. *arXiv preprint arXiv:1906.02589* (2019).
- [101] Recht, B., Roelofs, R., Schmidt, L. & Shankar, V. Do CIFAR-10 classifiers generalize to CIFAR-10? *arXiv preprint arXiv:1806.00451* (2018).
- [102] Recht, B., Roelofs, R., Schmidt, L. & Shankar, V. Do ImageNet classifiers generalize to ImageNet? *arXiv preprint arXiv:1902.10811* (2019).
- [103] Levesque, H., Davis, E. & Morgenstern, L. The Winograd Schema Challenge. In *Thirteenth International Conference on the Principles of Knowledge Representation and Reasoning* (2012).

- [104] Trichelair, P., Emami, A., Trischler, A., Suleman, K. & Cheung, J. C. K. How reasonable are common-sense reasoning tasks: A case-study on the Winograd Schema Challenge and SWAG. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 3373–3378 (2019).
- [105] Hornik, K., Stinchcombe, M. & White, H. Multilayer feedforward networks are universal approximators. *Neural Networks* **2**, 359–366 (1989).
- [106] d’Ascoli, S., Sagun, L., Bruna, J. & Biroli, G. Finding the needle in the haystack with convolutions: On the benefits of architectural bias. *arXiv preprint arXiv:1906.06766* (2019).
- [107] Ulyanov, D., Vedaldi, A. & Lempitsky, V. Deep image prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 9446–9454 (2018).
- [108] Vaswani, A. *et al.* Attention is all you need. In *Advances in Neural Information Processing Systems*, 5998–6008 (2017).
- [109] Hein, M., Andriushchenko, M. & Bitterwolf, J. Why ReLU networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 41–50 (2019).
- [110] Madry, A., Makelov, A., Schmidt, L., Tsipras, D. & Vladu, A. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations* (2018).
- [111] Arjovsky, M., Bottou, L., Gulrajani, I. & Lopez-Paz, D. Invariant risk minimization. *arXiv preprint arXiv:1907.02893* (2019).
- [112] Wu, L., Zhu, Z. & E, W. Towards understanding generalization of deep learning: Perspective of loss landscapes. *arXiv preprint arXiv:1706.10239* (2017).
- [113] De Palma, G., Kiani, B. T. & Lloyd, S. Deep neural networks are biased towards simple functions. *arXiv preprint arXiv:1812.10156* (2018).
- [114] Valle-Perez, G., Camargo, C. Q. & Louis, A. A. Deep learning generalizes because the parameter-function map is biased towards simple functions. In *International Conference on Learning Representations* (2019).
- [115] Sun, K. & Nielsen, F. Lightlike neuromanifolds, Occam’s Razor and deep learning. *arXiv preprint arXiv:1905.11027* (2019).
- [116] Arpit, D. *et al.* A closer look at memorization in deep networks. In *International Conference on Machine Learning* (2017).
- [117] Li, Y., Wei, C. & Ma, T. Towards explaining the regularization effect of initial large learning rate in training neural networks. *arXiv preprint arXiv:1907.04595* (2019).
- [118] Bartlett, P. L., Long, P. M., Lugosi, G. & Tsigler, A. Benign overfitting in linear regression. *arXiv preprint arXiv:1906.11300* (2019).
- [119] Zipf, G. K. *Human Behavior and the Principle of Least Effort* (Addison-Wesley press, 1949).
- [120] Ohala, J. J. The phonetics and phonology of aspects of assimilation. *Papers in Laboratory Phonology* **1**, 258–275 (1990).
- [121] Vicentini, A. The economy principle in language. *Notes and Observations from early modern English grammars. Mots, Palabras, Words* **3**, 37–57 (2003).

- [122] Sinz, F. H., Pitkow, X., Reimer, J., Bethge, M. & Tolias, A. S. Engineering a less artificial intelligence. *Neuron* **103**, 967–979 (2019).
- [123] Cohen, T. & Welling, M. Group equivariant convolutional networks. In *International Conference on Machine Learning*, 2990–2999 (2016).
- [124] Cubuk, E. D., Zoph, B., Mane, D., Vasudevan, V. & Le, Q. V. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 113–123 (2019).
- [125] Berthelot, D. *et al.* Mixmatch: A holistic approach to semi-supervised learning. *arXiv preprint arXiv:1905.02249* (2019).
- [126] Hjelm, R. D. *et al.* Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670* (2018).
- [127] Oord, A. v. d., Li, Y. & Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748* (2018).
- [128] Schölkopf, B. Causality for machine learning. *arXiv preprint arXiv:1911.10500* (2019).
- [129] Schott, L., Rauber, J., Bethge, M. & Brendel, W. Towards the first adversarially robust neural network model on MNIST. In *International Conference on Learning Representations* (2019).
- [130] Engstrom, L. *et al.* Learning perceptually-aligned representations via adversarial robustness. *arXiv:1906.00945* (2019).
- [131] Peters, J., Bühlmann, P. & Meinshausen, N. Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **78**, 947–1012 (2016).
- [132] Dwork, C., Hardt, M., Pitassi, T., Reingold, O. & Zemel, R. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, 214–226 (2012).
- [133] Zemel, R., Wu, Y., Swersky, K., Pitassi, T. & Dwork, C. Learning fair representations. In *International Conference on Machine Learning*, 325–333 (2013).
- [134] Hardt, M., Price, E. & Srebro, N. Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems*, 3315–3323 (2016).
- [135] Kusner, M. J., Loftus, J., Russell, C. & Silva, R. Counterfactual fairness. In *Advances in Neural Information Processing Systems*, 4066–4076 (2017).
- [136] Schmidhuber, J. Evolutionary principles in self-referential learning. *On learning how to learn: The meta-meta-... hook.) Diploma thesis, Institut f. Informatik, Tech. Univ. Munich* **1**, 2 (1987).
- [137] Santoro, A., Bartunov, S., Botvinick, M., Wierstra, D. & Lillicrap, T. Meta-learning with memory-augmented neural networks. In *International Conference on Machine Learning*, 1842–1850 (2016).
- [138] Finn, C., Abbeel, P. & Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning* (2017).
- [139] Bengio, Y. *et al.* A meta-transfer objective for learning to disentangle causal mechanisms. *arXiv preprint arXiv:1901.10912* (2019).

- [140] Fetaya, E., Jacobsen, J.-H., Grathwohl, W. & Zemel, R. Understanding the limitations of conditional generative models. In *International Conference on Learning Representations* (2020).
- [141] Higgins, I. *et al.* Beta-VAE: Learning basic visual concepts with a constrained variational framework. *International Conference on Learning Representations* (2017).
- [142] Hyvärinen, A. & Oja, E. Independent component analysis: algorithms and applications. *Neural Networks* **13**, 411–430 (2000).
- [143] Richardson, J. *Vectors: aphorisms & ten-second essays* (Ausable Press, 2001).

## Appendix

### A Toy example: method details

The code to reproduce our toy example (Figure 2) is available from <https://github.com/rgeirhos/shortcut-perspective>. Two easily distinguishable shapes (star and moon) were placed on a  $200 \times 200$  dimensional 2D canvas. The training set is constructed out of 4000 images, where 2000 contain a star shape and 2000 a moon shape. The star shape is randomly placed in the top right and bottom left quarters of the canvas, whereas the moon shape is randomly placed in the top left and bottom right quarters of the canvas. At test time the setup is nearly identical, 1000 images with a star and 1000 images with a moon are presented. However, this time the position of star and moon shapes are randomised over the full canvas, i.e. in test images stars and moons can appear at any location.

We train two classifiers on this dataset: a fully connected network as well as a convolutional network. The classifiers are trained for five epochs with a batch size of 100 on the training set and evaluated on the test set. The training objective is standard crossentropy loss and the optimizer is Adam with a learning rate of 0.00001,  $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$  and  $\epsilon = 1e - 08$ . The fully connected network was a three-layer ReLU MLP (multilayer perceptron) with 1024 units in each layer and two output units corresponding to the two target classes. It reaches 100% accuracy at training time and approximately chance-level accuracy at test time (51.0%). The convolutional network had three convolutional layers with 128 channels, a stride of 2 and filter size of  $5 \times 5$  interleaved with ReLU nonlinearities, followed by a global average pooling and a linear layer mapping the 128 outputs to the logits. It reaches 100% accuracy on train and test set.

### B Image rights & attribution

Figure 1 consists of four images from different sources. The first image from the left was taken from <https://aiweirdness.com/post/171451900302/do-neural-nets-dream-of-electric-sheep> with permission of the author. The second image from the left was generated by ourselves. The third image from the left is from ref. [15]. It was released under the CC BY 4.0 license as stated here: <https://journals.plos.org/plosmedicine/article?id=10.1371/journal.pmed.1002683> and adapted by us from Figure 2B of the corresponding publication. The image on the right is Figure 1 from ref. [64]. It was released under CC BY 4.0 license as stated here: <https://www.aclweb.org/anthology/D17-1215/> (at the bottom) and retrieved by us from .

The image from Section 4.1 was adapted from Figure 1 of ref. [9] with permission from the authors (image cropped from original figure by us). The image from Section 4.2 was adapted from Figure 1 of ref. [38] with permission from the authors (image cropped from original figure by us). The image from Section 4.3 was adapted from Figure 1 of ref. [45] with permission from the authors (image cropped from original figure by us).

Figure 4 consists of a number of images from different sources. The first author of the corresponding publication is mentioned in the figure for identification. The images from ref. [8] were released under the CC BY 3.0 license as stated here: <https://arxiv.org/abs/1312.6199> and adapted by us from Figure 5a of the corresponding publication (images cropped from original figure by us). The images from ref. [50] were adapted from Figure 1 of the corresponding paper with permission from the authors (images cropped from original figure by us). The images from ref. [49] were adapted from Figure 1 of the corresponding paper with permission from the authors (images cropped from original figure by us). The images from ref. [38] were adapted from Figure 1 of the corresponding paper with permission from the authors (images cropped from original figure by us). The images from ref. [41] were adapted from Figure 1 of the corresponding paper with permission from the authors (images cropped from original figure by us). The images from ref. [36] were adapted from Figure 5 of the corresponding paper with permission from the authors (images cropped from original figure by us). The images from ref. [9] were adapted from Figure 1 of the

corresponding paper with permission from the authors (images cropped from original figure by us).  
The images from ref. [45] were adapted from Figure 1 and Figure 2 of the corresponding paper  
with permission from the authors (images cropped from original figures by us).



## 2.6 Partial success in closing the gap between human and machine vision

*Transparency notice* A much less comprehensive version of this work was presented as “Oral” at the 2020 NeurIPS workshop on “Shared Visual Representations in Human & Machine Intelligence” (Geirhos et al., 2020c).

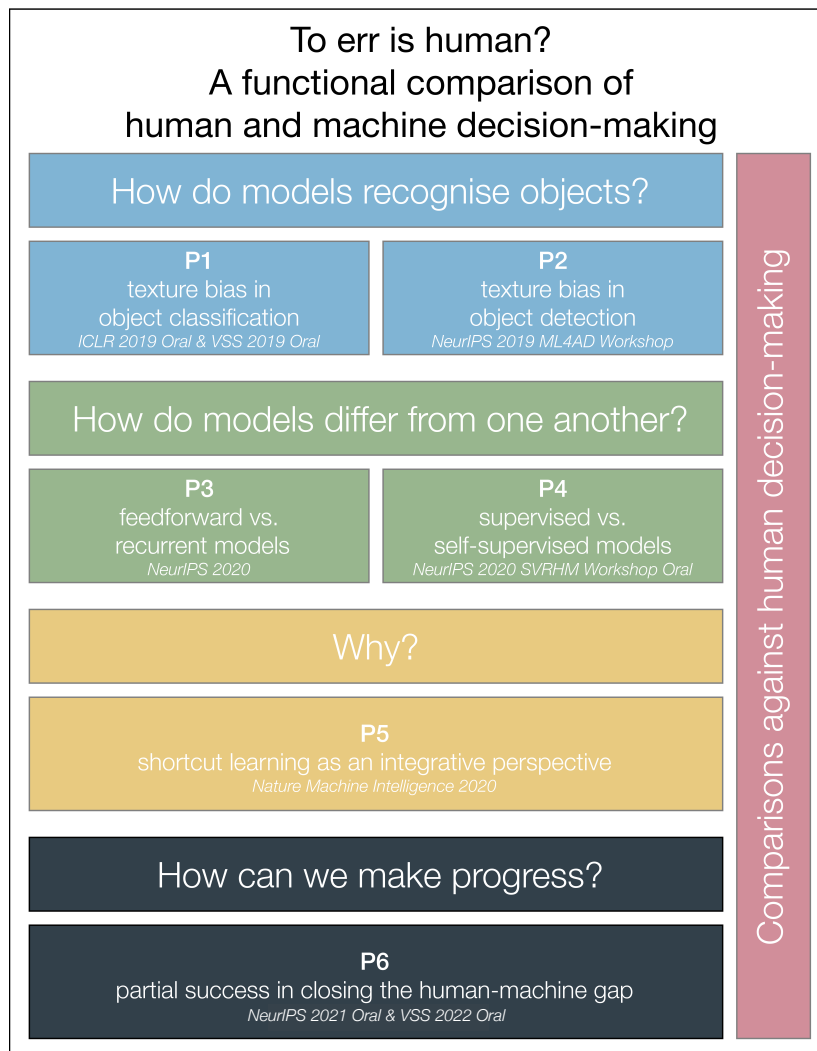


Figure 2.6: Schematic overview of the projects presented in this thesis. Projects P1–P4 ask “How do machines decide”, project P5 presents the concept of shortcut learning as an integrative perspective tackling the question “Why do machines decide the way they do”. Finally, project P6 presents a comprehensive benchmark to measure future progress, and reports first signs of (partial) success. Throughout the thesis, machine decision making will be compared against human decision making.

---

## Partial success in closing the gap between human and machine vision

---

Robert Geirhos<sup>1,2§</sup>    Kantharaju Narayanappa<sup>1</sup>    Benjamin Mitzkus<sup>1</sup>

Tizian Thieringer<sup>1</sup>    Matthias Bethge<sup>1\*</sup>    Felix A. Wichmann<sup>1\*</sup>    Wieland Brendel<sup>1\*</sup>

<sup>1</sup>University of Tübingen

<sup>2</sup>International Max Planck Research School for Intelligent Systems

\*Joint senior authors

§To whom correspondence should be addressed: robert.geirhos@uni-tuebingen.de

### Abstract

A few years ago, the first CNN surpassed human performance on ImageNet. However, it soon became clear that machines lack robustness on more challenging test cases, a major obstacle towards deploying machines “in the wild” and towards obtaining better computational models of human visual perception. Here we ask: Are we making progress in closing the gap between human and machine vision? To answer this question, we tested human observers on a broad range of out-of-distribution (OOD) datasets, recording 85,120 psychophysical trials across 90 participants. We then investigated a range of promising machine learning developments that crucially deviate from standard supervised CNNs along three axes: objective function (self-supervised, adversarially trained, CLIP language-image training), architecture (e.g. vision transformers), and dataset size (ranging from 1M to 1B).

Our findings are threefold. (1.) The longstanding *distortion robustness gap* between humans and CNNs is closing, with the best models now exceeding human feedforward performance on most of the investigated OOD datasets. (2.) There is still a substantial image-level *consistency gap*, meaning that humans make different errors than models. In contrast, most models systematically agree in their categorisation errors, even substantially different ones like contrastive self-supervised vs. standard supervised models. (3.) In many cases, human-to-model consistency improves when training dataset size is increased by one to three orders of magnitude. Our results give reason for cautious optimism: While there is still much room for improvement, the behavioural difference between human and machine vision is narrowing. In order to measure future progress, 17 OOD datasets with image-level human behavioural data and evaluation code are provided as a toolbox and benchmark at <https://github.com/bethgelab/model-vs-human/>.

### 1 Introduction

Looking back at the last decade, deep learning has made tremendous leaps of progress by any standard. What started in 2012 with AlexNet [1] as the surprise winner of the ImageNet Large-Scale Visual Recognition Challenge quickly became the birth of a new AI “summer”, a summer lasting much longer than just a season. With it, just like with any summer, came great expectations: the hope that the deep learning revolution will see widespread applications in industry, that it will propel breakthroughs in the sciences, and that it will ultimately close the gap between human and machine

perception. We have now reached the point where deep learning has indeed become a significant driver of progress in industry [e.g. 2, 3], and where many disciplines are employing deep learning for scientific discoveries [4–9]—*but are we making progress in closing the gap between human and machine vision?*

**IID vs. OOD benchmarking.** For a long time, the gap between human and machine vision was mainly approximated by comparing benchmark accuracies on IID (independent and identically distributed) test data: as long as models are far from reaching human-level performance on challenging datasets like ImageNet, this approach is adequate [10]. Currently, models are routinely matching and in many cases even outperforming humans on IID data. At the same time, it is becoming increasingly clear that models systematically exploit shortcuts shared between training and test data [11–14]. Therefore we are witnessing a major shift towards measuring model performance on out-of-distribution (OOD) data rather than IID data alone, which aims at testing models on more challenging test cases where there is still a ground truth category, but certain image statistics differ from the training distribution. Many OOD generalisation tests have been proposed: ImageNet-C [15] for corrupted images, ImageNet-Sketch [16] for sketches, Stylized-ImageNet [17] for image style changes, [18] for unfamiliar object poses, and many more [19–29]. While it is great to have many viable and valuable options to measure generalisation, most of these datasets unfortunately lack human comparison data. This is less than ideal, since we can no longer assume that humans reach near-ceiling accuracies on these challenging test cases as they do on standard noise-free IID object recognition datasets. In order to address this issue, we carefully tested human observers in the Wichmannlab’s vision laboratory on a broad range of OOD datasets, providing some 85K psychophysical trials across 90 participants. Crucially, we showed exactly the same images to multiple observers, which means that we are able to compare human and machine vision on the fine-grained level of individual images [30–32]). The focus of our datasets is measuring *distortion robustness*: we tested 17 variations that include changes to image style, texture, and various forms of synthetic additive noise.

**Contributions & outlook.** The resulting 17 OOD datasets with large-scale human comparison data enable us to investigate recent exciting machine learning developments that crucially deviate from “vanilla” CNNs along three axes: objective function (supervised vs. self-supervised, adversarially trained, and CLIP’s joint language-image training), architecture (convolutional vs. vision transformer) and training dataset size (ranging from 1M to 1B images). Taken together, these are some of the most promising directions our field has developed to date—but this field would not be machine learning if new breakthroughs weren’t within reach in the next few weeks, months and years. Therefore, we open-sourced `modelvshuman`, a Python toolbox that enables testing both PyTorch and TensorFlow models on our comprehensive benchmark suite of OOD generalisation data in order to measure future progress. Even today, our results give cause for (cautious) optimism. After a method overview (Section 2), we are able to report that the human-machine *distortion robustness gap* is closing: the best models now match or in many cases even exceed human feedforward performance on most of the investigated OOD datasets (Section 3). While there is still a substantial image-level *consistency gap* between humans and machines, this gap is narrowing on some—but not all—datasets when the size of the training dataset is increased (Section 4).

## 2 Methods: datasets, psychophysical experiments, models, metrics, toolbox

**OOD datasets with consistency-grade human data.** We collected human data for 17 generalisation datasets (visualized in Figures 7 and 8 in the Appendix, which also state the number of subjects and trials per experiment) on a carefully calibrated screen in a dedicated psychophysical laboratory (a total of 85,120 trials across 90 observers). Five datasets each correspond to a single manipulation (sketches, edge-filtered images, silhouettes, images with a texture-shape cue conflict, and stylized images where the original image texture is replaced by the style of a painting); the remaining twelve datasets correspond to parametric image degradations (e.g. different levels of noise or blur). Those OOD datasets have in common that they are designed to test ImageNet-trained models. OOD images were obtained from different sources: sketches from ImageNet-Sketch [16], stylized images from

Stylized-ImageNet [17], edge-filtered images, silhouettes and cue conflict images from [17]<sup>1</sup>, and the remaining twelve parametric datasets were adapted from [33]. For these parametric datasets, [33] collected human accuracies but unfortunately, they showed different images to different observers implying that we cannot use their human data to assess image-level consistency between humans and machines. Thus we collected psychophysical data for those images ourselves by showing exactly the same images to multiple observers for each of those twelve datasets. Additionally, we cropped the images from [33] to  $224 \times 224$  pixels to allow for a fair comparison to ImageNet models (all models included in our comparison receive  $224 \times 224$  input images; [33] showed  $256 \times 256$  images to human observers in many cases).

**Psychophysical experiments.** 90 observers were tested in a darkened chamber. Stimuli were presented at the center of a 22" monitor with  $1920 \times 1200$  pixels resolution (refresh rate: 120 Hz). Viewing distance was 107 cm and target images subtended  $3 \times 3$  degrees of visual angle. Human observers were presented with an image and asked to select the correct category out of 16 basic categories (such as chair, dog, airplane, etc.). Stimuli were balanced w.r.t. classes and presented in random order. For ImageNet-trained models, in order to obtain a choice from the same 16 categories, the 1,000 class decision vector was mapped to those 16 classes using the WordNet hierarchy [34]. In Appendix I, we explain why this mapping is optimal. We closely followed the experimental protocol defined by [33], who presented images for 200 ms followed by a  $1/f$  backward mask to limit the influence of recurrent processing (otherwise comparing to feedforward models would be difficult). Further experimental details are provided in Appendix C.

**Why not use crowdsourcing instead?** Our approach of investigating few observers in a high-quality laboratory setting performing many trials is known as the so-called "small-N design", the bread-and-butter approach in high-quality psychophysics—see, e.g., the review "Small is beautiful: In defense of the small-N design" [35]. This is in contrast to the "crowdsourcing approach" (many observers in a noisy setting performing fewer trials each). The highly controlled conditions of the Wichmannlab's psychophysical laboratory come with many advantages over crowdsourced data collection: precise timing control (down to the millisecond), carefully calibrated monitors (especially important for e.g. low-contrast stimuli), controlled viewing distance (important for foveal presentation), full visual acuity (we performed an acuity test with every observer prior to the experiment), observer attention (e.g. no multitasking or children running around during an experiment, which may happen in a crowdsourcing study), just to name a few [36]. Jointly, these factors contribute to high data quality.

**Models.** In order to disentangle the influence of objective function, architecture and training dataset size, we tested a total of 52 models: 24 standard ImageNet-trained CNNs [37], 8 self-supervised models [38–43],<sup>2</sup> 6 Big Transfer models [45], 5 adversarially trained models [46], 5 vision transformers [47, 48], two semi-weakly supervised models [49] as well as Noisy Student [50] and CLIP [51]. Technical details for all models are provided in the Appendix.

**Metrics.** In addition to *OOD accuracy* (averaged across conditions and datasets), the following three metrics quantify how closely machines are aligned with the decision behaviour of humans.

*Accuracy difference*  $A(m)$  is a simple aggregate measure that compares the accuracy of a machine  $m$  to the accuracy of human observers in different out-of-distribution tests,

$$A(m) : \mathbb{R} \rightarrow [0, 1], m \mapsto \frac{1}{|D|} \sum_{d \in D} \frac{1}{|H_d|} \sum_{h \in H_d} \frac{1}{|C_d|} \sum_{c \in C_d} (\text{acc}_{d,c}(h) - \text{acc}_{d,c}(m))^2 \quad (1)$$

where  $\text{acc}_{d,c}(\cdot)$  is the accuracy of the model or the human on dataset  $d \in D$  and condition  $c \in C_d$  (e.g. a particular noise level), and  $h \in H_d$  denotes a human observer tested on dataset  $d$ . Analogously, one can compute the average accuracy difference between a human observer  $h_1$  and all other human observers by substituting  $h_1$  for  $m$  and  $h \in H_d \setminus \{h_1\}$  for  $h \in H_d$  (which can also be applied for the two metrics defined below).

<sup>1</sup>For those three datasets consisting of 160, 160 and 1280 images respectively, consistency-grade psychophysical data was already collected by the authors and included in our benchmark with permission from the authors.

<sup>2</sup>We presented a preliminary and much less comprehensive version of this work at the NeurIPS 2020 workshop SVRHM [44].

Aggregated metrics like  $A(m)$  ignore individual image-level decisions. Two models with vastly different image-level decision behaviour might still end up with the same accuracies on each dataset and condition. Hence, we include two additional metrics in our benchmark that are sensitive to decisions on individual images.

*Observed consistency*  $O(m)$  [32] measures the fraction of samples for which humans and a model  $m$  get the same sample either both right or both wrong. More precisely, let  $b_{h,m}(s)$  be one if both a human observer  $h$  and  $m$  decide either correctly or incorrectly on a given sample  $s$ , and zero otherwise. We calculate the average observed consistency as

$$O(m) : \mathbb{R} \rightarrow [0, 1], m \mapsto \frac{1}{|D|} \sum_{d \in D} \frac{1}{|H_d|} \sum_{h \in H_d} \frac{1}{|C_d|} \sum_{c \in C_d} \frac{1}{|S_{d,c}|} \sum_{s \in S_{d,c}} b_{h,m}(s) \quad (2)$$

where  $s \in S_{d,c}$  denotes a sample  $s$  (in our case, an image) of condition  $c$  from dataset  $d$ . Note that this measure can only be zero if the accuracy of  $h$  and  $m$  are exactly the same in each dataset and condition.

*Error consistency*  $E(m)$  [32] tracks whether there is above-chance consistency. This is an important distinction, since e.g. two decision makers with 95% accuracy each will have at least 90% observed consistency, even if their 5% errors occur on non-overlapping subsets of the test data (intuitively, they both get most images correct and thus observed overlap is high). To this end, error consistency (a.k.a. Cohen’s kappa, cf. [52]) indicates whether the observed consistency is larger than what could have been expected given two independent binomial decision makers with matched accuracy, which we denote as  $\hat{o}_{h,m}$ . This can easily be computed analytically [e.g. 32, equation 1]. Then, the average error consistency is given by

$$E(m) : \mathbb{R} \rightarrow [-1, 1], m \mapsto \frac{1}{|D|} \sum_{d \in D} \frac{1}{|H_d|} \sum_{h \in H_d} \frac{1}{|C_d|} \sum_{c \in C_d} \frac{(\frac{1}{|S_{d,c}|} \sum_{s \in S_{d,c}} b_{h,m}(s)) - \hat{o}_{h,m}(S_{d,c})}{1 - \hat{o}_{h,m}(S_{d,c})} \quad (3)$$

**Benchmark & toolbox.**  $A(m)$ ,  $O(m)$  and  $E(m)$  each quantify a certain aspect of the human-machine gap. We use the mean rank order across these metrics to determine an overall model ranking (Table 2 in the Appendix). However, we would like to emphasise that the primary purpose of this benchmark is to generate insights, not winners. Since insights are best gained from detailed plots and analyses, we open-source `modelvshuman`, a Python project to benchmark models against human data.<sup>3</sup> The current model zoo already includes 50+ models, and an option to add new ones (both PyTorch and TensorFlow). Evaluating a model produces a 15+ page report on model behaviour. All plots in this paper can be generated for future models—to track whether they narrow the gap towards human vision, or to determine whether an algorithmic modification to a baseline model (e.g., an architectural improvement) changes model behaviour.

### 3 Robustness across models: the OOD distortion robustness gap between human and machine vision is closing

We are interested in measuring whether we are making progress in closing the gap between human and machine vision. For a long time, CNNs were unable to match human robustness in terms of generalisation beyond the training distribution—a large OOD *distortion robustness gap* [14, 33, 53–55]. Having tested human observers on 17 OOD datasets, we are now able to compare the latest developments in machine vision to human perception. Our core results are shown in Figure 1: the OOD distortion robustness gap between human and machine vision is closing (1a, 1b), especially for models trained on large-scale datasets. On the individual image level, a human-machine consistency gap remains (especially 1d), which will be discussed later.

**Self-supervised models** “If intelligence is a cake, the bulk of the cake is unsupervised learning, the icing on the cake is supervised learning and the cherry on the cake is reinforcement learning”,

<sup>3</sup>Of course, comparing human and machine vision is not limited to object recognition behaviour: other comparisons may be just as valid and interesting.

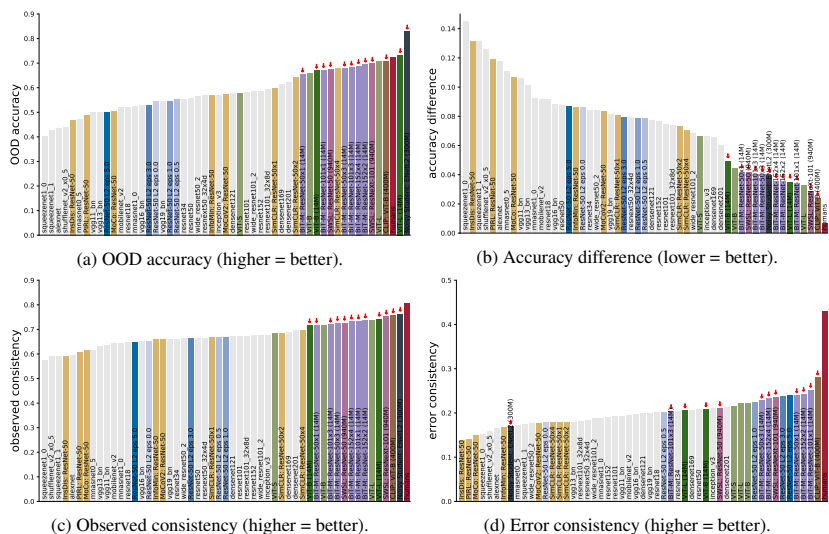


Figure 1: Core results, aggregated over 17 out-of-distribution (OOD) datasets: The OOD robustness gap between human and machine vision is closing (top), but an image-level consistency gap remains (bottom). Results compare humans, standard supervised CNNs, self-supervised models, adversarially trained models, vision transformers, noisy student, BiT, SWSL and CLIP. For convenience, ↓ marks models that are trained on large-scale datasets. Metrics defined in Section 2. Best viewed on screen.

Yann LeCun said in 2016 [56]. A few years later, the entire cake is finally on the table—the representations learned via self-supervised learning<sup>4</sup> now compete with supervised methods on ImageNet [43] and outperform supervised pre-training for object detection [41]. But how do recent self-supervised models differ from their supervised counterparts in terms of their behaviour? Do they bring machine vision closer to human vision? Humans, too, rapidly learn to recognise new objects without requiring hundreds of labels per instance; additionally a number of studies reported increased similarities between self-supervised models and human perception [57–61]. Figure 2 compares the generalisation behaviour of eight self-supervised models in orange (PIRL, MoCo, MoCoV2, InfoMin, InsDis, SimCLR-x1, SimCLR-x2, SimCLR-x4)—with 24 standard supervised models (grey). We find only marginal differences between self-supervised and supervised models: Across distortion types, self-supervised networks are well within the range of their poorly generalising supervised counterparts. However, there is one exception: the three SimCLR variants show strong generalisation improvements on uniform noise, low contrast, and high-pass images, where they are the three top-performing self-supervised networks—quite remarkable given that SimCLR models were trained on a different set of augmentations (random crop with flip and resize, colour distortion, and Gaussian blur). Curious by the outstanding performance of SimCLR, we asked whether the self-supervised objective function or the choice of training data augmentations was the defining factor. When comparing self-supervised SimCLR models with augmentation-matched baseline models trained in the standard supervised fashion (Figure 15 in the Appendix), we find that the augmentation scheme (rather than the self-supervised objective) indeed made the crucial difference: supervised baselines show just the same generalisation behaviour, a finding that fits well with [62], who observed that the influence of training data augmentations is stronger than the role of architecture or training objective. In conclusion, our analyses indicate that the “cake” of contrastive self-supervised learning currently (and disappointingly) tastes much like the “icing”.

<sup>4</sup>“Unsupervised learning” and “self-supervised learning” are sometimes used interchangeably. We use the term “self-supervised learning” since those methods use (label-free) supervision.

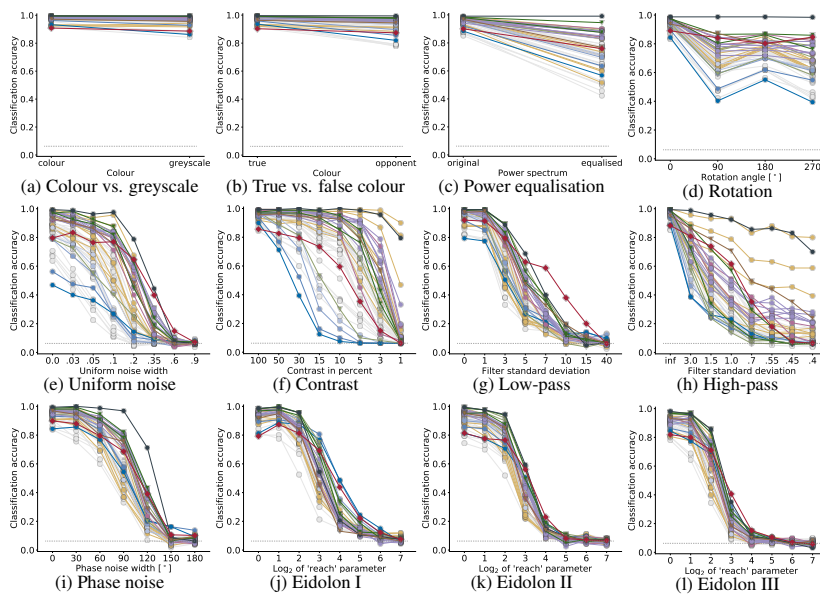


Figure 2: The OOD distortion robustness gap between human and machine vision is closing. Robustness towards parametric distortions for **humans**, standard supervised CNNs, **self-supervised models**, **adversarially trained models**, **vision transformers**, noisy student, BiT, SWSL, CLIP. Symbols indicate architecture type ( $\circ$  convolutional,  $\nabla$  vision transformer,  $\diamond$  human); best viewed on screen.

**Adversarially trained models** The vulnerability of CNNs to adversarial input perturbations is, arguably, one of the most striking shortcomings of this model class compared to robust human perception. A successful method to increase adversarial robustness is *adversarial training* [e.g. 63, 64]. The resulting models were found to transfer better, have meaningful gradients [65], and enable interpolating between two input images [66]: “robust optimization can actually be viewed as inducing a *human prior* over the features that models are able to learn” [67, p. 10]. Therefore, we include five models with a ResNet-50 architecture and different accuracy-robustness tradeoffs, adversarially trained on ImageNet with Microsoft-scale resources by [46] to test whether models with “perceptually-aligned representations” also show human-aligned OOD generalisation behaviour—as we would hope. This is not the case: the stronger the model is trained adversarially (darker shades of blue in Figure 2), the more susceptible it becomes to (random) image degradations. Most strikingly, a simple rotation by 90 degrees leads to a 50% drop in classification accuracy. Adversarial robustness seems to come at the cost of increased vulnerability to large-scale perturbations.<sup>5</sup> On the other hand, there is a silver lining: when testing whether models are biased towards texture or shape by testing them on cue conflict images (Figure 3), in accordance with [69, 70] we observe a perfect relationship between shape bias and the degree of adversarial training, a big step in the direction of human shape bias (and a stronger shape bias than nearly all other models).

**Vision transformers** In computer vision, convolutional networks have become by far the dominant model class over the last decade. Vision transformers [47] break with the long tradition of using convolutions and are rapidly gaining traction [71]. We find that the best vision transformer (ViT-L trained on 14M images) even *exceeds* human OOD accuracy (Figure 1a shows the average across 17 datasets). There appears to be an additive effect of architecture and data: vision transformers trained on 1M images (light green) are already better than standard convolutional models; training on 14M images (dark green) gives another performance boost. In line with [72, 73], we observe a higher shape bias compared to most standard CNNs.

<sup>5</sup>This might be related to [68], who studied a potentially related tradeoff between selectivity and invariance.

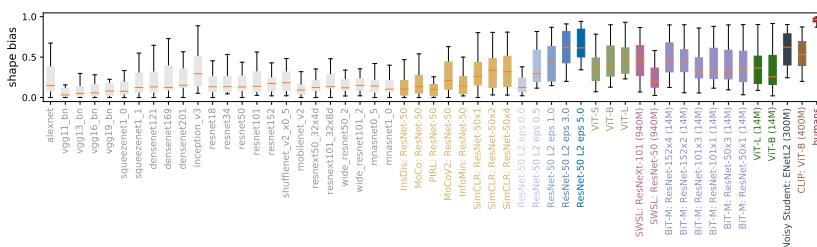


Figure 3: Shape vs. texture biases of different models. While human shape bias is not yet matched, several approaches improve over vanilla CNNs. Box plots show category-dependent distribution of shape / texture biases (shape bias: high values, texture bias: low values).

**Standard models trained on more data: BIT-M, SWSL, Noisy Student** Interestingly, the biggest effect on OOD robustness we find simply comes from training on larger datasets, not from advanced architectures. When standard models are combined with large-scale training (14M images for BIT-M, 300M for Noisy Student and a remarkable 940M for SWSL), OOD accuracies reach levels not known from standard ImageNet-trained models; these models even outperform a more powerful architecture (vision transformer ViT-S) trained on less data (1M) as shown in Figure 1a. Simply training on (substantially) more data substantially narrows the gap to human OOD accuracies (1b), a finding that we quantified in Appendix H by means of a regression model. (The regression model also revealed a significant interaction between dataset size and objective function, as well as a significant main effect for transformers over CNNs.) Noisy Student in particular outperforms humans by a large margin overall (Figure 1a)—the beginning of a new human-machine gap, this time in favour of machines?

**CLIP** CLIP is special: trained on 400M images<sup>6</sup> (more data) with joint language-image supervision (novel objective) and a vision transformer backbone (non-standard architecture), it scores close to humans across all of our metrics presented in Figure 1; most strikingly in terms of error consistency (which will be discussed in the next section). We tested a number of hypotheses to disentangle why CLIP appears “special”. *H1: because CLIP is trained on a lot of data?* Presumably no: Noisy Student—a model trained on a comparably large dataset of 300M images—performs very well on OOD accuracy, but poorly on error consistency. A caveat in this comparison is the quality of the labels: while Noisy Student uses pseudolabeling, CLIP receives web-based labels for all images. *H2: because CLIP receives higher-quality labels?* About 6% of ImageNet labels are plainly wrong [74]. Could it be the case that CLIP simply performs better since it doesn’t suffer from this issue? In order to test this, we used CLIP to generate new labels for all 1.3M ImageNet images: (a) hard labels, i.e. the top-1 class predicted by CLIP; and (b) soft labels, i.e. using CLIP’s full posterior distribution as a target. We then trained ResNet-50 from scratch on CLIP hard and soft labels (for details see Appendix E). However, this does not show any robustness improvements over a vanilla ImageNet-trained ResNet-50, thus different/better labels are not a likely root cause. *H3: because CLIP has a special image+text loss?* Yes and no: CLIP training on ResNet-50 leads to astonishingly poor OOD results, so training a standard model with CLIP loss alone is insufficient. However, while neither architecture nor loss alone sufficiently explain why CLIP is special, we find a clear interaction between architecture and loss (described in more detail in the Appendix along with the other “CLIP ablation” experiments mentioned above).

#### 4 Consistency between models: data-rich models narrow the substantial image-level consistency gap between human and machine vision

In the previous section we have seen that while self-supervised and adversarially trained models lack OOD distortion robustness, models based on vision transformers and/or trained on large-scale datasets now match or exceed human feedforward performance on most datasets. Behaviourally, a

<sup>6</sup>The boundary between IID and OOD data is blurry for networks trained on big proprietary datasets. We consider it unlikely that CLIP was exposed to many of the exact distortions used here (e.g. eidolon or cue conflict images), but CLIP likely had greater exposure to some conditions such as grayscale or low-contrast images.



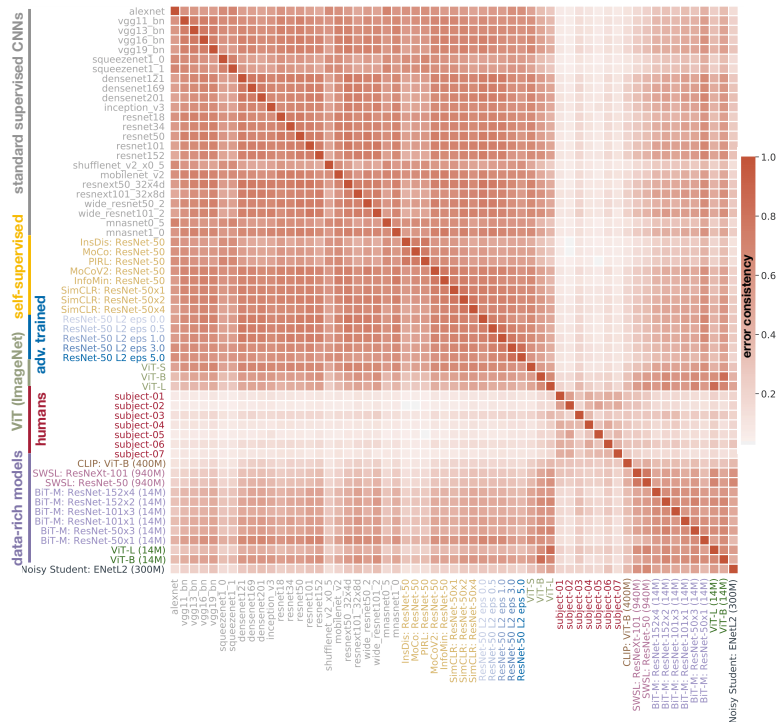


Figure 4: Data-rich models narrow the substantial image-level consistency gap between humans and machines. Error consistency analysis on a single dataset (sketch images; for other datasets see Appendix, Figures 9, 11, 12, 13, 14) shows that most models cluster (dark red = highly consistent errors) irrespective of their architecture and objective function; humans cluster differently (high human-to-human consistency, low human-to-model consistency); but some data-rich models including CLIP and SWSL blur the boundary, making more human-like errors than standard models.

natural follow-up question is to ask not just how many, but *which* errors models make—i.e., do they make errors on the same individual images as humans on OOD data (an important characteristic of a “human-like” model, cf. [32, 75])? This is quantified via *error consistency* (defined in Section 2); which additionally allows us to compare models with each other, asking e.g. which model classes make similar errors. In Figure 4, we compare all models with each other and with humans, asking whether they make errors on the same images. On this particular dataset (sketch images), we can see one big model cluster. Irrespective of whether one takes a standard supervised model, a self-supervised model, an adversarially trained model or a vision transformer, all those models make highly systematic errors (which extends the results of [32, 76] who found similarities between standard vanilla CNNs). Humans, on the other hand, show a very different pattern of errors. Interestingly, the boundary between humans and some data-rich models at the bottom of the figure—especially CLIP (400M images) and SWSL (940M)—is blurry: some (but not all) data-rich models much more closely mirror the patterns of errors that humans make, and we identified the first models to achieve higher error consistency with humans than with other (standard) models. Are these promising results shared across datasets, beyond the sketch images? In Figures 1c and 1d, aggregated results over 17 datasets are presented. Here, we can see that data-rich models approach human-to-human observed consistency, but not error consistency. Taken in isolation, *observed* consistency is not a good measure of image-level consistency since it does not take consistency by chance into account; *error* consistency tracks whether there is consistency beyond chance; here we see that there is still

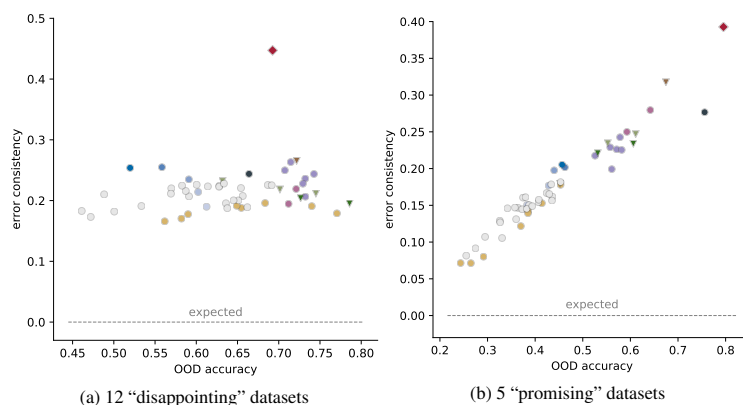


Figure 5: Partial failure, partial success: Error consistency with humans aggregated over multiple datasets. Left: 12 datasets where model accuracies exceed human accuracies; here, there is still a substantial image-level consistency gap to humans. Right: 5 datasets (sketch, silhouette, edge, cue conflict, low-pass) where humans are more robust. Here, OOD accuracy is a near-perfect predictor of image-level consistency; especially data-rich models (e.g. CLIP, SWSL, BiT) narrow the consistency gap to humans. Symbols indicate architecture type ( $\circ$  convolutional,  $\nabla$  vision transformer,  $\diamond$  human).

a substantial image-level *consistency gap* between human and machine vision. However, several models improve over vanilla CNNs, especially BiT-M (trained on 14M images) and CLIP (400M images). This progress is non-trivial; at the same time, there is ample room for future improvement.

How do the findings from Figure 4 (showing nearly human-level error consistency for sketch images) and from Figure 1d (showing a substantial consistency gap when aggregating over 17 datasets) fit together? Upon closer inspection, we discovered that there are two distinct cases. On 12 datasets (stylized, colour/greyscale, contrast, high-pass, phase-scrambling, power-equalisation, false colour, rotation, eidolonI, -II and -III as well as uniform noise), the human-machine gap is large; here, more robust models do not show improved error consistency (as can be seen in Figure 5a). On the other hand, for five datasets (sketch, silhouette, edge, cue conflict, low-pass filtering), there is a completely different result pattern: Here, OOD accuracy is a near-perfect predictor of error consistency, which means that improved generalisation robustness leads to more human-like errors (Figure 5b). Furthermore, training on large-scale datasets leads to considerable improvements along both axes for standard CNNs. Within models trained on larger datasets, CLIP scores best; but models with a standard architecture (SWSL: based on ResNet-50 and ResNeXt-101) closely follow suit.

It remains an open question why the training dataset appears to have the most important impact on a model’s decision boundary as measured by error consistency (as opposed to other aspects of a model’s inductive bias). Datasets contain various shortcut opportunities [14], and if two different models are trained on similar data, they might converge to a similar solution simply by exploiting the same shortcuts—which would also fit well to the finding that adversarial examples typically transfer very well between different models [77, 78]. Making models more flexible (such as transformers, a generalisation of CNNs) wouldn’t change much in this regard, since flexible models can still exploit the same shortcuts. Two predictions immediately follow from this hypothesis: (1.) error consistency between two identical models trained on very different datasets, such as ImageNet vs. Stylized-ImageNet, is much lower than error consistency between very different models (ResNet-50 vs. VGG-16) trained on the same dataset. (2.) error consistency between ResNet-50 and a highly flexible model (e.g., a vision transformer) is much higher than error consistency between ResNet-50 and a highly constrained model like BagNet-9 [79]. We provide evidence for both predictions in Appendix B, which makes the shortcut hypothesis of model similarity a potential starting point for future analyses. Looking forward, it may be worth exploring the links between shortcut learning and image difficulty, such as understanding whether many “trivially easy” images in common datasets like ImageNet causes models to exploit the same characteristics irrespective of their architecture [80].

## 5 Discussion

**Summary** We set out to answer the question: *Are we making progress in closing the gap between human and machine vision?* In order to quantify progress, we performed large-scale psychophysical experiments on 17 out-of-distribution distortion datasets (open-sourced along with evaluation code as a benchmark to track future progress). We then investigated models that push the boundaries of traditional deep learning (different objective functions, architectures, and dataset sizes ranging from 1M to 1B), asking how they perform relative to human visual perception. We found that the OOD distortion robustness gap between human and machine vision is closing, as the best models now match or exceed human accuracies. At the same time, an image-level consistency gap remains; however, this gap that is at least in some cases narrowing for models trained on large-scale datasets.

**Limitations** Model robustness is studied from many different viewpoints, including adversarial robustness [77], theoretical robustness guarantees [e.g. 81], or label noise robustness [e.g. 82]. The focus of our study is robustness towards non-adversarial out-of-distribution data, which is particularly well-suited for comparisons with humans. Since we aimed at a maximally fair comparison between feedforward models and human perception, presentation times for human observers were limited to 200 ms in order to limit the influence of recurrent processing. Therefore, human ceiling performance might be higher still (given more time); investigating this would mean going beyond “core object recognition”, which happens within less than 200 ms during a single fixation [83]. Furthermore, human and machine vision can be compared in many different ways. This includes comparing against neural data [84, 85], contrasting Gestalt effects [e.g. 86], object similarity judgments [87], or mid-level properties [61] and is of course not limited to studying object recognition. By no means do we mean to imply that our behavioural comparison is the only feasible option—on the contrary, we believe it will be all the more exciting to investigate whether our behavioural findings have implications for other means of comparison!

**Discussion** We have to admit that we view our results concerning the benefits of increasing dataset size by one-to-three orders of magnitude with mixed feelings. On the one hand, “simply” training standard models on (a lot) more data certainly has an intellectually disappointing element—particularly given many rich ideas in the cognitive science and neuroscience literature on which architectural changes might be required to bring machine vision closer to human vision [88–93]. Additionally, large-scale training comes with infrastructure demands that are hard to meet for many academic researchers. On the other hand, we find it truly exciting to see that machine models are closing not just the OOD distortion robustness gap to humans, but that also, at least for some datasets, those models are actually making more human-like decisions on an individual image level; image-level response consistency is a much stricter behavioural requirement than just e.g. matching overall accuracies. Taken together, our results give reason to celebrate partial success in closing the gap between human and machine vision. In those cases where there is still ample room for improvement, our psychophysical benchmark datasets and toolbox may prove useful in quantifying future progress.

### Acknowledgments and disclosure of funding

We thank Andreas Geiger, Simon Kornblith, Kristof Meding, Claudio Michaelis and Ludwig Schmidt for helpful discussions regarding different aspects of this work; Lukas Huber, Maximus Mutschler, David-Elias Künstle for feedback on the manuscript; Ken Kahn for pointing out typos; Santiago Cadena for sharing a PyTorch implementation of SimCLR; Katherine Hermann and her collaborators for providing supervised SimCLR baselines; Uli Wannek and Silke Gramer for infrastructure/administrative support; the many authors who made their models publicly available; and our anonymous reviewers for many valuable suggestions.

Furthermore, we are grateful to the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting R.G.; the Collaborative Research Center (Projektnummer 276693517—SFB 1233: Robust Vision) for supporting M.B. and F.A.W. This work was supported by the German Federal Ministry of Education and Research (BMBF): Tübingen AI Center, FKZ: 01IS18039A (W.B. and M.B.). F.A.W. is a member of the Machine Learning Cluster of Excellence, EXC number 2064/1—Project number 390727645. M.B. and W.B. acknowledge funding from the MICrONS program of the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/Interior Business Center (DoI/IBC) contract number D16PC00003. W.B. acknowledges financial support via the Emmy Noether Research Group on The Role of Strong Response Consistency for Robust and Explainable Machine Vision funded by the German Research Foundation (DFG) under grant no. BR 6382/1-1.

### Author contributions

Project idea: R.G. and W.B.; project lead: R.G.; coding toolbox and model evaluation pipeline: R.G., K.N. and B.M. based on a prototype by R.G.; training models: K.N. with input from R.G., W.B. and M.B.; data visualisation: R.G., B.M. and K.N. with input from M.B., F.A.W. and W.B.; psychophysical data collection: T.T. (12 datasets) and B.M. (2 datasets) under the guidance of R.G. and F.A.W.; curating stimuli: R.G.; interpreting analyses and findings: R.G., M.B., F.A.W. and W.B.; guidance, feedback, infrastructure & funding acquisition: M.B., F.A.W. and W.B.; paper writing: R.G. with help from F.A.W. and W.B. and input from all other authors.

### References

- [1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.
- [2] Jinjiang Wang, Yulin Ma, Laibin Zhang, Robert X Gao, and Dazhong Wu. Deep learning for smart manufacturing: Methods and applications. *Journal of Manufacturing Systems*, 48:144–156, 2018.
- [3] Javier Villalba-Diez, Daniel Schmidt, Roman Gevers, Joaquín Ordieres-Meré, Martin Buchwitz, and Wanja Wellbrock. Deep learning for industrial computer vision quality control in the printing industry 4.0. *Sensors*, 19(18):3987, 2019.
- [4] Christof Angermueller, Tanel Pärnamaa, Leopold Parts, and Oliver Stegle. Deep learning for computational biology. *Molecular Systems Biology*, 12(7):878, 2016.
- [5] Adam H Marblestone, Greg Wayne, and Konrad P Kording. Toward an integration of deep learning and neuroscience. *Frontiers in Computational Neuroscience*, 10:94, 2016.
- [6] Garrett B Goh, Nathan O Hodas, and Abhinav Vishnu. Deep learning for computational chemistry. *Journal of Computational Chemistry*, 38(16):1291–1307, 2017.
- [7] Travers Ching, Daniel S Himmelstein, Brett K Beaulieu-Jones, Alexandr A Kalinin, Brian T Do, Gregory P Way, Enrico Ferrero, Paul-Michael Agapow, Michael Zietz, Michael M Hoffman, et al. Opportunities and obstacles for deep learning in biology and medicine. *Journal of The Royal Society Interface*, 15(141): 20170387, 2018.
- [8] Dan Guest, Kyle Cranmer, and Daniel Whiteson. Deep learning and its application to LHC physics. *Annual Review of Nuclear and Particle Science*, 68:161–181, 2018.
- [9] Andrew W Senior, Richard Evans, John Jumper, James Kirkpatrick, Laurent Sifre, Tim Green, Chongli Qin, Augustin Židek, Alexander WR Nelson, Alex Bridgland, et al. Improved protein structure prediction using potentials from deep learning. *Nature*, 577(7792):706–710, 2020.
- [10] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.
- [11] Jason Jo and Yoshua Bengio. Measuring the tendency of CNNs to learn surface statistical regularities. *arXiv preprint arXiv:1711.11561*, 2017.
- [12] Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *Proceedings of the European Conference on Computer Vision*, pages 456–473, 2018.
- [13] Timothy Niven and Hung-Yu Kao. Probing neural network comprehension of natural language arguments. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4658–4664, 2019.
- [14] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2:665–673, 2020.
- [15] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019.
- [16] Haohan Wang, Songwei Ge, Eric P Xing, and Zachary C Lipton. Learning robust global representations by penalizing local predictive power. *arXiv preprint arXiv:1905.13549*, 2019.
- [17] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2019.

- [18] Michael A Alcorn, Qi Li, Zhitao Gong, Chengfei Wang, Long Mai, Wei-Shinn Ku, and Anh Nguyen. Strike (with) a pose: Neural networks are easily fooled by strange poses of familiar objects. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.
- [19] Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.
- [20] Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. ObjectNet: a large-scale bias-controlled dataset for pushing the limits of object recognition models. In *Advances in Neural Information Processing Systems*, pages 9448–9458, 2019.
- [21] Claudio Michaelis, Benjamin Mitzkus, Robert Geirhos, Evgenia Rusak, Oliver Bringmann, Alexander S Ecker, Matthias Bethge, and Wieland Brendel. Benchmarking robustness in object detection: Autonomous driving when winter is coming. *arXiv:1907.07484*, 2019.
- [22] Rohan Taori, Achal Dave, Vaishaal Shankar, Nicholas Carlini, Benjamin Recht, and Ludwig Schmidt. Measuring robustness to natural distribution shifts in image classification. In *Advances in Neural Information Processing Systems*, 2020.
- [23] Josip Djolonga, Jessica Yung, Michael Tschannen, Rob Romijnders, Lucas Beyer, Alexander Kolesnikov, Joan Puigcerver, Matthias Minderer, Alexander D’Amour, Dan Moldovan, et al. On robustness and transferability of convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 16458–16468, 2021.
- [24] Isaac Dunn, Hadrien Pouget, Daniel Kroening, and Tom Melham. Exposing previously undetectable faults in deep neural networks. *arXiv preprint arXiv:2106.00576*, 2021.
- [25] Yue He, Zheyang Shen, and Peng Cui. Towards non-iid image classification: A dataset and baselines. *Pattern Recognition*, 110:107383, 2021.
- [26] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8340–8349, 2021.
- [27] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 15262–15271, 2021.
- [28] Spandan Madan, Tomotake Sasaki, Tzu-Mao Li, Xavier Boix, and Hanspeter Pfister. Small in-distribution changes in 3D perspective and lighting fool both CNNs and transformers. *arXiv preprint arXiv:2106.16198*, 2021.
- [29] Vaishaal Shankar, Achal Dave, Rebecca Roelofs, Deva Ramanan, Benjamin Recht, and Ludwig Schmidt. Do image classifiers generalize across time? In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9661–9669, 2021.
- [30] David M. Green. Consistency of auditory detection judgments. *Psychological Review*, 71(5):392–407, 1964.
- [31] Kristof Meding, Dominik Janzing, Bernhard Schölkopf, and Felix A. Wichmann. Perceiving the arrow of time in autoregressive motion. *Advances in Neural Information Processing Systems (NeurIPS)*, 32: 2303–2314, 2019.
- [32] Robert Geirhos, Kristof Meding, and Felix A Wichmann. Beyond accuracy: quantifying trial-by-trial behaviour of CNNs and humans by measuring error consistency. *Advances in Neural Information Processing Systems*, 33, 2020.
- [33] Robert Geirhos, Carlos RM Temme, Jonas Rauber, Heiko H Schütt, Matthias Bethge, and Felix A Wichmann. Generalisation in humans and deep neural networks. In *Advances in Neural Information Processing Systems*, 2018.
- [34] George A Miller. WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41, 1995.
- [35] Philip L Smith and Daniel R Little. Small is beautiful: In defense of the small-N design. *Psychonomic Bulletin & Review*, 25(6):2083–2101, 2018.

- [36] Siavash Haghir, Patricia Rubisch, Robert Geirhos, Felix Wichmann, and Ulrike von Luxburg. Comparison-based framework for psychophysics: lab versus crowdsourcing. *arXiv preprint arXiv:1905.07234*, 2019.
- [37] Sébastien Marcel and Yann Rodriguez. Torchvision the machine-vision package of torch. In *Proceedings of the 18th ACM International Conference on Multimedia*, pages 1485–1488, 2010.
- [38] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3733–3742, 2018.
- [39] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.
- [40] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020.
- [41] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6707–6717, 2020.
- [42] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning. *arXiv preprint arXiv:2005.10243*, 2020.
- [43] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.
- [44] Robert Geirhos, Kantharaju Narayanappa, Benjamin Mitzkus, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. On the surprising similarities between supervised and self-supervised models. *arXiv preprint arXiv:2010.08377*, 2020.
- [45] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (BiT): General visual representation learning. *arXiv preprint arXiv:1912.11370*, 6(2):8, 2019.
- [46] Hadi Salman, Andrew Ilyas, Logan Engstrom, Ashish Kapoor, and Aleksander Madry. Do adversarially robust ImageNet models transfer better? *arXiv preprint arXiv:2007.08489*, 2020.
- [47] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.
- [48] Ross Wightman. PyTorch image models. <https://github.com/rwightman/pytorch-image-models>, 2019.
- [49] I Zeki Yalniz, Hervé Jégou, Kan Chen, Manohar Paluri, and Dhruv Mahajan. Billion-scale semi-supervised learning for image classification. *arXiv preprint arXiv:1905.00546*, 2019.
- [50] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves ImageNet classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10687–10698, 2020.
- [51] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Aspell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021.
- [52] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46, 1960.
- [53] Samuel Dodge and Lina Karam. A study and comparison of human and deep learning recognition performance under visual distortions. In *26th International Conference on Computer Communication and Networks*, pages 1–7. IEEE, 2017.
- [54] Felix A Wichmann, David HJ Janssen, Robert Geirhos, Guillermo Aguilar, Heiko H Schütt, Marianne Maertens, and Matthias Bethge. Methods and measurements to compare men against machines. *Electronic Imaging, Human Vision and Electronic Imaging*, 2017(14):36–45, 2017.
- [55] Thomas Serre. Deep learning: the good, the bad, and the ugly. *Annual Review of Vision Science*, 5:399–426, 2019.

- [56] Yann LeCun. Predictive learning, 2016. URL <https://www.youtube.com/watch?v=0unt2Y4qxQo>.
- [57] William Lotter, Gabriel Kreiman, and David Cox. A neural network trained for prediction mimics diverse features of biological neurons and perception. *Nature Machine Intelligence*, 2(4):210–219, 2020.
- [58] A Emin Orhan, Vaibhav V Gupta, and Brenden M Lake. Self-supervised learning through the eyes of a child. *arXiv preprint arXiv:2007.16189*, 2020.
- [59] Talia Konkle and George A Alvarez. Instance-level contrastive learning yields human brain-like representation without category-supervision. *bioRxiv*, 2020.
- [60] Chengxu Zhuang, Siming Yan, Aran Nayebi, Martin Schrimpf, Michael Frank, James DiCarlo, and Daniel Yamins. Unsupervised neural network models of the ventral visual stream. *bioRxiv*, 2020.
- [61] Katherine R Storrs, Barton L Anderson, and Roland W Fleming. Unsupervised learning predicts human perception and misperception of gloss. *Nature Human Behaviour*, pages 1–16, 2021.
- [62] Katherine Hermann, Ting Chen, and Simon Kornblith. The origins and prevalence of texture bias in convolutional neural networks. *Advances in Neural Information Processing Systems*, 33, 2020.
- [63] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.
- [64] Ruitong Huang, Bing Xu, Dale Schuurmans, and Csaba Szepesvári. Learning with a strong adversary. *arXiv preprint arXiv:1511.03034*, 2015.
- [65] Simran Kaur, Jeremy Cohen, and Zachary C Lipton. Are perceptually-aligned gradients a general property of robust classifiers? *arXiv preprint arXiv:1910.08640*, 2019.
- [66] Shibani Santurkar, Dimitris Tsipras, Brandon Tran, Andrew Ilyas, Logan Engstrom, and Aleksander Madry. Image synthesis with a single (robust) classifier. *arXiv:1906.09453*, 2019.
- [67] Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Brandon Tran, and Aleksander Madry. Adversarial robustness as a prior for learned representations. *arXiv preprint arXiv:1906.00945*, 2019.
- [68] Florian Tramèr, Jens Behrmann, Nicholas Carlini, Nicolas Papernot, and Jörn-Henrik Jacobsen. Fundamental tradeoffs between invariance and sensitivity to adversarial perturbations. In *International Conference on Machine Learning*, pages 9561–9571. PMLR, 2020.
- [69] Tianyuan Zhang and Zhanxing Zhu. Interpreting adversarially trained convolutional neural networks. In *International Conference on Machine Learning*, pages 7502–7511. PMLR, 2019.
- [70] Peijie Chen, Chirag Agarwal, and Anh Nguyen. The shape and simplicity biases of adversarially robust ImageNet-trained CNNs. *arXiv preprint arXiv:2006.09373*, 2020.
- [71] Kai Han, Yunhe Wang, Hanting Chen, Xinghao Chen, Jianyuan Guo, Zhenhua Liu, Yehui Tang, An Xiao, Chunjing Xu, Yixing Xu, et al. A survey on visual transformer. *arXiv preprint arXiv:2012.12556*, 2020.
- [72] Muzammal Naseer, Kanchana Ranasinghe, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Intriguing properties of vision transformers. *arXiv preprint arXiv:2105.10497*, 2021.
- [73] Shikhar Tuli, Ishita Dasgupta, Erin Grant, and Thomas L Griffiths. Are convolutional neural networks or transformers more like human vision? *arXiv preprint arXiv:2105.07197*, 2021.
- [74] Curtis G Northcutt, Anish Athalye, and Jonas Mueller. Pervasive label errors in test sets destabilize machine learning benchmarks. *arXiv preprint arXiv:2103.14749*, 2021.
- [75] Rishi Rajalingham, Elias B Issa, Pouya Bashivan, Kohitij Kar, Kailyn Schmidt, and James J DiCarlo. Large-scale, high-resolution comparison of the core visual object recognition behavior of humans, monkeys, and state-of-the-art deep artificial neural networks. *Journal of Neuroscience*, 38(33):7255–7269, 2018.
- [76] Horia Mania, John Miller, Ludwig Schmidt, Moritz Hardt, and Benjamin Recht. Model similarity mitigates test set overuse. *Advances in Neural Information Processing Systems*, 2019.
- [77] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv:1312.6199*, 2013.
- [78] Florian Tramèr, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. The space of transferable adversarial examples. *arXiv preprint arXiv:1704.03453*, 2017.

- [79] Wieland Brendel and Matthias Bethge. Approximating CNNs with bag-of-local-features models works surprisingly well on ImageNet. In *International Conference on Learning Representations*, 2019.
- [80] Kristof Meding, Luca M Schulze Buschoff, Robert Geirhos, and Felix A Wichmann. Trivial or impossible—dichotomous data difficulty masks model differences (on ImageNet and beyond). *arXiv preprint arXiv:2110.05922*, 2021.
- [81] Matthias Hein and Maksym Andriushchenko. Formal guarantees on the robustness of a classifier against adversarial manipulation. *arXiv preprint arXiv:1705.08475*, 2017.
- [82] Arash Vahdat. Toward robustness against label noise in training deep discriminative neural networks. *arXiv preprint arXiv:1706.00038*, 2017.
- [83] James J DiCarlo, Davide Zoccolan, and Nicole C Rust. How does the brain solve visual object recognition? *Neuron*, 73(3):415–434, 2012.
- [84] Daniel LK Yamins, Ha Hong, Charles F Cadieu, Ethan A Solomon, Darren Seibert, and James J DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, 111(23):8619–8624, 2014.
- [85] Jonas Kubilius, Martin Schrimpf, Kohitij Kar, Rishi Rajalingham, Ha Hong, Najib Majaj, Elias Issa, Pouya Bashivan, Jonathan Prescott-Roy, Kailyn Schmidt, et al. Brain-like object recognition with high-performing shallow recurrent ANNs. *Advances in Neural Information Processing Systems*, 32:12805–12816, 2019.
- [86] Been Kim, Emily Reif, Martin Wattenberg, Samy Bengio, and Michael C Mozer. Neural networks trained on natural scenes exhibit Gestalt closure. *Computational Brain & Behavior*, pages 1–13, 2021.
- [87] Martin N Hebart, Charles Y Zheng, Francisco Pereira, and Chris I Baker. Revealing the multidimensional mental representations of natural objects underlying human similarity judgements. *Nature Human Behaviour*, 4(11):1173–1185, 2020.
- [88] Nikolaus Kriegeskorte. Deep neural networks: a new framework for modeling biological vision and brain information processing. *Annual Review of Vision Science*, 1:417–446, 2015.
- [89] Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40, 2017.
- [90] Amirhossein Tavanaei, Masoud Ghodrati, Saeed Reza Kheradpisheh, Timothée Masquelier, and Anthony Maida. Deep learning in spiking neural networks. *Neural Networks*, 111:47–63, 2019.
- [91] Fabian H Sinz, Xaq Pitkow, Jacob Reimer, Matthias Bethge, and Andreas S Tolias. Engineering a less artificial intelligence. *Neuron*, 103(6):967–979, 2019.
- [92] Joel Dapello, Tiago Marques, Martin Schrimpf, Franziska Geiger, David D Cox, and James J DiCarlo. Simulating a primary visual cortex at the front of CNNs improves robustness to image perturbations. *bioRxiv*, 2020.
- [93] Benjamin D Evans, Gaurav Malhotra, and Jeffrey S Bowers. Biological convolutions improve DNN robustness to noise and generalisation. *bioRxiv*, 2021.
- [94] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8026–8037, 2019.
- [95] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv:1807.03748*, 2018.
- [96] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: a large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.
- [97] Evgenia Rusak, Steffen Schneider, Peter Gehler, Oliver Bringmann, Wieland Brendel, and Matthias Bethge. Adapting ImageNet-scale models to complex distribution shifts with self-learning. *arXiv preprint arXiv:2104.12928*, 2021.
- [98] Eleanor Rosch. Principles of categorization. In E. Margolis and S. Laurence, editors, *Concepts: Core Readings*, pages 189–206. 1999.



## Appendix

We here provide details on models (A), describe additional predictions and experiments regarding error consistency mentioned in Section 4 (B), report experimental details regarding our psychophysical experiments (C), provide license information (D), and describe training with ImageNet labels provided by CLIP (E) as well as experiments with supervised SimCLR baseline models (F), provide overall benchmark scores ranking different models (G), describe a regression model (H) and motivate the choice of behavioural response mapping (I). Stimuli are visualized in Figures 7 and 8.

Our Python library, “modelvshuman”, to test and benchmark models against high-quality human psychophysical data is available from <https://github.com/bethgelab/model-vs-human/>.

### A Model details

**Standard supervised models.** We used all 24 available pre-trained models from the PyTorch model zoo version 1.4.0 (VGG: with batch norm).

**Self-supervised models.** InsDis [38], MoCo [39], MoCoV2 [40], PIRL [41] and InfoMin [42] were obtained as pre-trained models from the PyContrast model zoo. We trained one linear classifier per model on top of the self-supervised representation. A PyTorch [94] implementation of SimCLR [43] was obtained via `simclr-converter`. All self-supervised models use a ResNet-50 architecture and a different training approach within the framework of contrastive learning [e.g. 95].

**Adversarially trained models.** We obtained five adversarially trained models [46] from the `robust-models-transfer` repository. All of them have a ResNet-50 architecture, but a different accuracy-L2-robustness tradeoff indicated by  $\epsilon$ . Here are the five models that we used, in increasing order of adversarial robustness:  $\epsilon = 0, 0.5, 1.0, 3.0, 5.0$ .

**Vision transformers.** Three ImageNet-trained vision transformer (ViT) models [47] were obtained from `pytorch-image-models` [48]. Specifically, we used `vit_small_patch16_224`, `vit_base_patch16_224` and `vit_large_patch16_224`. They are referred to as ViT-S, ViT-B and ViT-L throughout the paper. Additionally, we included two transformers that were pre-trained on ImageNet21K [96], i.e. 14M images with some 21K classes, before they were fine-tuned on “standard” ImageNet-1K. These two models are referred to as ViT-L (14M) and ViT-B (14M) in the paper. They were obtained from the `PyTorch-Pretrained-ViT` repository, where they are called `L_16_imagenet1k` and `B_16_imagenet1k`. (No ViT-S model was available from the repository.) Note that the “imagenet1k” suffix in the model names does not mean the model was only trained on ImageNet1K. On the contrary, this indicates fine-tuning on ImageNet; as mentioned above these models were pre-trained on ImageNet21K before fine-tuning.

**CLIP.** OpenAI trained a variety of CLIP models using different backbone networks [51]. Unfortunately, the best-performing model has not been released so far, and it is not currently clear whether it will be released at some point according to issue #2 of OpenAI’s CLIP github repository. We included the most powerful released model in our analysis, a model with a ViT-B/32 backbone.

**Noisy Student** One pre-trained Noisy Student model was obtained from `pytorch-image-models` [48], where the model is called `tf_efficientnet_e2_ns_475`. This involved the following preprocessing (taken from [97]):

```
from PIL.Image import Image
from torchvision.transforms import Compose, Resize, CenterCrop, ToTensor, Normalize

def get_noisy_student_preprocessing():
    normalize = Normalize(mean=[0.485, 0.456, 0.406],
                        std=[0.229, 0.224, 0.225])

    img_size = 475
    crop_pct = 0.936
    scale_size = img_size / crop_pct
    return Compose([
        Resize(scale_size, interpolation=PIL.Image.BICUBIC),
        CenterCrop(img_size),
        ToTensor(),
        normalize,
    ])
```

**SWSL** Two pre-trained SWSL (semi-weakly supervised) models were obtained from semi-supervised-ImageNet1K-models, one with a ResNet-50 architecture and one with a ResNeXt101\_32x16d architecture.

**BiT-M** Six pre-trained Big Transfer models were obtained from pytorch-image-models [48], where they are called resnetv2\_50x1\_bitm, resnetv2\_50x3\_bitm, resnetv2\_101x1\_bitm, resnetv2\_101x3\_bitm, resnetv2\_152x2\_bitm and resnetv2\_152x4\_bitm.

**Linear classifier training procedure.** The PyContrast repository by Yonglong Tian contains a Pytorch implementation of unsupervised representation learning methods, including pre-trained representation weights. The repository provides training and evaluation pipelines, but it supports only multi-node distributed training and does not (currently) provide weights for the classifier. We have used the repository’s linear classifier evaluation pipeline to train classifiers for InsDis [38], MoCo [39], MoCoV2 [40], PIRL [41] and InfoMin [42] on ImageNet. Pre-trained weights of the model representations (without classifier) were taken from the provided Dropbox link and we then ran the training pipeline on a NVIDIA TESLA P100 using the default parameters configured in the pipeline. Detailed documentation about running the pipeline and parameters can be found in the PyContrast repository (commit #3541b82).

## B Error consistency predictions

Table 1: Error consistency across all five non-parametric datasets. Specifically, this comparison compares the influence of dataset vs. architecture (top) and the influence of flexibility vs. constraints (bottom). Results are described in Section B.

	sketch	stylized	edge	silhouette	cue conflict
ResNet-50 vs. VGG-16	<b>0.74</b>	<b>0.56</b>	<b>0.68</b>	<b>0.71</b>	<b>0.59</b>
ResNet-50 vs. ResNet-50 trained on Stylized-ImageNet	0.44	0.09	0.10	0.67	0.27
ResNet-50 vs. vision transformer (ViT-S)	<b>0.67</b>	<b>0.43</b>	<b>0.41</b>	<b>0.68</b>	<b>0.48</b>
ResNet-50 vs. BagNet-9	0.31	0.17	0.32	0.14	0.44

In Section 4, we hypothesised that shortcut opportunities in the dataset may be a potential underlying cause of high error consistency between models, since all sufficiently flexible models will pick up on those same shortcuts. We then made two predictions which we test here.

**Dataset vs. architecture.** *Prediction:* error consistency between two identical models trained on very different datasets, such as ImageNet vs. Stylized-ImageNet, is much lower than error consistency between very different models (ResNet-50 vs. VGG-16) trained on the same dataset. *Observation:* According to Table 1, this is indeed the case—training ResNet-50 on a different dataset, Stylized-ImageNet [17], leads to lower error consistency than comparing two ImageNet-trained CNNs with different architecture. While this relationship is not perfect (e.g., the difference is small for silhouette images), we have confirmed that this is a general pattern not limited to the specific networks in the table.

**Flexibility vs. constraints.** *Prediction:* error consistency between ResNet-50 and a highly flexible model (e.g., a vision transformer) is much higher than error consistency between ResNet-50 and a highly constrained model like BagNet-9 [79]. *Observation:* A vision transformer (ViT-S) indeed shows higher error consistency with ResNet-50 than with BagNet-9 (see Table 1). However, this difference is not large for one out of five datasets (cue conflict). One could imagine different reasons for this: perhaps BagNet-9 is still flexible enough to learn a decision rule close to the one of standard ResNet-50 for cue conflict images; and of course there is also the possibility that the hypothesis is wrong. Further insights could be gained by testing successively more constrained versions of the same base model.

## C Experimental details regarding psychophysical experiments

### C.1 Participant instructions and preparation

Participants were explained how to respond (via mouse click), instructed to respond as accurately as possible, and to go with their best guess if unsure. In order to rule out any potential misunderstanding, participants were asked to name all 16 categories on the response screen. Prior to the experiment, visual acuity was measured with a Snellen chart to ensure normal or corrected to normal vision. Furthermore, four blocks of 80 practice trials each (320 practice trials in total) on undistorted colour or greyscale images were conducted (non-overlapping with experimental stimuli) to gain familiarity with the task. During practice trials, but not experimental trials, visual and auditory feedback was provided: the correct category was highlighted and a “beep” sound was played for incorrect or missed trials. The experiment itself consisted of blocks of 80 trials each, after each block participants were free to take a break. In order to increase participant motivation, aggregated performance over the last block was displayed on the screen.

### C.2 Participant risks

Our experiment was a standard perceptual experiment, for which no IRB approval was required. The task consisted of viewing corrupted images and clicking with a computer mouse. In order to limit participant risks related to a COVID-19 infection, we implemented the following measures: (1.) The experimenter was tested for corona twice per week. (2.) Prior to participation in our experiments, participants were explained that they could perform a (cost-free) corona test next to our building, and that if they choose to do so, we would pay them 10€/hour for the time spent doing the test and waiting for the result (usually approx. 15–30min). (3.) Experimenter and participant adhered to a strict distance of at least 1.5m during the entire course of the experiment, including instructions and practice trials. During the experiment itself, the participant was the only person in the room; the experimenter was seated in an adjacent room. (4.) Wearing a medical mask was mandatory for both experimenter and participant. (5.) Participants were asked to disinfect their hands prior to the experiment; additionally the desk, mouse etc. were disinfected after completion of an experiment. (6.) Participants were tested in a room where high-performance ventilation was installed; in order to ensure that the ventilation was working as expected we performed a one-time safety check measuring CO2 parts-per-million before we decided to go ahead with the experiments.

### C.3 Participant remuneration

Participants were paid 10€ per hour or granted course credit. Additionally, an incentive bonus of up to 15€ could be earned on top of the standard remuneration. This was meant to further motivate our participants to achieve their optimal performance. The minimum performance for receiving a bonus was set as 15% below the mean of the previous experiments accuracy. The bonus then was linearly calculated with the maximal bonus being given from 15% above the previous experiments mean. The total amount spent on participant compensation amounts to 647,50€.

### C.4 Participant declaration of consent

Participants were asked to review and sign the following declaration of consent (of which they received a copy):

***Psychophysical study***

*Your task consists of viewing visual stimuli on a computer monitor and evaluating them by pressing a key. Participation in a complete experimental session is remunerated at 10 Euros/hour.*

***Declaration of consent***

*Herewith I agree to participate in a behavioural experiment to study visual perception. My participation in the study is voluntary. I am informed that I can stop the experiment at any time and without giving any reason without incurring any disadvantages. I know that I can contact the experimenter at any time with questions about the research project.*

***Declaration of consent for data processing and data publication***

*Herewith I agree that the experimental data obtained in the course of the experiment may be used in semianonymised form for scientific evaluation and publication. I agree that my personal data (e.g.*

name, phone number, address) may be stored in digital form; they will not be used for any other purpose than for contacting me. This personal data will remain exclusively within the Wichmannlab and will not be passed on to third parties at any time.

## D Licenses

Licenses for datasets, code and models are included in our code (see directory “licenses/”, file “LICENSES\_OVERVIEW.md” of <https://github.com/bethgelab/model-vs-human>).

## E Training with CLIP labels

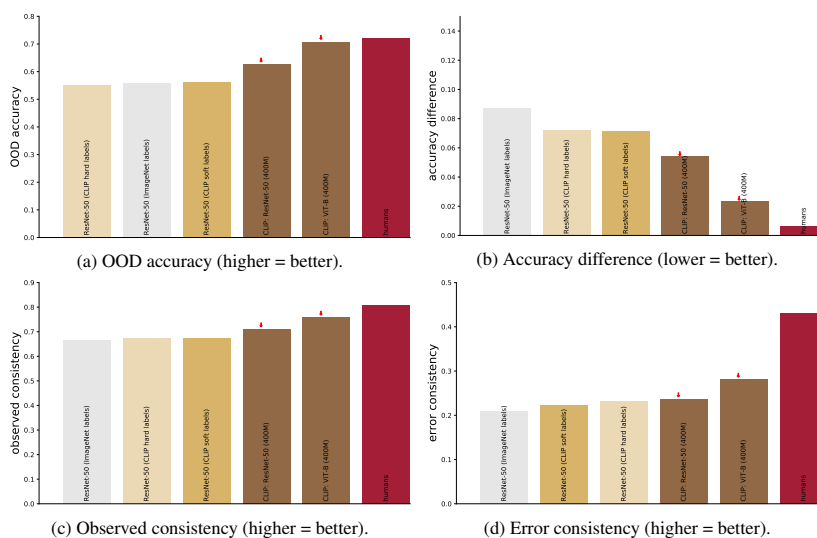


Figure 6: Aggregated results comparing models with and without CLIP-provided labels. Comparison of standard ResNet-50 (light grey), CLIP with vision transformer backend (brown), CLIP with ResNet-50 backend (brown), and standard ResNet-50 with hard labels (bright yellow) vs. soft labels (dark yellow) provided by evaluating standard CLIP on ImageNet; as well as humans (red diamonds) for comparison. Detailed performance across datasets in Figure 16.

As CLIP performed very well across metrics, we intended to obtain a better understanding for why this might be the case. One hypothesis is that CLIP might just receive better labels: About 6% of ImageNet validation images are mis-labeled according to Northcutt et al. [74]. We therefore designed an experiment where we re-labeled the entire ImageNet training and validation dataset using CLIP predictions as ground truth (<https://github.com/kantharajucn/CLIP-imagenet-evaluation>). Having re-labeled ImageNet, we then trained a standard ResNet-50 model from scratch on this dataset using the standard PyTorch ImageNet training script. Training was performed on our on-premise cloud using four RTX 2080 Ti GPUs for five days. We ran the training pipeline in distributed mode with an ncl backed using the default parameters configured in the script, except for the number of workers which we changed to 25. Cross-entropy loss was used to train two models, once with CLIP hard labels (the top-1 class predicted by CLIP) and once with CLIP soft labels (using CLIP’s full posterior distribution as training target). The accuracies on the original ImageNet validation dataset of the resulting models ResNet50-CLIP-hard-labels and ResNet-50-CLIP-soft-labels are 63.53 (top-1), 86.97 (top-5) and 64.63 (top-1), 88.60 (top-5) respectively. In order to make sure that the model trained on soft labels had indeed learned to approximate CLIP’s posterior distribution on ImageNet, we calculated the KL divergence between CLIP soft labels and probability distributions

from ResNet-50 trained on the CLIP soft labels. The resulting value of 0.001 on both ImageNet training and validation dataset is sufficiently small to conclude that the model had successfully learned to approximate CLIP’s posterior distribution on ImageNet. The results are visualised in Figure 6. The results indicate that simply training a standard ResNet-50 model with labels provided by CLIP does not lead to strong improvements on any metric, which means that ImageNet label errors are unlikely to hold standard models back in terms of OOD accuracy and consistency with human responses.

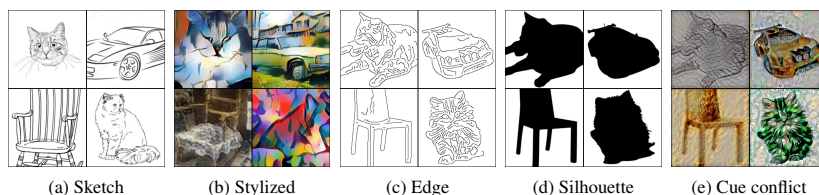


Figure 7: Exemplary stimuli (nonparametric image manipulations) for the following datasets: sketch (7 observers, 800 trials each), stylized (5 observers, 800 trials each), edge (10 observers, 160 trials each), silhouette (10 observers, 160 trials each), and cue conflict (10 observers, 1280 trials each). Figures c–e reprinted from [32] with permission from the authors. [32] also analyzed “diagnostic” images, i.e. stimuli that most humans correctly classified (but few networks) and vice-versa.

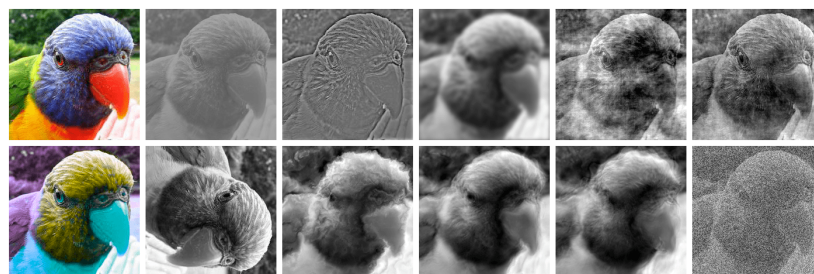


Figure 8: Exemplary stimuli (parametric image manipulations). Manipulations are either binary (e.g. colour vs. grayscale) or they have a parameter (such as the degree of rotation, or the contrast level). Top row: colour vs. grayscale (4 observers, 1280 trials each), low contrast (4 observers, 1280 trials each), high-pass (4 observers, 1280 trials each), low-pass/blurring (4 observers, 1280 trials each), phase noise (4 observers, 1120 trials each), true power spectrum vs. power equalisation (4 observers, 1120 trials each). Bottom row: true vs. opponent colour (4 observers, 1120 trials each), rotation (4 observers, 1280 trials each), Eidolon I (4 observers, 1280 trials each), Eidolon II (4 observers, 1280 trials each), Eidolon III (4 observers, 1280 trials each), additive uniform noise (4 observers, 1280 trials each). Figure adapted from [33] with permission from the authors.

## F Supervised SimCLR baseline models

Figure 15 compares the noise generalisation performance self-supervised SimCLR models against augmentation-matched baseline models. The results indicate that the superior performance of SimCLR in Figure 2 are largely a consequence of SimCLR’s data augmentation scheme, rather than a property of the self-supervised contrastive loss.

## G Benchmark scores

Figure 1 in the main paper shows aggregated scores for the most robust model in terms of OOD accuracy (Figure 1a), and for the most human-like models in terms of accuracy, observed and error consistency (Figures 1b, 1c, 1d). Numerically, these metrics are represented in two tables,

ranking the models according to out-of-distribution robustness (Table 3) and human-like behaviour (Table 2). Since the latter is represented by three different metrics (each characterising a distinct aspect), the mean rank across those three metrics is used to obtain a final ordering. The following conditions and datasets influence benchmark scores: For the five nonparametric datasets, all datasets are taken into account. For the twelve parametric datasets, we also take all datasets into account (overall, all 17 datasets are weighted equally); however, we exclude certain conditions for principled reasons. First of all, the easiest condition is always excluded since it does not test out-of-distribution behaviour (e.g., for the contrast experiment, 100% contrast is more of a baseline condition rather than a condition of interest). Furthermore, we exclude all conditions for which human average performance is strictly smaller than 0.2, since e.g. comparisons against human error patterns are futile if humans are randomly guessing since they cannot identify the stimuli anymore. For these reasons, the following conditions are not taken into account when computing the benchmark scores. Colour vs. greyscale experiment: condition “colour”. True vs. false colour experiment: condition “true colour”. Uniform noise experiment: conditions 0.0, 0.6, 0.9. Low-pass experiment: conditions 0, 15, 40. Contrast experiment: conditions 100, 3, 1. High-pass experiment: conditions inf, 0.55, 0.45, 0.4. Eidolon I experiment: conditions 0, 6, 7. Phase noise experiment: conditions 0, 150, 180. Eidolon II experiment: conditions 0, 5, 6, 7. Power-equalisation experiment: condition “original power spectrum”. Eidolon III experiment: conditions 0, 4, 5, 6, 7. Rotation experiment: condition 0.

## H Regression model

In order to quantify the influence of known independent variables (architecture: transformers vs. ConvNets; data: small (ImageNet) vs. large (“more” than standard ImageNet); objective: supervised vs. self-supervised) on known dependent variables (OOD accuracy and error consistency with humans), we performed a regression analysis using R version 3.6.3 (functions `lm` for fitting and `anova` for regression model comparison). We modelled the influence of those predictors on OOD accuracy, and on error consistency with human observers in two separate linear regression models (one per dependent variable). To this end, we used incremental model building, i.e. starting with one significant predictor and subsequently adding predictors if the reduction of degrees of freedom is justified by a significantly higher degree of explained variance (alpha level: .05). Both error consistency and accuracy, for our 52 models, followed (approximately) a normal distribution, as confirmed by density and Q-Q-plots. That being said, the fit was better for error consistency than for accuracy.

The final regression model for error consistency showed:

- a significant main effect for transformers over CNNs ( $p = 0.01936$  \*),
- a significant main effect for large datasets over small datasets ( $p = 3.39e-05$  \*\*\*),
- a significant interaction between dataset size and objective function ( $p = 0.00625$  \*\*),
- no significant main effect of objective function ( $p = 0.10062$ , n.s.)

Residual standard error: 0.02156 on 47 degrees of freedom  
 Multiple R-squared: 0.5045, Adjusted R-squared: 0.4623  
 F-statistic: 11.96 on 4 and 47 DF, p-value: 8.765e-07  
 Significance codes: 0 ‘\*\*\*\*’ 0.001 ‘\*\*\*’ 0.01 ‘\*\*’ 0.05

The final regression model for OOD accuracy showed:

- a significant main effect for large datasets over small datasets ( $p = 4.65e-09$  \*\*\*),
- a significant interaction between dataset size and architecture type (transformer vs. CNNs;  $p = 0.0174$  \*),
- no significant main effect for transformers vs. CNNs ( $p = 0.8553$ , n.s.)

Residual standard error: 0.0593 on 48 degrees of freedom  
 Multiple R-squared: 0.5848, Adjusted R-squared: 0.5588  
 F-statistic: 22.53 on 3 and 48 DF, p-value: 3.007e-09  
 Significance codes: 0 ‘\*\*\*\*’ 0.001 ‘\*\*\*’ 0.01 ‘\*\*’ 0.05

Limitations: a linear regression model can only capture linear effects; furthermore, diagnostic plots showed a better fit for the error consistency model (where residuals roughly followed the expected distribution as confirmed by a Q-Q-plot) than for the OOD accuracy model (where residuals were not perfectly normal distributed).

## I Mapping behavioural decisions

Comparing model and human classification decisions comes with a challenge: we simply cannot ask human observers to classify objects into 1,000 classes (as for standard ImageNet models). Even if this were feasible in terms of experimental time constraints, most humans don't routinely know the names of a hundred different dog breeds. What they do know, however, is how to tell dogs apart from cats and from airplanes, chairs and boats. Those are so-called "basic" or "entry-level" categories [98]. In line with previous work [17, 32, 33], we therefore used a set of 16 basic categories in our experiments. For ImageNet-trained models, to obtain a choice from the same 16 categories, the 1,000 class decision vector was mapped to those 16 classes using the WordNet hierarchy [34]. Those 16 categories were chosen to reflect a large chunk of ImageNet (227 classes, i.e. roughly a quarter of all ImageNet categories is represented by those 16 basic categories). In order to obtain classification decisions from ImageNet-trained models for those 16 categories, at least two choices are conceivable: re-training the final classification layer or using a principled mapping. Since any training involves making a number of choices (hyperparameters, optimizer, dataset, ...) that may potentially influence and in the worst case even bias the results (e.g. for ShuffleNet, more than half of the model's parameters are contained in the final classification layer!), we decided against training and for a principled mapping by calculating the probability of a coarse class as the average of the probabilities of the corresponding fine-grained classes. Why is this mapping principled? As derived by [33] (pages 22 and 23 in the Appendix of the arXiv version, <https://arxiv.org/pdf/1808.08750.pdf>), this is the optimal way to map (i.e. aggregate) probabilities from many fine-grained classes to a few coarse classes. Essentially, the aggregation can be derived by calculating the posterior distribution of a discriminatively trained CNN under a new prior chosen at test time (here: 1/16 over coarse classes).

Table 2: Benchmark table of model results. The three metrics “accuracy difference” “observed consistency” and “error consistency” (plotted in Figure 1) each produce a different model ranking. The mean rank of a model across those three metrics is used to rank the models on our benchmark.

model	accuracy diff. ↓	obs. consistency ↑	error consistency ↑	mean rank ↓
CLIP: ViT-B (400M)	<b>0.023</b>	0.758	<b>0.281</b>	<b>1.333</b>
SWSL: ResNeXt-101 (940M)	0.028	0.752	0.237	4.000
BiT-M: ResNet-101x1 (14M)	0.034	0.733	0.252	4.333
BiT-M: ResNet-152x2 (14M)	0.035	0.737	0.243	5.000
ViT-L	0.033	0.738	0.222	6.667
BiT-M: ResNet-152x4 (14M)	0.035	0.732	0.233	7.667
BiT-M: ResNet-50x3 (14M)	0.040	0.726	0.228	9.333
BiT-M: ResNet-50x1 (14M)	0.042	0.718	0.240	9.667
ViT-L (14M)	0.035	0.744	0.206	9.667
SWSL: ResNet-50 (940M)	0.041	0.727	0.211	11.667
ViT-B	0.044	0.719	0.223	12.000
BiT-M: ResNet-101x3 (14M)	0.040	0.720	0.204	14.333
densenet201	0.060	0.695	0.212	15.000
ViT-B (14M)	0.049	0.717	0.209	15.000
ViT-S	0.066	0.684	0.216	16.667
densenet169	0.065	0.688	0.207	17.333
inception_v3	0.066	0.677	0.211	17.667
Noisy Student: ENetL2 (300M)	0.040	<b>0.764</b>	0.169	18.000
ResNet-50 L2 eps 1.0	0.079	0.669	0.224	21.000
ResNet-50 L2 eps 3.0	0.079	0.663	0.239	22.000
wide_resnet101_2	0.068	0.676	0.187	24.333
SimCLR: ResNet-50x4	0.071	0.698	0.179	24.667
SimCLR: ResNet-50x2	0.073	0.686	0.180	25.333
ResNet-50 L2 eps 0.5	0.078	0.668	0.203	25.333
densenet121	0.077	0.671	0.200	25.333
resnet101	0.074	0.671	0.192	25.667
resnet152	0.077	0.675	0.190	25.667
resnext101_32x8d	0.074	0.674	0.182	26.667
ResNet-50 L2 eps 5.0	0.087	0.649	0.240	27.000
resnet50	0.087	0.665	0.208	28.667
resnet34	0.084	0.662	0.205	29.333
vgg19_bn	0.081	0.660	0.200	30.000
resnext50_32x4d	0.079	0.666	0.184	30.333
SimCLR: ResNet-50x1	0.080	0.667	0.179	32.000
resnet18	0.091	0.648	0.201	34.667
vgg16_bn	0.088	0.651	0.198	34.667
wide_resnet50_2	0.084	0.663	0.176	35.667
MoCoV2: ResNet-50	0.083	0.660	0.177	36.000
mobilenet_v2	0.092	0.645	0.196	37.000
ResNet-50 L2 eps 0.0	0.086	0.654	0.178	37.333
mnasnet1_0	0.092	0.646	0.189	38.333
vgg11_bn	0.106	0.635	0.193	38.667
InfoMin: ResNet-50	0.086	0.659	0.168	39.333
vgg13_bn	0.101	0.631	0.180	41.000
mnasnet0_5	0.110	0.617	0.173	45.000
MoCo: ResNet-50	0.107	0.617	0.149	47.000
alexnet	0.118	0.597	0.165	47.333
squeezenet1_1	0.131	0.593	0.175	47.667
PIRL: ResNet-50	0.119	0.607	0.141	48.667
shufflenet_v2_x0_5	0.126	0.592	0.160	49.333
InsDis: ResNet-50	0.131	0.593	0.138	50.667
squeezenet1_0	0.145	0.574	0.153	51.000



Table 3: Benchmark table of model results (accuracy).

model	OOD accuracy $\uparrow$	rank $\downarrow$
Noisy Student: ENetL2 (300M)	<b>0.829</b>	<b>1.000</b>
ViT-L (14M)	0.733	2.000
CLIP: ViT-B (400M)	0.708	3.000
ViT-L	0.706	4.000
SWSL: ResNeXt-101 (940M)	0.698	5.000
BiT-M: ResNet-152x2 (14M)	0.694	6.000
BiT-M: ResNet-152x4 (14M)	0.688	7.000
BiT-M: ResNet-101x3 (14M)	0.682	8.000
BiT-M: ResNet-50x3 (14M)	0.679	9.000
SimCLR: ResNet-50x4	0.677	10.000
SWSL: ResNet-50 (940M)	0.677	11.000
BiT-M: ResNet-101x1 (14M)	0.672	12.000
ViT-B (14M)	0.669	13.000
ViT-B	0.658	14.000
BiT-M: ResNet-50x1 (14M)	0.654	15.000
SimCLR: ResNet-50x2	0.644	16.000
densenet201	0.621	17.000
densenet169	0.613	18.000
SimCLR: ResNet-50x1	0.596	19.000
resnext101_32x8d	0.594	20.000
resnet152	0.584	21.000
wide_resnet101_2	0.583	22.000
resnet101	0.583	23.000
ViT-S	0.579	24.000
densenet121	0.576	25.000
MoCoV2: ResNet-50	0.571	26.000
inception_v3	0.571	27.000
InfoMin: ResNet-50	0.571	28.000
resnext50_32x4d	0.569	29.000
wide_resnet50_2	0.566	30.000
resnet50	0.559	31.000
resnet34	0.553	32.000
ResNet-50 L2 eps 0.5	0.551	33.000
ResNet-50 L2 eps 1.0	0.547	34.000
vgg19_bn	0.546	35.000
ResNet-50 L2 eps 0.0	0.545	36.000
ResNet-50 L2 eps 3.0	0.530	37.000
vgg16_bn	0.530	38.000
mnasnet1_0	0.524	39.000
resnet18	0.521	40.000
mobilenet_v2	0.520	41.000
MoCo: ResNet-50	0.502	42.000
ResNet-50 L2 eps 5.0	0.501	43.000
vgg13_bn	0.499	44.000
vgg11_bn	0.498	45.000
PIRL: ResNet-50	0.489	46.000
mnasnet0_5	0.472	47.000
InsDis: ResNet-50	0.468	48.000
shufflenet_v2_x0_5	0.440	49.000
alexnet	0.434	50.000
squeezenet1_1	0.425	51.000
squeezenet1_0	0.401	52.000

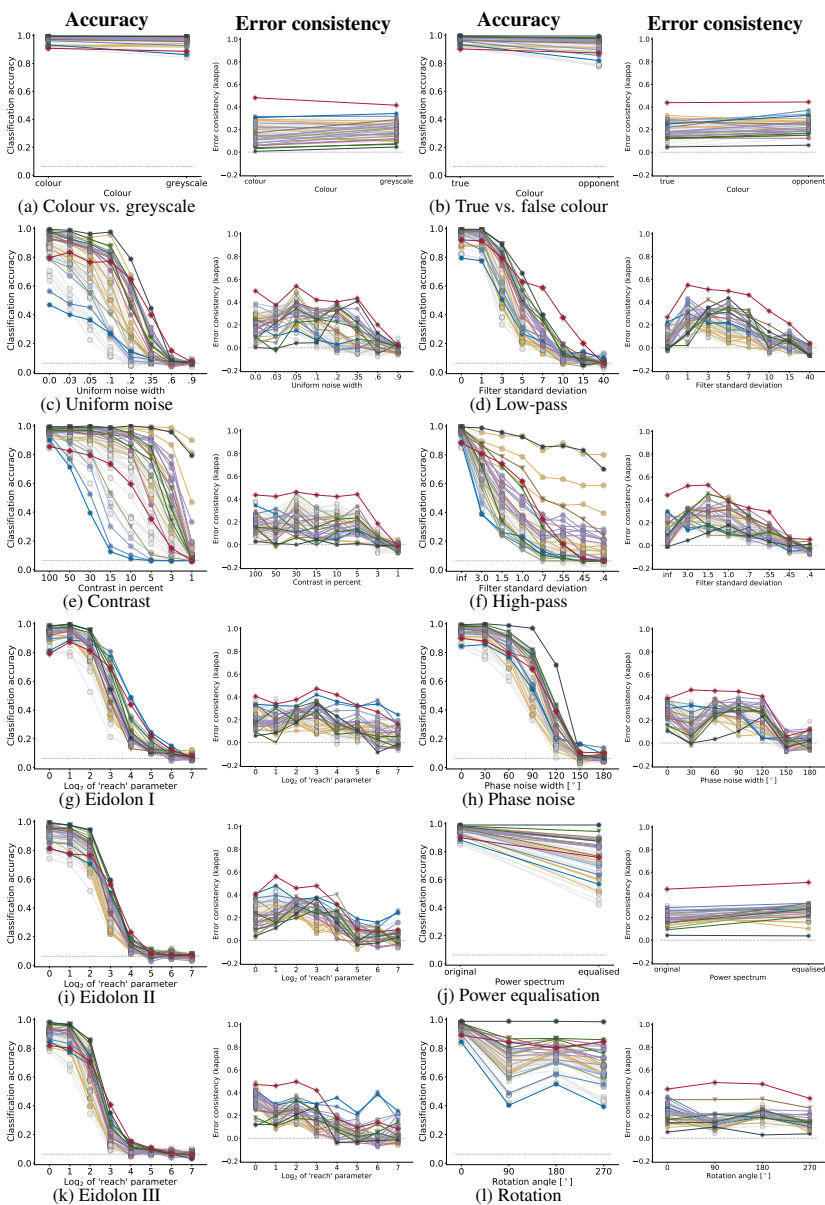


Figure 9: OOD generalisation and error consistency results for humans, standard supervised CNNs, self-supervised models, adversarially trained models, vision transformers, noisy student, BiT, SWSL, CLIP. Symbols indicate architecture type ( $\circ$  convolutional,  $\nabla$  vision transformer,  $\diamond$  human); best viewed on screen. ‘Accuracy’ measures recognition performance (higher is better), ‘error consistency’ how closely image-level errors are aligned with humans. Accuracy results are identical to Figure 2 in the main paper. In many cases, human-to-human error consistency increases for moderate distortion levels and drops afterwards.

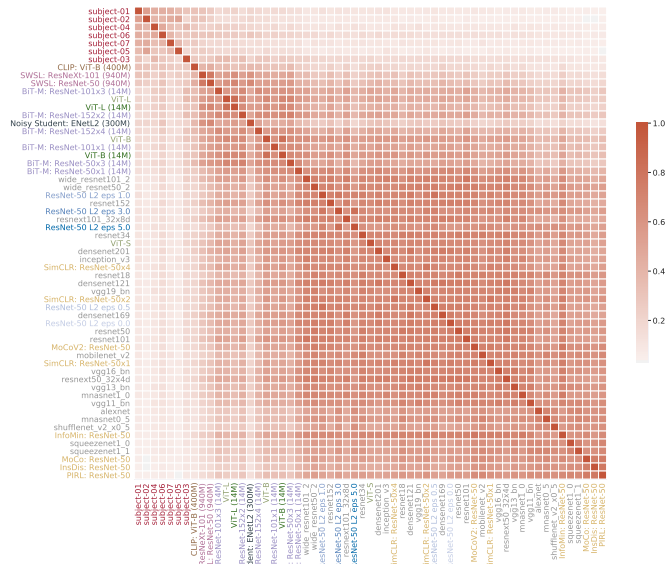


Figure 10: Error consistency for 'sketch' images (same as Figure 4 but sorted w.r.t. mean error consistency with humans).

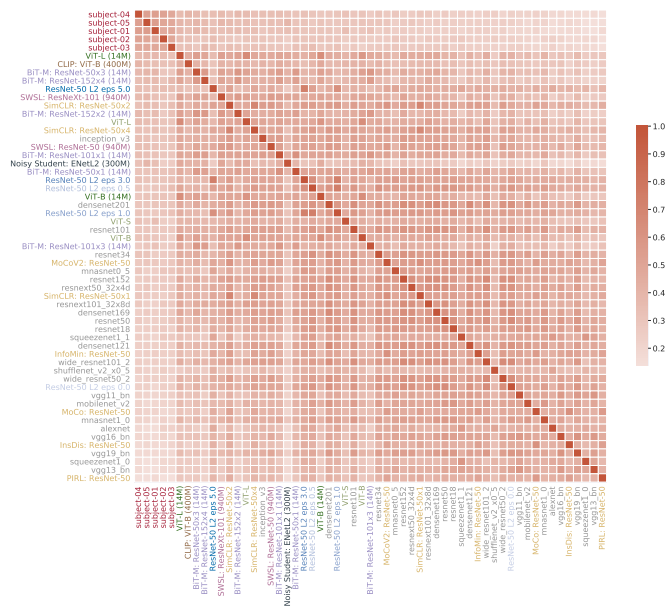


Figure 11: Error consistency for 'stylized' images (sorted w.r.t. mean error consistency with humans).

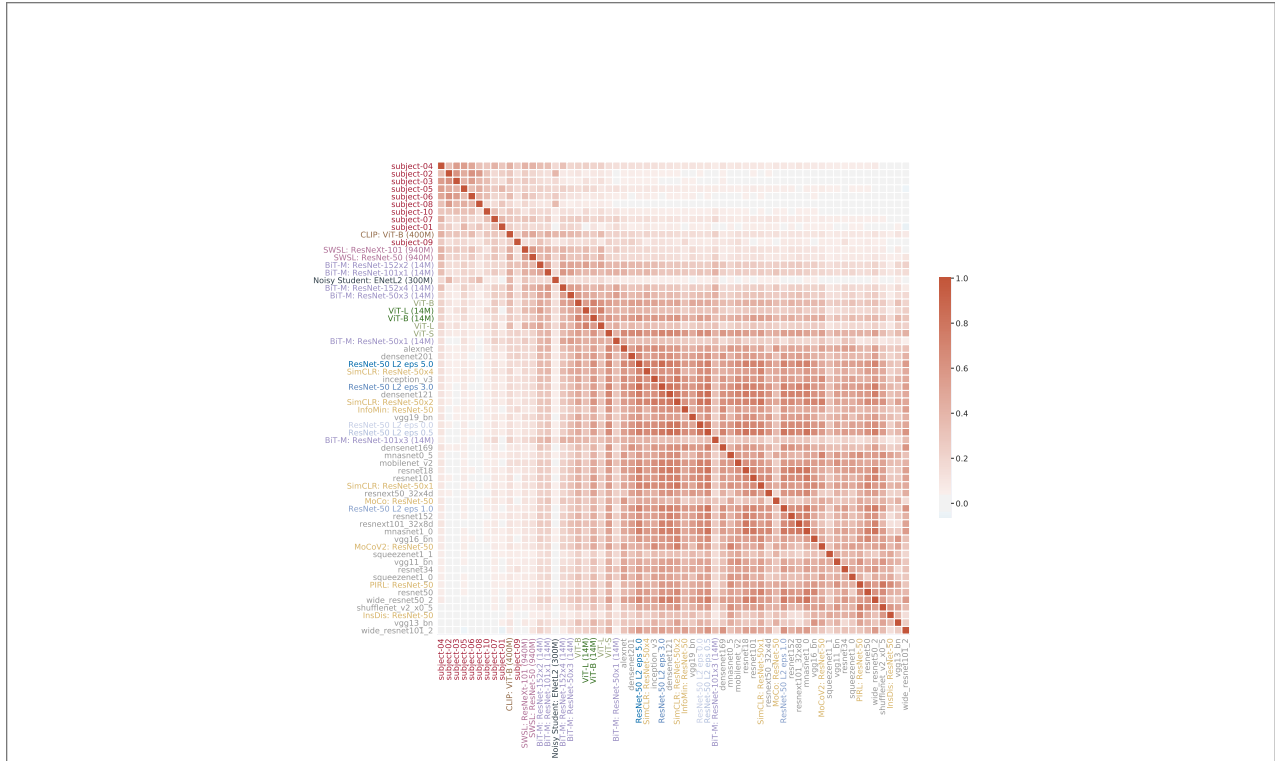


Figure 12: Error consistency for 'edge' images (sorted w.r.t. mean error consistency with humans).



Figure 13: Error consistency for 'silhouette' images (sorted w.r.t. mean error consistency with humans).

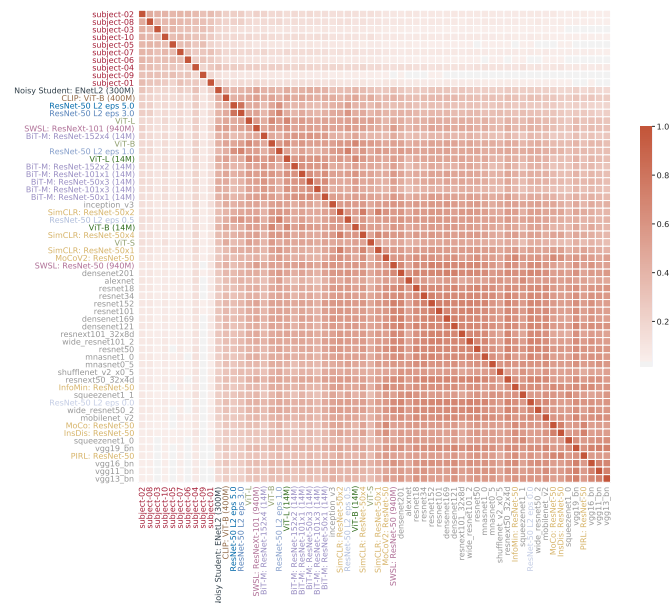


Figure 14: Error consistency for 'cue conflict' images (sorted w.r.t. mean error consistency with humans).

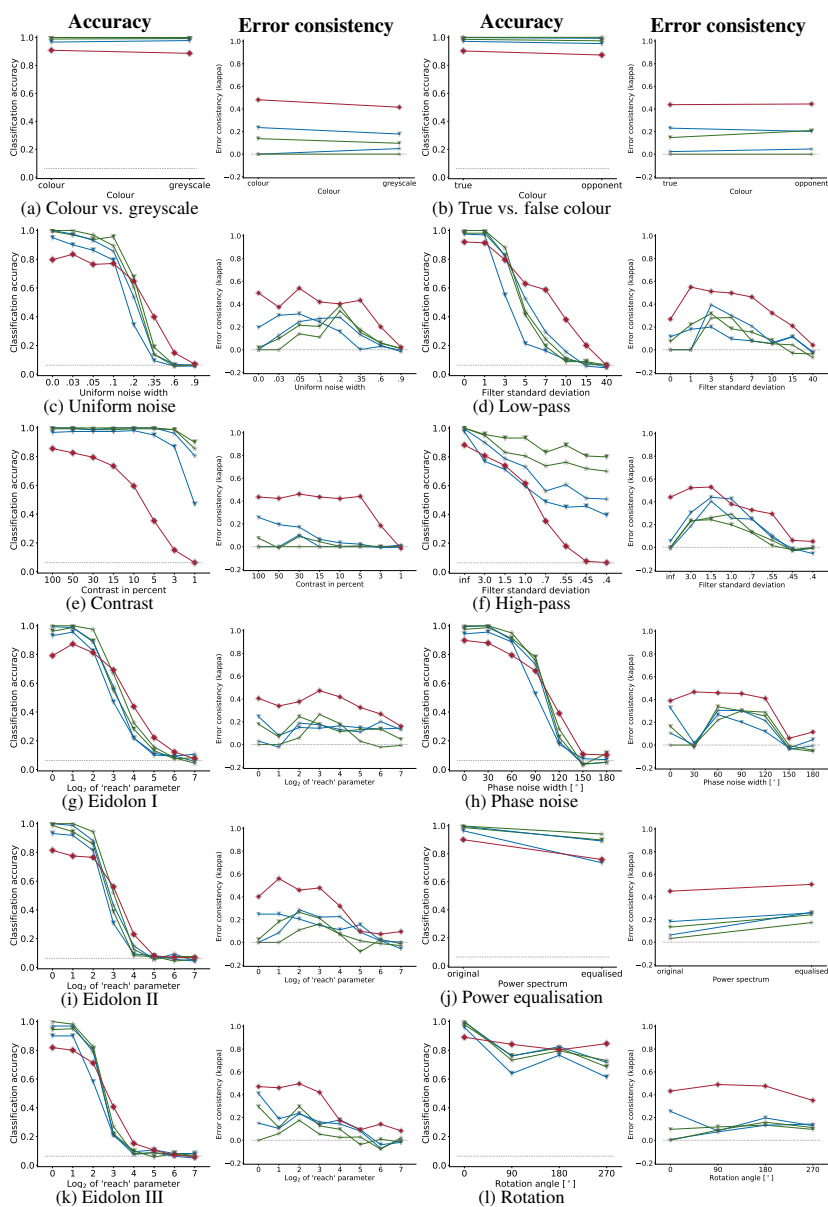


Figure 15: Comparison of self-supervised SimCLR models with supervised, augmentation-matched baseline models. Note that for better visibility, the colours and symbols deviate from previous plots. Plotting symbols: triangles for self-supervised models, stars for supervised baselines. Two different model-baseline pairs are plotted; they differ in the model width: blue models have 1x ResNet width, green models have 4x ResNet width [43]. For context, human observers are plotted as red diamonds. Baseline models kindly provided by Hermann et al. [62].

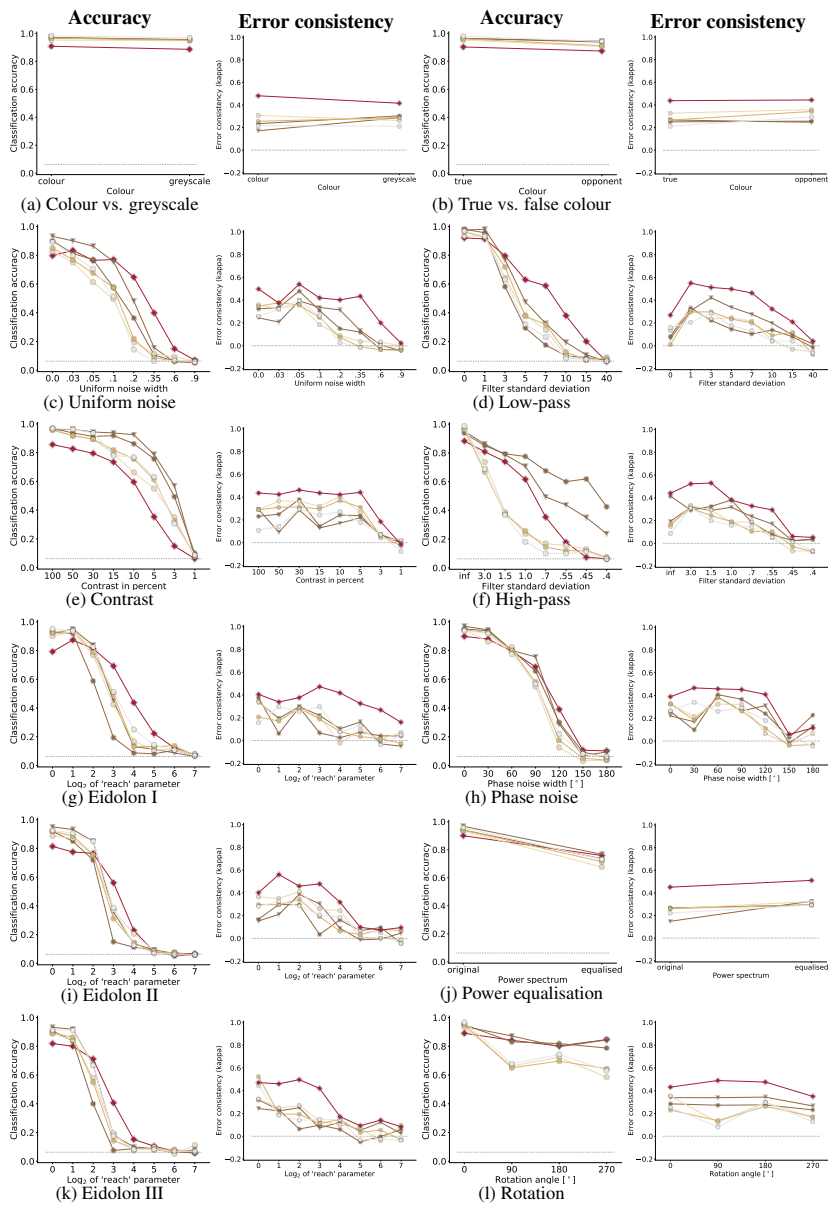


Figure 16: Do CLIP-provided labels lead to better performance? Comparison of standard ResNet-50 (light grey circles), CLIP with vision transformer backend (brown triangles), CLIP with ResNet-50 backend (brown circles), and standard ResNet-50 with hard labels (bright yellow circles) vs. soft labels (dark yellow circles) provided by evaluating standard CLIP on ImageNet; as well as humans (red diamonds) for comparison. Symbols indicate architecture type (○ convolutional, ▽ vision transformer, ◇ human); best viewed on screen. With the exception of high-pass filtered images, standard CLIP training with a ResNet-50 backbone performs fairly poorly.





## 3 Discussion

DEEP LEARNING is here to stay, and looking back at how its importance rapidly grew over the last decade already gives a glimpse of the even bigger impact it is likely to have in the future. In order for this to be a net positive impact, it is crucial that our understanding of machine decision-making will be able to keep pace with applications of deep learning. Over the course of six projects, I set out to develop a better understanding of machine decision-making through a functional comparison with human decision-making. While there are many different lenses through which these projects can be viewed, I would like to specifically discuss two perspectives here: the *inductive bias perspective* (Section 3.1), which is a machine-centric point of view, and the *model of human object recognition perspective* (Section 3.2), a human-centric point of view. This chapter will conclude with a discussion of *limitations* (Section 3.3). Finally, an *outlook* chapter takes on a more forward-looking role by discussing two important questions for future research (Chapter 4).

### 3.1 The inductive bias perspective

BIAS IS GOOD. This is one of the most simple and yet most important insights of machine learning.<sup>1</sup> Without bias, generalisation is impossible. This was already recognised by Mitchell (1980, p. 2): “the power of a generalization system follows directly from its biases [...]. Therefore, progress toward understanding learning mechanisms depends upon understanding the sources of, and justification for, various biases.”

Nearly two decades later, this insight was formalized in the so-called *No Free Lunch* theorem (Wolpert & Macready, 1997), which states that the performance of any two algorithms (such as two classifiers), averaged over all possible input configurations, is equal. For instance, consider the following input-output pairs: (0,0), (2,2), (7,7), (22,22), (45, 45), (58,58). Given unseen input 42, which output would we predict? It seems reasonable to infer a linear relationship between input and output, and hence predict output 42 for input 42—but making this

<sup>1</sup>Of course, many forms of bias are harmful, whether they are exhibited by humans or algorithms (e.g. Mehrabi et al., 2019). While it is impossible to obtain a successful algorithm that is unbiased in every regard, this certainly does not imply that we should accept harmful forms of algorithmic bias.

inference step falls nothing short of making an assumption about the relationship between input and output—in other words, being biased. In a world without assumptions,  $(42,42)$  would be as reasonable a prediction as  $(42,-44)$ ,  $(42,\pi)$ , or  $(42,0)$ .<sup>2</sup>

The assumptions that a machine learning model makes before seeing any data is called the model's *inductive bias*. The process of developing a specific machine learning model (which can also be thought of as a hypothesis about the relationship between input and output) can be conceptualised as follows:



Starting with the set of all possible hypotheses, one can narrow this down to a model's hypothesis space by selecting a model's inductive bias. For instance, the class of linear models already has much fewer hypotheses in its hypothesis space compared to the set of all conceivable hypotheses. (It may be important to note that a model's inductive bias is more than just the size of the model's hypothesis space: it is also a matter of how easily certain hypotheses are learned—for instance, some hypotheses may be representable by a model in principle but the model's optimisation never selects them, which would also be a type of bias.) Finally, data is used to estimate model parameters and to obtain a final hypothesis. In the case of a linear model, this would typically mean computing intercept and slope such that the model optimally fits the (known) data. The resulting single hypothesis is then used to obtain a prediction for previously unseen data.

This already shows that there is a fundamental trade-off between inductive bias and data: The more constrained a model's hypothesis space is, the less the model needs to learn from data. (In the extreme case, if the hypothesis space were to contain only a single hypothesis, no data would be needed.) In machine learning, this traditionally motivated incorporating domain knowledge into models, for instance through hand-engineered features. However, in recent times, hand-engineered features often turned out to be inferior to simply learning model parameters through a lot of data (LeCun et al., 2015). As a consequence, modern deep learning is often lacking a good understanding of the inductive bias that certain aspects of a model's architecture, task or loss function bring along. To give an example of what might happen if this understanding is lacking, one cautionary tale is reported by Rendsburg et al. (2020). The authors investigated why NetGAN (Bojchevski et al., 2018), a rather complicated method to sample new graphs similar to input graphs, works well in practice: not because NetGAN uses random walks, not because a GAN can somehow es-

<sup>2</sup> Given different input-output pairs such as  $(1,1)$ ,  $(2,4)$ ,  $(3,9)$ ,  $(5,25)$ ,  $(6,36)$  you probably would have inferred a different (here: quadratic) relationship. However, assuming linearity “whenever the data looks linear” is an example of a bias, even though it is certainly a useful one in the specific world we live in.

cape the No Free Lunch theorem (Wolpert & Macready, 1997), and not because an LSTM is part of the process—instead, it works well simply since the procedure inadvertently introduced a certain low-rank bias.<sup>3</sup> Stripping NetGAN of the GAN/LSTM/random walk components and using the low-rank bias directly works just as well, but is much faster and interpretable. This goes to show how important it is to understand the inductive bias of a model, especially for complex deep learning models.

Many of the projects in my thesis can be considered to contribute a few sentences to the conversation about the inductive biases of deep learning models. In Geirhos et al. (2020b), we discovered that despite large engineering efforts around the development of new *architectures*, these choices hardly matter in terms of the resulting decision boundary: some models are more accurate than others, but they all make highly consistent errors. In a similar vein, in Geirhos et al. (2020c) we demonstrated that despite considerable excitement around contrastive self-supervised learning (which is based on new *loss functions* that do not require labels), the resulting models' inductive biases hardly differ from the inductive biases of their supervised counterparts. Finally, in Michaelis et al. (2019) and Geirhos et al. (2020a) we highlighted that many properties exhibited by models trained on classification also appear across models trained on different *tasks* such as object detection and even natural language processing.

In contrast, there is one aspect that appeared to truly matter over and over again across projects: the decisive role of *data*. The training dataset can change a model's texture bias to a shape bias (Geirhos et al., 2019a), it can drastically improve out-of-distribution robustness (Geirhos et al., 2019a; Michaelis et al., 2019; Geirhos et al., 2021), and it can even lead to improved consistency with human behavioural error patterns (Geirhos et al., 2021). In my interpretation these findings indicate that currently common choices of architectures, loss functions and tasks impose fewer constraints on the hypothesis space than previously thought (and perhaps intended).<sup>4</sup> This leads to a large influence of data when it comes to selecting the final hypothesis—with potential downsides in terms of shortcut learning (Geirhos et al., 2020a) or dataset bias (Wichmann et al., 2010; Torralba & Efros, 2011). Ultimately, much remains to be understood about inductive biases of deep learning models: how they can be formalised, how they interact with different datasets, and how they might be specified as desired.

Like any decision-maker, humans too have an inductive bias, and comparing human against machine decision-making has been a common thread throughout the projects of this thesis. How these results fit into the discussion about deep neural networks as potential models of human object recognition is discussed in the following section.

<sup>3</sup> GAN: Generative Adversarial Network (Goodfellow et al., 2014); LSTM: Long Short-Term Memory (Hochreiter & Schmidhuber, 1997).

<sup>4</sup> This may be related to the finding that common training pipelines are often *underspecified*, i.e. models with identical architectures/loss functions/tasks can have large differences in behaviour depending on innocuous aspects such as the initial random seed (D'Amour et al., 2020).

### 3.2 *The model of human object recognition perspective*

DEEP NEURAL NETWORKS have been proposed as a “new framework for modeling biological vision and brain information processing” by Kriegeskorte (2015, p. 417)—a proposition I termed the *deep learning metaphor of the brain* in the introduction.<sup>5</sup> Today’s considerable excitement around deep neural networks (DNNs) as potential models of primate ventral stream object recognition is primarily based on investigations reporting similar representational spaces (Yamins et al., 2013, 2014; Cadieu et al., 2014; Khaligh-Razavi & Kriegeskorte, 2014; Güçlü & van Gerven, 2015). However, comparisons of behaviour have been rare.<sup>6</sup> This thesis can be seen as an attempt to change this. Our comprehensive toolbox and benchmark (*model-vs-human*) aim at enabling functional comparisons of human and machine decision-making for everyone—including those who do not have a psychophysical laboratory at their disposal (Geirhos et al., 2021). This benchmark brought together many of the building blocks developed in my preceding projects. Comparing human and machine behaviour requires high-quality psychophysical *datasets* (Geirhos et al., 2018, 2019a), precise *metrics* to track behaviour at a much finer scale than overall accuracy (Geirhos et al., 2019a, 2020b), and a well-maintained *codebase* and *benchmark* incorporating leading models of different flavours (Geirhos et al., 2020b,c, 2021).

What can we conclude from this functional comparison about DNNs as potential models of human object recognition? My findings advise caution when it comes to broad-ranging claims about “human-like” models. For instance, recurrent CORnet-S, touted the “the current best model of the primate ventral visual stream” (Kubilius et al., 2019, p. 1) turned out to be a near-equivalent of standard ResNet-50 on a functional level (Geirhos et al., 2020b). This is particularly concerning in the light of substantial hopes that have been placed in recurrent models (O’Reilly et al., 2013; Spoerer et al., 2017; Kietzmann et al., 2019a,b; Lindsay, 2020; van Bergen & Kriegeskorte, 2020). Given that shallow recurrent networks are functionally identical to deep feedforward networks with weight sharing (Liao & Poggio, 2016), one cannot help but wonder whether it is truly “inevitable that computational neuroscience will come to rely increasingly on complex models, likely from the family of deep recurrent neural networks” (Kietzmann et al., 2019a, p. 2), or whether recurrence might instead be mostly an implementational (but not a functional) difference between brains and standard models.

In spite of the numerous behavioural discrepancies between brains and machines that we found, I would argue that it would be wrong to come to a final conclusion about the potential of DNNs as mod-

<sup>5</sup> Strikingly similar excitement for (shallow) neural networks as models for brains and minds was already seen in the 1980s under the banner of parallel distributed processing (McClelland & Rumelhart, 1986; Rumelhart et al., 1986a).

<sup>6</sup> Notable exceptions include e.g. Kheradpisheh et al. (2016a); Rajalingham et al. (2018); Funke et al. (2021).

els of human visual perception just yet. I certainly have been surprised by the rapid pace of progress more than once. For instance, in 2020 we found radically different behavioural error patterns between humans and standard convolutional neural networks (Geirhos et al., 2020b). One year later, vision transformers (Dosovitskiy et al., 2020) are increasingly replacing convolutional networks, and training vision transformers on large-scale datasets already leads to considerable improvements when it comes to error consistency with human observers (Geirhos et al., 2021). This highlights how fast the field is still progressing, and I personally look forward to scrutinising the many improvements yet to come. Whether or not they will change the role of DNNs as models of human object recognition—the tools, datasets and methods developed in this thesis will be available to help answer this question.

### 3.3 *Limitations*

ALTHOUGH EACH OF THE SIX STUDIES presented in this thesis has its own approach and consequently its own limitations, there are some that they have in common. These include, among others:

*(1.) Infinitely many out-of-distribution test sets are conceivable*

A focus of my thesis is comparing humans and machine learning models on out-of-distribution (OOD) data, i.e. on data that does not originate from the same distribution as the standard training and IID test data (IID stands for independent and identically distributed data). As argued in Geirhos et al. (2020a), this approach is sensible since it is impossible to distinguish whether a model just learned a shortcut or the intended solution by testing on IID data alone. The reason for this is that shortcuts—such as, for instance, image backgrounds like “grass” predictive of category “cow”—lead to deceptively good IID performance: in standard datasets, cows typically happen to be photographed against a grass landscape. However, if tested on OOD data, such as a cow on the beach instead of the typical setting, the “grass” shortcut would no longer lead to good performance (Beery et al., 2018).

While OOD testing is a very reliable method to detect shortcut learning, it comes with the challenge that infinitely many OOD test sets are conceivable. This entails that good performance on an OOD test can only be a necessary, but never a sufficient, condition for asserting that a model indeed learned the intended solution. Furthermore, in many cases, there is a clear-cut distinction between IID and OOD data (such as between natural images and silhouettes), but sometimes the boundaries are not as sharp. In the case of ImageNet-V2, for in-

stance, [Recht et al. \(2019\)](#) closely followed the original ImageNet data collection procedure to obtain a non-overlapping test set. Intended to be drawn from exactly the same distribution, models trained on standard ImageNet nonetheless showed a large accuracy drop (on the order of 11% top-1 accuracy) when tested on this new test set: “in practice it is hard to argue whether two high-dimensional distributions are exactly the same. We typically lack a precise definition of either distribution, and collecting a real dataset involves a plethora of design choices.” ([Recht et al., 2019](#), p. 3).<sup>7</sup> These design choices, which also come with the creation of OOD test sets, are certainly a limitation of the overall approach of OOD generalisation testing.

(2.) *Conclusions are limited to tested models*

This limitation may sound obvious, but it is still a very important caveat. We cannot generalise any of the findings of this thesis to models not investigated. While I would certainly hope (and expect) that the findings transfer to other models of the same kind, this remains speculation until tested. I have attempted to make this clear throughout my projects, but there are cases where I did not fully succeed. For instance, the title of [Geirhos et al. \(2019a\)](#) starts with “ImageNet-trained CNNs are biased towards texture” in an effort to be explicit about the fact that the conclusions are limited to *ImageNet-trained* models. Still, this title is a generalisation: Had we been completely precise (at the expense of brevity), the title would have started with “ImageNet-trained ResNet-50, ResNet-152, VGG-16, GoogLeNet, AlexNet, DenseNet-121 and Squeezenet1\_1 are biased towards texture”.

Since the conclusions of my projects will always be limited to the investigated models—which, in all likelihood, will soon be replaced by even better models—most of these projects have aimed to achieve two goals at the same time: on the one hand, to gain a better understanding of current (yet presumably short-lived) state-of-the-art models; on the other hand, to develop metrics, propose benchmarks and make large-scale behavioural datasets openly available.<sup>8</sup> These latter aspects of my PhD thesis will hopefully continue to be useful on a much longer time scale than insights about specific models that are currently in vogue.

(3.) *Different behaviour does not imply different interm. representation*

A functional comparison of human and machine decision-making is limited to drawing statements about behaviour. This means that even though we found many instances of behavioural differences between biological and artificial systems, it cannot be ruled out that they still have highly similar intermediate representations on the basis of these investigations. The relationship between behaviour and intermediate representations will be an important aspect of the following chapter.

<sup>7</sup> In cases like these, it can be beneficial to train a classifier to distinguish between those two near-identical test sets, as e.g. argued by [Geirhos et al. \(2020a\)](#). If the classifier succeeds, one can conclude that there must have been a systematical distribution shift even if this difference was not noticeable to humans.

<sup>8</sup> An example would be the “model-vs-human” toolbox: <https://github.com/bethgelab/model-vs-human>

## 4 Outlook: “Big Questions” for the future

RESEARCH IS ABOUT FINDING ANSWERS—but perhaps equally importantly, about asking the right questions. Good scientists have many more questions in their mind than they have time to answer them. Therefore, an important aspect of a scientist’s life is choosing which questions to tackle next, and usually a good moment to ask this question is after a project is completed. Typically, there are numerous follow-up questions directly related to that project—for instance, in the case of an experimental project, one might ask: Would the investigated effect  $a$  also hold under manipulation  $b$ ? When increasing stimulus presentation time from  $c$  to  $d$ , how would the results change? Similarly, for a computational project: Can we improve on the benchmark even further by replacing architecture  $e$  with architecture  $f$ ?

In many cases, these rather “obvious” follow-up questions can lead to some form of progress, occasionally even to unexpected contradictions or challenges to the previous interpretation. (And, most certainly, control experiments are integral to the scientific routine.) However, while the tree of knowledge does indeed grow from branching out in ever more fine-grained directions, at some point the time has come to take a step back and ask: Are we growing the right branch? Is the direction of our work a direction we consider truly important, a direction that tackles some of the biggest problems in our field? [Hamming \(1986\)](#) once said, “If you do not work on an important problem, it’s unlikely you’ll do important work. It’s perfectly obvious.”

In this spirit, I would like to focus the outlook of this thesis not on the many “obvious” follow-up questions to my PhD research, but on two “Big Questions” arising from and related to my results, questions that I personally consider truly important to our field: “How can the contradiction between behavioural and neural results be resolved?” (Section 4.1), and “What does a network’s representation tell us about the network’s behaviour?” (Section 4.2). Finally, this chapter will end with a few concluding remarks (Section 4.3).

#### 4.1 *How can the contradiction between behavioural and neural results be resolved?*

EVERY ANALYSIS TOOL provides a unique perspective on the subject of study, but ultimately different tools ought to complement each other. For instance, in biology, the macroscopic study of an ecosystem gives a glimpse of the evolutionary pressures at work, which can then inform microscopic investigations about, for instance, the muscular structure of a bird's wing. Different analysis tools have different advantages, but in the end, one would hope to converge on a comprehensive and unanimous interpretation.<sup>1</sup>

However, in the study of deep neural networks as potential models of human feedforward object recognition, the interpretations following two different analysis approaches could not be more different. On the one hand, many of those using tools from neuroscience are rather enthusiastic, praising CNNs as the leading models of primate object recognition. On the other hand, the majority of those using behavioural analysis tools paint a much darker picture, observing markedly non-human behaviour in standard CNNs. Therefore, a crucial question is: How can the contradiction between behavioural and neural results be resolved?

It may be important to note that in [Geirhos et al. \(2021\)](#), we found promising behavioural results for the latest generation of models such as vision transformers and CNNs trained with up to one billion images; a finding that is in line with [Muttenthaler & Hebart \(2021\)](#) who found that one such model (CLIP) also predicts human behavioural similarity ratings very well. At the time of writing, however, it is not clear whether these models also lead to improvements on neural metrics, thus the following discussion on the contradiction between neural and behavioural results focuses on standard CNNs.

##### *Neural enthusiasm*

A central goal of computational neuroscience is to predict the activity of biological neurons for complex sensory input like natural images. While many neurons at an early stage of processing (such as V1) are characterised fairly well by models based on Gabor filter banks, predicting neural responses for higher layers like the inferior temporal cortex (IT) has long been beyond reach. This changed with a series of landmark investigations using intermediate representations of CNNs to predict neural activity, an approach that outperformed all existing approaches despite using networks purely optimised for categorisation performance rather than neural predictivity. In their 2016 review

<sup>1</sup> In a similar vein, David Marr introduced three levels of analysis applicable to any information processing system, including brains: the *computational* (what is the goal of the system?), the *algorithmic* (how does it achieve this algorithmically?), and the *implementational* level (how is the algorithm implemented physically or biologically?). Studying each of these levels has its own merits, but for a complete understanding, all levels need to be known and, of course, be coherent ([Marr, 1982](#)).



article, Yamins & DiCarlo (2016, p. 364) summarise the far-reaching impact of this line of research: “deep hierarchical neural networks are beginning to transform neuroscientists’ ability to produce quantitatively accurate computational models of the sensory systems, especially in higher cortical areas where neural response properties had previously been enigmatic. Such models have already achieved several notable results, explaining multiple lines of neuroscience data in both humans and monkeys”. This enthusiasm is shared by many others, including Kriegeskorte (2015, p. 417), who noted “surprisingly similar representational spaces” between CNNs and primate brains. Overall, today’s “neural enthusiasm” for CNNs is based on two main feats: their ability to fit neural data, and their ability to predict how neural activity can be increased.

(1.) *Fitting neural data* Yamins et al. (2013, 2014) were the first to report an improved match to neural data when using CNNs. Their approach based on Hierarchical Modular Optimization (HMO), i.e. an optimisation procedure different from today’s predominant stochastic gradient descent, led to a model that accounted for about 50% of explainable IT variance, a substantial leap compared to previous models. Soon, their findings would be corroborated and extended by a number of laboratories (Agrawal et al., 2014; Cadieu et al., 2014; Khaligh-Razavi & Kriegeskorte, 2014; Güçlü & van Gerven, 2015; Cichy et al., 2016; Kubilius et al., 2016). Today, CNNs successfully predict neural data across all stages of the ventral stream. This is not just the case for high-level areas like IT where previous attempts fell short but also for low-level areas like V1. Here, despite the decent predictive performance of established models, CNNs outperform all other approaches slightly but significantly (Cadena et al., 2019).

(2.) *Predicting how to increase neural activity* “All models are wrong, but some are useful”, the aphorism knows (cf. Box, 1976). Therefore, instead of asking whether CNNs as models of human visual perception are right or wrong, good or bad, we might better be asking whether they are *useful*. The usefulness of a model is best assessed by its ability to make (potentially counter-intuitive) predictions. In this regard, CNNs do make a highly counter-intuitive prediction: when trained to predict the firing patterns of neurons for natural image input, CNNs can then be used to predict how input patterns should look like such that neural activity would be increased even further for a certain neuron. Crucially, even for a low-level visual area (mouse V1), these input patterns show complex structures, some of which do not resemble Gabor filters at all—and indeed, when these complex patterns are subsequently presented on a screen, neural activity increases

strongly, more so than control stimuli like Gabors or natural images (Walker et al., 2019).<sup>2</sup>

Similar closed-loop investigations have reported equally promising results for monkey areas V<sub>4</sub> (Bashivan et al., 2019) and IT (Ponce et al., 2019), which speaks to the ability of CNNs not only to fit neural data but also—when trained accordingly—to predict the counter-intuitive effect of an intervention, namely the manipulation of input patterns which elicit higher responses than both standard filters and natural images. Overall, the approach of fitting a machine learning model to reproduce aspects of a black-box system (such as a biological brain) and then testing whether the model's solution sheds light onto the computations of the investigated system is called machine learning system-identification. This approach predates present-day investigations; in the past, it has helped understand various aspects of visual processing (e.g. Wichmann et al., 2005; Kienzle et al., 2009; Macke & Wichmann, 2010), auditory perception (e.g. Schönfelder & Wichmann, 2013) and even bat echolocation (Yovel et al., 2008).

### *Behavioural disappointment*

In contrast to enthusiasm on the representational (neural) side, behavioural investigations comparing humans and CNNs report a number of striking discrepancies: (1.) CNNs and humans often use different image cues. (2.) Consequently, when images are manipulated, they generalise differently. (3.) In addition to differences in the number of errors, error patterns (or the distribution of errors) diverge as well. Those three aspects are described below; Ma & Peters (2020) provide an accessible overview of the many analysis approaches available to a psychophysicist.

(1.) *Features / cues / strategies* There are numerous cases where CNNs use different image cues than humans. For instance, CNNs may latch onto the background of an image, recognising a cow only if the cow is placed in front of a green grass landscape (Beery et al., 2018). Additionally, as we have seen in Geirhos et al. (2019a), CNNs trained on ImageNet preferably detect image textures instead of object shapes—a strategy not usually employed by human observers, who mainly recognise objects by their global shape (Baker et al., 2018; Geirhos et al., 2019a). This finding is corroborated by Doerig et al. (2020), who studied crowding effects and found that in typical CNNs, global aspects of a stimulus do not influence how local aspects are processed—in contrast to humans, where global configuration often shapes local processing, leading to the famous statement “forest before trees” (Navon, 1977, p. 353). Moreover, image regions on which humans rely during

<sup>2</sup>Gabor filters have traditionally been thought to be near-optimal for mouse, monkey and human V<sub>1</sub> neurons. Should those complex patterns that excite mouse or monkey V<sub>1</sub> neurons much more than Gabors also have their counterpart in human visual area V<sub>1</sub>, then one might be able to confirm this with a behavioural experiment in humans, for instance with the paradigm of Watson et al. (1983), which might help in developing a better understanding of the non-linear nature of those neurons.

categorisation differ from those selected by CNNs (Karimi-Rouzbahani et al., 2017).

(2.) *Generalisation* CNNs and humans both reach near-ceiling accuracies when tested on unmanipulated, noise-free colour images. However, as soon as the signal-to-noise ratio decreases through image degradations, standard CNNs typically generalise poorly (Geirhos et al., 2018) and they struggle to cope with large levels of variation (Ghodrati et al., 2014). According to Kheradpisheh et al. (2016b), CNNs may sometimes agree with humans on the *order* of manipulation difficulty (e.g. rotating an image is more difficult than a shift in object position), but CNNs fail to predict how a certain amount of noise should be distributed on an image such that humans wouldn't recognise it (Berardino et al., 2017).

Taken together, one may be tempted to conclude that CNNs simply cannot cope with test images deviating too much from the image statistics seen during training, i.e. that CNNs simply generalise much worse than humans. However, this conclusion would be premature. While CNNs and humans indeed select different image cues even in simple toy experiments (Geirhos et al., 2020a; Funke et al., 2021), CNNs only generalise poorly when those selected cues are affected through an image manipulation. In many cases, CNNs can even recognise images completely unrecognisable to humans—as long as the crucial texture statistics are intact (Brendel & Bethge, 2018). In short: both humans and machines generalise, but they often generalise differently (Geirhos et al., 2020a). For instance, CNNs are known to assign a highly confident prediction to pattern-like input images mostly unrecognisable to humans (Nguyen et al., 2015). Perhaps the most striking difference in generalisation is the adversarial vulnerability of CNNs, which constitutes a major behavioural discrepancy to human visual perception (Szegedy et al., 2013). While humans may sometimes be able to predict the category that a CNN recognises in a seemingly random pattern (Zhou & Firestone, 2019), this holds only under certain restricted experimental assumptions (Dujmović et al., 2020), and there is no evidence to suggest that humans can be fooled by adversarial examples in general. Overall, striking generalisation differences between human and machine vision are a key factor behind the “behavioural disappointment”.<sup>3</sup>

(3.) *Error patterns* As described above, depending on the image manipulation, CNNs and humans make a different *number* of errors—but do they differ only in accuracy or also in terms of their *error patterns*? A few studies using different methods have consistently shown that this is indeed the case: humans make different errors than standard

<sup>3</sup> In spite of some rather “disappointing” behavioural results, it is important to keep in mind that deep neural networks perform much better than previous pre-deep-learning algorithms in many challenging problem settings.

CNNs (Kheradpisheh et al., 2016a; Geirhos et al., 2017, 2018, 2020b; Rajalingham et al., 2018). On a coarse level, CNNs are reported to capture category-level confusion, but not image-level confusion patterns (Rajalingham et al., 2018). This study differs in a number of experimental choices from the paradigm used in this thesis (synthetic instead of natural objects, online crowdsourcing instead of controlled lab environment, object presentation not masked, total number of objects limited to 24 stimuli, repeated presentation of the same object to an observer, target object displayed on choice screen in every trial); additionally, responses were aggregated across participants. Despite these experimental differences, the core finding is shared between the methodologically highly distinct studies by Rajalingham et al. (2018) and Geirhos et al. (2020b): image-level error patterns between standard CNNs and humans differ. In our interpretation, this behavioural analysis indicates that there are important functional differences between current human and machine vision.

**BOTH ANALYSIS APPROACHES**—the neuroscientific one and the behavioural one—investigate the same subject of study, namely CNNs as potential models for human visual object recognition. Resolving the contradiction between those two interpretations will be important since it has direct implications on the role of deep learning for the brain sciences: are CNNs best used only as a tool, as a model, or perhaps even not at all? Besides, while there are beautiful cases of neuroscientific and behavioural experiments working hand in hand to investigate a certain phenomenon, one cannot help but wonder whether they will always be the right partners to investigate biological brain function if they lead to contradicting interpretations even for comparatively simple artificial neural networks built from just a few canonical primitives. Currently, a few different hypothesis options are conceivable to explain the gap between neuroscientific enthusiasm and behavioural disappointment.

*Hypothesis 1: The behavioural glass is half empty, the neural one half full*

This hypothesis states that the seemingly contradicting neuroscientific enthusiasm and behavioural disappointment are first and foremost a matter of interpretation. Behavioural and neuroscientific results are computed on different scales and metrics, thus mathematically relating one to another is challenging—instead, one typically resorts to comparing them using natural language. This in turn opens the comparison to influences from the authors' personal perspectives. Expectations for the usefulness of CNNs as models of higher-level cogni-

tion may differ, and consequently, so might interpretations—but essentially, irrespective of whether the glass is perceived as half empty or half full, it would still be the same amount of water in the glass.

A prediction following from this hypothesis is that there should be a positive rank-order correlation between measurements of brain and behavioural fit: if both approaches measure the same underlying phenomenon, then better models of human behaviour should also be better models of neuroscientific data. Putting this prediction to a test, this does not seem to be the case. On average, error consistency (a behavioural metric) is not correlated with brain measurements at all, as can be seen in Figures SF 9–12 of Geirhos et al. (2020b). Likewise, on the Brain-Score benchmark (Schrimpf et al., 2018), neural and behavioural scores are only weakly linked, if at all (see Figure 4.1). This indicates that irrespective of potential nuances in interpretation (which may well be at play), there is still an underlying discrepancy between neural and behavioural analyses that cannot be accounted for by the “glass half empty, half full” hypothesis.

#### *Hypothesis 2: Good representation, poor objective*

This hypothesis<sup>4</sup> resolves the contradiction between an excellent representation and poor behaviour by assuming that, essentially, both points of view are accurate: CNNs acquire a human-like intermediate representation, but the last few layers of the network are predominantly shaped by the objective function which leads to non-human behaviour. This explanation is somewhat supported by Kornblith et al. (2020), who observed that networks trained with different loss functions hardly differed in approximately the first two-thirds of their representation, while in the last third of the layers representations diverged substantially, which resulted in very different network behaviour depending on the specific loss function—consistent with the “good representation, poor objective” hypothesis. In contrast, when the same loss function is used, even initially different high-level representations might end up in a very similar regime at the output layer, as shown in Figure 4.2 plotting an embedding of different networks differing only in their random seed—a finding that might explain why standard networks consistently end up with very similar behaviour (Geirhos et al., 2020b).

The plausibility of this hypothesis is further corroborated by a thought experiment showing how the output-level behaviour of a representation is determined by the last few layers: Take a CNN with a perfectly human-like representation up to layer  $N$ . The behaviour of the  $M$ -layer network, however, can be influenced to a large degree by the layers  $N+1, \dots, M$ . For instance, even a single last layer could assign

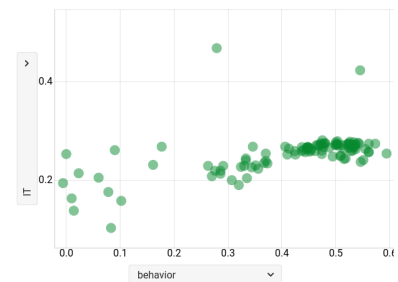


Figure 4.1: Scatter plot of neural (Inferior temporal cortex, IT) vs. behavioural scores for models tested on Brain-Score. If anything, there is a very weak correlation between those two metrics.

<sup>4</sup> This hypothesis was informally put forward as a possible explanation by Tim Kietzmann (personal communication).

zero weight to the input it receives from the previous layer and instead use the bias to always predict an arbitrary class irrespective of the input. Or, in another example of what might happen after layer  $N$ , if there are no constraints on either the width or the depth of layers  $N+1, \dots, M$ , they can represent any function according to the universal function approximation theorems (Hornik et al., 1989). In short: even with the best intermediate representation, the behaviour of the network can be determined by the last layer(s), and those are known to be shaped by the objective function. Therefore, a suboptimal objective function might lead to non-human behaviour.

Experimentally, based on this hypothesis one would predict that the neuroscientific fit to primate neural data is best for intermediate layers and starts to decline strongly at exactly the point where Kornblith et al. (2020) identified diverging representations depending on the choice of objective function. Furthermore, this hypothesis would explain why Kubilius et al. (2016) observed a human-like shape representation in CNNs whereas the experiments presented earlier in this thesis clearly show that CNN behaviour is determined by object texture rather than object shape (Geirhos et al., 2019a).

And yet, despite the apparent appeal of this hypothesis, one crucial question remains: Given the well-documented preference of CNNs to exploit shortcuts whenever they can (Geirhos et al., 2020a), often learning nothing but the most predictive feature, why would CNNs acquire a human-like representation in the first place? If a good texture representation is sufficient to solve the task of object recognition, why would CNNs learn a human-like shape representation, too? If detecting certain high-frequency patterns correlated with object class (Jo & Bengio, 2017; Ilyas et al., 2019) is sufficient to recognise objects, why would CNNs learn a human-like object representation, too? In this regard, a human-like intermediate representation would be an epiphenomenon of training: during training, a human-like representation is caused (for some reason), but this does not cause human-like behaviour at the output level. Essentially, if the “good representation, poor objective” hypothesis were true, this would mean that we have solved one riddle (why there is neuroscientific enthusiasm yet behavioural disappointment) but ended up with another one—why even a poor choice of objective function along with the preference of CNNs to learn shortcuts would still lead to the development of a human-like representation.

### *Hypothesis 3: A matter of stimuli*

A completely different hypothesis asks whether broad comparisons between neural and behavioural studies might be strained simply since both approaches commonly use different stimulus classes. For stan-

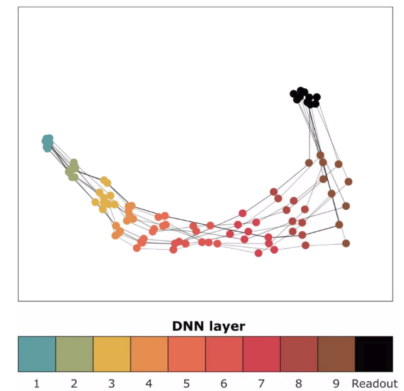


Figure 4.2: How similar are the representations of 10 different training instances of a convolutional neural network across layers? This plot shows a 2D embedding based on multidimensional scaling (Shepard, 1962; Kruskal, 1964), with dimension 1 on the abscissa against dimension 2 on the ordinate. The analysis is based on all pairs of distances between representational dissimilarity matrices (Kriegeskorte et al., 2008); the figure is adapted from Figure 3 of Mehrer et al. (2020) to include the readout layer (Tim Kietzmann, personal communication). Network differences grow with increasing depth but converge again at the very last (readout) layer.

standard non-distorted images of natural objects, behavioural object recognition accuracies are well within the range of CNN accuracies. Many behavioural discrepancies only become apparent when switching from non-distorted images to images with reduced signal-to-noise ratio, where certain image cues are systematically altered: a common practice in behavioural generalisation experiments. In contrast, landmark studies on the representational similarities between primate and machine vision are based on standard objects, often placed against a random background but not systematically distorted. In other words, could it be the case that comparing behavioural and neural studies would be a comparison of apples and oranges? Can the different interpretations simply be accounted for by the potential confound that many behavioural interpretations are based on out-of-distribution generalisation, whereas neural experiments mostly focus on standard objects closer to the training distribution?

Interestingly, [Xu & Vaziri-Pashkam \(2021\)](#) recently analysed the proportion of variance in human brains that a range of different CNNs account for—when using either standard, low-pass filtered, or high-pass filtered images. This approach bridges the gap between common neural and behavioural experimental approaches by using distorted stimuli for the analysis of neural representations.<sup>5</sup> However, while the authors find a substantial reduction in explained variance when switching from real-world objects to artificial objects, the differences between explained variance for standard and spectrum-altered images are much smaller. Although more experiments and studies will be needed, these findings cast doubt on the hypothesis that discrepancies from neural and behavioural analyses can be fully accounted for by a mere difference in experimental stimuli.

<sup>5</sup> Of course, neuroscience has a long history of using controlled artificial stimuli, see e.g. [Rust & Movshon \(2005\)](#)—however, many prominent studies comparing primate and CNN representations are instead based on undistorted object images.

#### *Hypothesis 4: A matter of baselines*

Whether or not one is thrilled or disappointed by the performance of CNNs naturally depends on the baseline against which they are compared. Measured against an ant, even a mouse appears as large as a giraffe: but are ants the right choice of comparison? Similarly, in neuroscience, trained CNNs are routinely compared against the predictive performance of untrained models, begging the question of whether this might unduly overestimate their performance.

In behavioural object recognition experiments, an untrained baseline would be trivially easy to beat: any machine algorithm performing better than chance would be closer to human performance. But is this sufficient to assert that the algorithm uses somewhat human-like computations? Not necessarily so. In our error consistency analysis ([Geirhos et al., 2020b](#)), we compared CNNs to *performance-matched*

baselines: do we observe consistency to human behaviour beyond what can be expected by any algorithm that can do the task? This distinction turned out to be crucial—all CNNs perform better than the random baseline, but only a few better than the performance-matched baseline.

In those neural experiments where baselines are used, untrained baselines are still the standard—presumably mostly because it is not easy to define an appropriate performance-matched baseline. For instance, this might involve defining a random intermediate representation subject to the constraint that the output layer matches the object recognition performance of the CNN in question. This means that there might currently be a fair chance of overestimating the predictive accuracy of CNNs for neural data. Trained CNNs certainly predict neural data better than untrained ones (Yamins et al., 2014; Güçlü & van Gerven, 2015), but only by about 5–10% accuracy difference according to Saxe et al. (2020). What if a substantial portion of this already small advantage were to vanish when comparing against a performance-matched baseline?<sup>6</sup> As a cautionary tale on the importance of appropriately strong baselines, Macke & Wichmann (2010) showed that even for a simple two-class problem, human-to-model correlations can be driven by high accuracies on both classes. In essence, if both a model and a human observer achieve an accuracy of 0.9 for each of the classes, their overall responses will be strongly correlated simply because they both can do the task fairly well, so this cannot be used as a criterion to assess whether they use similar processing mechanisms. When conditioning on a ground truth class, however, their correlation will only be high if they indeed make the same mistakes (which would be a much stronger indication of similar processing mechanisms).

In the light of this hypothesis, the contradiction between neuroscientific enthusiasm and behavioural disappointment would be resolved if one can experimentally show that CNNs perform better than a random baseline, but only slightly better than a performance-matched baseline.

### *Taken together*

Behavioural and neural analyses have led to two opposing viewpoints, with important implications for the role of neural networks as models of human perception. We have seen that these discrepancies are deeper than a mere matter of interpretation and that there are a number of hypotheses that may explain the contradiction. Ultimately, only experiments will bring answers—but I hope to have convinced you that it is a question worth asking.

<sup>6</sup> This hypothesis was informally put forward by Felix Wichmann (personal communication).



#### 4.2 *What does a network’s representation tell us about the network’s behaviour?*

WE HAVE SEEN a number of different hypotheses on where the contradiction between behavioural and neural results may originate from. The importance of solving this problem is clear—not only are we currently lacking an agreed-upon understanding of the value that CNNs bring to brain and behavioural research, but this issue also points to a much bigger context: It seems that we currently lack a good understanding of what a network’s (intermediate) representation tells us about the network’s behaviour. One usually expects two decision-makers (such as two models, or a model and a human observer) with a similar intermediate representation to also show similar behaviour. For instance, Kornblith et al. (2020, p. 1) write that studies seek to “understand the *behavior* of neural networks by *comparing representations* between layers and between different trained models”, Morcos et al. (2018, p. 1) state that “*comparing different neural network representations* and determining how representations evolve over time remain challenging open questions in our understanding of the *function* of neural networks”, and Blanchard et al. (2019, p. 5404) “hypothesize that networks exhibiting brain-like activation behaviour [as measured by *representational dissimilarity matrices*] will demonstrate brain-like characteristics, e.g., stronger *generalization capabilities*”. In all three quotes, emphasis was added to point out how easily expectations for (intermediate) model representations become expectations for model function and behaviour. However, whether this expectation is truly warranted or whether it is just one of the untested assumptions in a field that eventually become “common knowledge” without ever being questioned remains to be determined.<sup>7</sup>

#### *Understanding the function of an intermediate representation: receptive and projective fields*

Which statements can we make about the relationship between an intermediate representation and behaviour? From a slightly different angle, this question has been the subject of debate in computational neuroscience as well. Here, instead of assessing the role of a layer, one is often interested in understanding the function of an individual neuron. The *receptive field* of a neuron is determined by the chain of neurons from which the neuron receives its input, similar to how a neural network layer’s activation depends on the preceding layer’s activation. For quite some time, it was—at least implicitly—assumed that this would be all there is to know, that the function of a neu-

<sup>7</sup> In an effect used by rhetoricians and politicians alike, people are more likely to believe in the validity of a statement if the statement is repeated multiple times—irrespective of whether the statement is true or false (Hasher et al., 1977).

ron can be completely characterised by understanding the neuron's receptive field (i.e. the patterns to which the neuron responds most strongly). For instance, in the classic case of [Hubel & Wiesel \(1959\)](#), if one has discovered that a neuron most strongly responds to oriented bars, then it is easy to believe that one has succeeded in understanding the role of that neuron. However, in an early example of the value of computational modelling with artificial neural networks, [Lehky & Sejnowski \(1988\)](#) showed that this perspective is incomplete. Even for a simple three-layer network, it is just as important to consider the *projective field* of a neuron, which is the set of neurons that receive input from the neuron in question. Depending on connectivity and weight patterns, a neuron with fixed receptive field can serve many different purposes. In the case of [Lehky & Sejnowski \(1988\)](#), intermediate neurons appeared to detect edges, but examining their projective field revealed that they were being used to extract surface curvature from shading. Relating this simple yet fundamental insight back to the representation vs. behaviour discussion, it seems clear that the role of a network's intermediate representation cannot fully be understood if one does not take the projective field into account.

This might sound rather obvious, but even so, it sometimes appears to be underappreciated in the context of deep learning, particularly when it comes to neural network interpretability methods. Two prominent interpretability approaches are *feature visualisation* and *attribution* techniques (e.g. [Olah et al., 2018](#)). These two methods ask related yet different questions. Attribution techniques are concerned with understanding, for instance, how much a certain input pixel contributes to the network's output. In contrast, feature visualisation is asking what a unit (or channel) is selective for, much like the study of receptive fields in neuroscience. Yet in practice, feature visualisations sometimes take on a similar role as attribution techniques when intermediate visualisations are taken as evidence for a network's decision, such as in the following common pattern: "How did the network recognise this dog? Intermediate unit  $x$  is highly activated, a unit that is selective for dog snouts (according to feature visualisation), therefore 'dog' is predicted by the network."

As plausible as this interpretation may seem at first glance, one cannot decide whether it is correct without examining the projective field of unit  $x$ . In the extreme case, this unit's projective field may consist only of zero-weight connections, rendering the unit activation irrelevant to the network's output decision: "it is obvious that a neuron without any output cannot have a computational function" ([Sejnowski, 2006](#), p. 396). To give a prominent example where the role of the projective field is neglected, network dissection ([Bau et al., 2017](#))—a method based exclusively on the receptive field of a unit, without

any consideration of its projective field—has been described as aiming at a “comprehensive functional understanding of the model” in a well-known tutorial and survey article (Montavon et al., 2018, p. 2), where ‘functional understanding’, according to the authors, refers to a characterisation of “the model’s black-box behavior, without however trying to elucidate its inner workings or shed light on its internal representations”. In contrast, Lehky & Sejnowski (1988, p. 454) already pointed out that a functional understanding of a neuron “appears to require not only knowledge of the pattern of input connections forming its receptive field, but also knowledge of the pattern of output connections, which forms its projective field”.

To make matters worse, the term ‘function of a neuron’ is not always used in the same way, which can be the source of additional confusion: Schubert et al. (2021, p. 1) state that “feature visualization allows us to establish a causal link between each neuron and its function”. Here, it seems plausible to assume that ‘function’ is used in the mathematical sense, as a complete characterisation of the input-output relationship of a neuron (if it were used in the traditional neuroscientific sense, the statement would be wrong since feature visualization is blind to a neuron’s projective field). In this sense, if the input is known, the function determines the network’s output, just like knowing function  $f(x) = x^2$  and input  $x = 4$  can be used to determine the output  $f(4) = 16$ . In contrast, neuroscientists typically use the term “function of a neuron” in a broader sense, as in “the function that a neuron fulfils” within a certain context, like a nervous system (abstracted away from the question of *how* the neuron may implement this). Here, the projective field plays an important role: whether a sensory neuron may connect to other sensory neurons, or to motor neurons, or both, makes a crucial difference in terms of the neuron’s function. Taken together, irrespective of whether one would like to understand the function of a neuron within a brain or the function of an intermediate unit within a neural network—it is crucial to assess both the receptive as well as the projective field of that neuron or unit. The same holds for collections of neurons or units, and for entire network layers.

#### *Extreme cases: the role of the projective field*

We are interested in understanding how much a network’s intermediate representation determines the network’s behaviour. Since the output behaviour of a network naturally depends upon the computations that happen in-between such an intermediate representation and the output layer (in other words, on the projective field of the intermediate representation), we can start to understand the relationship between representation and behaviour by considering two extreme cases

of projective fields. To this end, we ask: How much is the output of an  $M$ -layer network influenced, constrained or even determined by the representation at layer  $N$ , for some  $N < M$ ? As we will see, it is possible that the final network output at layer  $M$  is *completely unrelated* to the intermediate representation at layer  $N$ , but it is also possible that the output of layer  $M$  is *completely determined* (for instance, identical) to the output of layer  $N$ .

*Completely unrelated.* The layers of standard neural networks form a Markov chain (Tishby & Zaslavsky, 2015), where the output of layer  $l_i$  is computed as follows:  $l_i = \text{ReLU}(w \cdot l_{i-1} + b)$ . Here,  $w$  refers to the weight matrix and  $b$  to the bias vector. According to the data processing inequality (Cover, 1999), mutual information  $I$  along subsequent processing stages  $A$ ,  $B$  and  $C$  (here: network layers) cannot increase:  $I(A;C) \leq I(B;C)$ . Therefore, it is clear that once information about the input is lost in layers  $1, \dots, N$ , then later layers will never be able to re-gain this information. In this sense, the network's output (or behaviour) will be clearly constrained by the amount of information that the intermediate representation has kept or discarded. In an extreme case, one can achieve mutual information of zero bits between the representation at layer  $N$  and the representation at layer  $M$ , simply with a single layer somewhere in-between  $N$  and  $M$  where the weight matrix  $w$  is a zero matrix. Then, all information provided by the preceding layer is lost, and the network output will simply be  $\text{ReLU}(b)$ , i.e. a rectified version of the constant bias vector  $b$  which does not depend on the input at all. In effect, this would mean that all inputs are mapped to an arbitrary constant output, irrespective of which input patterns the network is exposed to. Therefore, in this extreme case, the output is *completely unrelated* to the representation at intermediate layer  $N$ .

*Completely determined.* In contrast, it is equally possible to construct a case where the output is *identical* to the representation at intermediate layer  $N$ . A prerequisite is that there is no bottleneck in-between layers  $N$  and  $M$ , i.e. that all layers from  $N + 1$  to  $M$  have at least  $k$  units, where  $k$  is equal to the number of units in output layer  $M$ . If this condition is fulfilled (which is the case for nearly all standard architectures, e.g. those performing well on ImageNet), then it is possible to create a bijective mapping between  $k$  (arbitrary) units of layer  $N$  to  $k$  units of layer  $N + 1$  to  $k$  units of layer  $N + 2$  and so forth, unit one maps  $k$  units of layer  $M - 1$  to  $k$  units of layer  $M$ . This can be achieved by using weights of 1 between units that form this mapping, weights of zero between units that are not part of this mapping, and biases of zero in general. Effectively, this means that the activations of  $k$

output units are completely identical to the output of  $k$  units in layer  $N$  (for standard units with a ReLU activation function). Therefore, in this artificial extreme case, the output is *completely determined* by the representation at intermediate layer  $N$ .

### *Relating representational to behavioural metrics*

As highlighted by those two extreme cases, in principle the network output can be completely identical to some intermediate representation or completely unrelated to it—depending on the projective field of the intermediate layer. In practice, the situation will likely be much more nuanced, falling somewhere along the spectrum spanned by those extremes—but where exactly remains to be understood.

In order to find out more about the relationship between representation and behaviour in standard trained neural networks, one could test whether networks that are similar in terms of their representation are also similar in terms of their behaviour. For instance, one could make a scatter plot of networks in terms of their respective similarities according to representational vs. behavioural metrics. In terms of representational metrics, sensible choices could include a few widely used metrics like RSA (Kriegeskorte et al., 2008), SVCCA (Raghu et al., 2017), CCA (Morcos et al., 2018) and CKA (Kornblith et al., 2019); in terms of behavioural metrics it might be interesting to look at overall accuracy (do two networks make a similar number of errors?), error consistency (Geirhos et al., 2020b, asking whether two networks make similar errors), along with other behavioural similarity metrics. Having tested those networks, one could then assess the relationship of representational vs. behavioural similarity on a scatter plot (potentially grouped by architecture types), and assess whether there is a positive rank-order correlation between the two.

### *Realistic extreme cases*

After investigating what happens in practice, i.e. to which degree representational and behavioural metrics are related for common models, a logical next step would then be to set the obtained results into context by understanding *realistic* extreme cases. The two theoretical extreme cases presented above may be instructive, but they are not plausible or realistic since network performance would be completely destroyed, while in practice, we know that networks have decent test performance. This leads to the following two questions: Given fixed intermediate representations, and a target accuracy, what is the maximal behavioural difference that we can possibly achieve? Conversely, given a fixed degree of behavioural similarity between two networks (including, but potentially not limited to, similar target accuracies),

what is the maximal representational difference that we can possibly achieve?

While to the best of my knowledge none of these questions has been studied so far, there are a few building blocks on which further investigations could build. Changing as much about the internal representation as possible while keeping performance at a high level is related to the concepts of *random classifiers* and *random intermediate layers*.

*Random classifier* Hoffer et al. (2018) showed that it is possible to set the linear classifier of a network—usually a fully-connected layer—to a random but orthonormal projection which is then kept fixed and never updated during training. On a variety of architectures trained on ImageNet, this leads to comparable performance even though much fewer parameters are trainable—in the case of ShuffleNet (Zhang et al., 2018), the last fully connected layer even accounts for more than half of the model’s parameters.

Usually, the motivation behind keeping a proportion of network weights fixed is of engineering nature, as one can use this to increase the speed of training or reduce a network’s sample complexity. However, there are epistemic consequences as well. Much of what is “commonly known” (or, what might sometimes be a more appropriate description, “commonly assumed”) about deep learning is based on the notion that deep neural networks are powerful function approximators which *learn* a suitable representation given enough training data. The concept of a random layer never updated during training contrasts with the intuition that learning always plays such a crucial role.

*Random intermediate layers* Just like it is possible to use a random final layer (i.e. a random classifier), it is also possible to insert a certain (and often surprisingly high) degree of randomness into intermediate network layers. Often, one starts by freezing randomly assigned initial weights and then learning a standard linear classifier on top of this (partly or fully) fixed random representation. The approach of using one or more layers with a fixed random projection has a long history in machine learning. For instance, even the classic *perceptron* by Rosenblatt (1958) contains a randomly connected layer. Since then, the use of random connections has repeatedly been explored under different names: the *Gamba perceptron* (Minsky & Papert, 1969), *Extreme Learning Machines* (Huang et al., 2006)—a term that sparked great controversy since it essentially rebranded an old idea (Wang & Wan, 2008), but nonetheless sparked renewed interest in the concept of a random layer—furthermore, there are *Random Kitchen Sinks* (Rahimi et al., 2007; Rahimi & Recht, 2008), and, in the context of recurrent systems, *Echo State Networks* (Jaeger, 2002) along with *Liquid State Machines* (Maass

et al., 2002) and other variants of what today is usually termed *Reservoir Computing* (Lukoševičius & Jaeger, 2009). An accessible overview of the history of random connections in neural networks can be found in Shen et al. (2020), who introduce *Reservoir Transformers* as one of the most recent examples of using fixed random instead of trained weights. While it may sound counter-intuitive to learn useful representations without actually learning much (i.e. by keeping randomly chosen weights fixed), there is theoretical support for this idea: Cover (1965) showed that high-dimensional non-linear transformations increase the chances of a representation becoming linearly separable, indicating that even random transformations can be useful.

What does this mean for the relationship between network representation and network behaviour? Essentially, if it could be shown that a network can reach human-like behaviour even if a large fraction of its weights remain random and are never updated during training, then the specific representation (whether it is fully trained or mostly random) would matter less than previously thought. This would also imply that “learning” might be overrated: much of the magic would happen through cascades of non-linear and random linear transformations. While this remains pure speculation and contrasts with our finding of the decisive importance of data (Geirhos et al., 2019a, 2021), it has been argued that learning might indeed play a larger role in machines than it should. By contrast, biological brains often rely on innate mechanisms which scaffold and speed up learning. According to Zador (2019, p. 1), “most animal behavior is not the result of clever learning algorithms—supervised or unsupervised—but is encoded in the genome”. The human genome is about six orders of magnitude too small to store all connections in the brain (Zador, 2019). This means that the genome is quite clearly not a lookup table in which the perfect connections and weights are stored, which fits well with the intuition that random connections are valuable as well.

### *Taken together*

Motivated by the fact that brains and CNNs appear to have “surprisingly similar representational spaces” (Kriegeskorte, 2015, p. 417) while at the same time there are “marked behavioural differences between ImageNet-trained CNNs and human observers” (Geirhos et al., 2019a, p. 9), we asked whether there is more to understand in the relationship between a network’s representation and its behaviour (or function). Typical analyses of intermediate representations are blind to the projective field. Depending on the projective field, intermediate representations can be either completely unrelated or completely identical to the network output—in practice, typical networks will most

likely fall somewhere along this broad spectrum, but it remains to be investigated where exactly, for instance by correlating representational and behavioural similarity metric results for a range of standard networks. In a second step, one could then seek to understand whether networks of a certain performance level always end up in a specific region of this space. Here, it might be helpful to test networks with a certain degree of randomness (such as a random classifier or random intermediate layers). These partly random networks strongly deviate from standard models in terms of their processing, and likely in their representation, but not at the cost of a detrimental effect on network performance, which makes them ideal candidates for such a comparison. Even though numerous representational and behavioural analyses have been conducted separately, we know surprisingly little about how a network's intermediate representation is linked to network behaviour.

### 4.3 *Concluding remarks*

WE CAN CONSIDER OURSELVES LUCKY to live in a world where machine learning is possible. Perception is a process of inferring—typically reasonably accurate—hypotheses about the world around us. If this world were unpredictable, neither human nor machine perception would be able to form accurate hypotheses—and as a consequence, it is unlikely that either machines or humans would have evolved in the first place. Successful hypotheses require predictability, and they are tailored to the world we live in. As Gregory (1967, p. 174) puts it, “when we are transferred to an alien or bizarre environment, where our filing cards [i.e. our assumptions or hypotheses] are inappropriate, we interpret the images in the eyes according to principles found reliable in the previous, familiar world—but now they may systematically mislead and then perception goes wrong. Space travellers beware!”

Given the importance of appropriate assumptions for successful perception, it is remarkable how little is understood about the assumptions that modern deep neural networks make—and how they relate to those of human perceptual decision-making. In this thesis, I conducted a functional comparison of human and machine behaviour on visual object recognition. Starting with the simple question “How do models recognise objects?”, investigating models for both object classification and object detection revealed a texture bias (in contrast to the textbook explanation of neural network object recognition). Overcoming this texture bias through data augmentation induces a human-like shape bias instead, and leads to improved robustness towards image distortions. In a second step, I asked how assumptions vary across



models, in other words, “How do models differ from one another?”. Using error consistency as a behavioural metric, we found remarkable similarities between otherwise very different models, such as between recurrent and feedforward models and between supervised and self-supervised models. By contrast, human observers also made highly consistent errors with other human observers—but the consistency between models and humans was only slightly beyond chance, indicating that they are making different assumptions. Why this might be the case was the topic of a perspective article on shortcut learning, a concept that unifies many of deep learning’s failures. Since shortcut learning leads to deceptively high performance on standard test sets, we argued that out-of-distribution testing will need to take on a much more prominent role. Consequently, a comprehensive comparison of human and machine out-of-distribution generalisation was the topic of my last project, which was able to report partial success in closing the gap between human and machine vision.

ON A BROADER LEVEL, my findings indicate that our understanding of machine decision-making is riddled with (often untested) assumptions, but they can be put on a solid empirical footing through rigorous quantitative experiments and functional comparisons to human decision-making: for when humans better understand machines, we will be able to build machines that better understand humans—and the world we all share.



# Bibliography

- Agrawal, P., Stansbury, D., Malik, J., and Gallant, J. L. Pixels to voxels: Modeling visual representation in the human brain. *arXiv preprint arXiv:1407.5104*, 2014.
- Baker, N., Lu, H., Erlikhman, G., and Kellman, P. J. Deep convolutional networks do not classify based on global object shape. *PLoS Computational Biology*, 14(12):e1006613, 2018.
- Bashivan, P., Kar, K., and DiCarlo, J. J. Neural population control via deep image synthesis. *Science*, 364(6439), 2019.
- Bau, D., Zhou, B., Khosla, A., Oliva, A., and Torralba, A. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6541–6549, 2017.
- Beery, S., Van Horn, G., and Perona, P. Recognition in terra incognita. In *Proceedings of the European Conference on Computer Vision*, pp. 456–473, 2018.
- Berardino, A., Laparra, V., Ballé, J., and Simoncelli, E. P. Eigen-distortions of hierarchical representations. In *Advances in Neural Information Processing Systems*, 2017.
- Blanchard, N., Kinnison, J., RichardWebster, B., Bashivan, P., and Scheirer, W. J. A neurobiological evaluation metric for neural network model search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5404–5413, 2019.
- Bojchevski, A., Shchur, O., Zügner, D., and Günnemann, S. NetGAN: Generating graphs via random walks. In *International Conference on Machine Learning*, pp. 610–619. PMLR, 2018.
- Borges, J. L. *On exactitude in science. Collected Fictions*, volume 325. New York: Penguin, 1998.
- Borowski, J., Zimmermann, R., Schepers, J., Geirhos, R., Wallis, T. S. A., Bethge, M., and Brendel, W. Exemplary natural images explain CNN activations better than feature visualizations. In *International Conference on Learning Representations*, 2021.
- Box, G. E. Science and statistics. *Journal of the American Statistical Association*, 71(356): 791–799, 1976.
- Brendel, W. and Bethge, M. Approximating CNNs with bag-of-local-features models works surprisingly well on ImageNet. In *International Conference on Learning Representations*, 2018.
- Buckner, C. The Comparative Psychology of Artificial Intelligences, May 2019. URL <http://philsci-archive.pitt.edu/16034/>.
- Cadena, S. A., Denfield, G. H., Walker, E. Y., Gatys, L. A., Tolias, A. S., Bethge, M., and Ecker, A. S. Deep convolutional models improve predictions of macaque V1 responses to natural images. *PLoS Computational Biology*, 15(4):e1006897, 2019.
- Cadiou, C. F., Hong, H., Yamins, D. L., Pinto, N., Ardila, D., Solomon, E. A., Majaj, N. J., and DiCarlo, J. J. Deep neural networks rival the representation of primate IT cortex for core visual object recognition. *PLoS Computational Biology*, 10(12):e1003963, 2014.

- Campbell, M., Hoane Jr, A. J., and Hsu, F.-h. Deep Blue. *Artificial Intelligence*, 134(1-2): 57–83, 2002.
- Cichy, R. M., Khosla, A., Pantazis, D., Torralba, A., and Oliva, A. Deep neural networks predict hierarchical spatio-temporal cortical dynamics of human visual object recognition. *arXiv preprint arXiv:1601.02970*, 2016.
- Cover, T. M. Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE Transactions on Electronic Computers*, pp. 326–334, 1965.
- Cover, T. M. *Elements of information theory*. John Wiley & Sons, 1999.
- D’Amour, A., Heller, K., Moldovan, D., Adlam, B., Alipanahi, B., Beutel, A., Chen, C., Deaton, J., Eisenstein, J., Hoffman, M. D., et al. Underspecification presents challenges for credibility in modern machine learning. *arXiv preprint arXiv:2011.03395*, 2020.
- Descartes, R. Treatise on man. *The Nature of Life: Classical and Contemporary Perspectives from Philosophy and Science (published in 2010)*, pp. 15–20, 1662.
- Doerig, A., Bornet, A., Choung, O.-H., and Herzog, M. H. Crowding reveals fundamental differences in local vs. global processing in humans and machines. *Vision Research*, 167:39–45, 2020.
- Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.
- Dujmović, M., Malhotra, G., and Bowers, J. S. What do adversarial images tell us about human vision? *Elife*, 9:e55978, 2020.
- Felleman, D. J. and Van Essen, D. C. Distributed hierarchical processing in the primate cerebral cortex. *Cerebral Cortex*, 1(1):1–47, 1991.
- Frith, U. Fast lane to slow science. *Trends in Cognitive Sciences*, 24(1):1–2, 2020.
- Funke, C. M., Borowski, J., Stosio, K., Brendel, W., Wallis, T. S., and Bethge, M. Five points to check when comparing visual perception in humans and machines. *Journal of Vision*, 21(3):16–16, 2021.
- GDPR. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing (Directive 95/46). *Official Journal of the European Union (OJ)*, 59(1-88):294, 2016.
- Geirhos, R., Janssen, D. H., Schütt, H. H., Rauber, J., Bethge, M., and Wichmann, F. A. Comparing deep neural networks against humans: object recognition when the signal gets weaker. *arXiv preprint arXiv:1706.06969*, 2017.
- Geirhos, R., Medina Temme, C. R., Rauber, J., Schütt, H. H., Bethge, M., and Wichmann, F. A. Generalisation in humans and deep neural networks. In *Advances in Neural Information Processing Systems*, 2018.
- Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., and Brendel, W. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2019a.
- Geirhos, R., Rubisch, P., Rauber, J., Medina Temme, C. R., Michaelis, C., Brendel, W., Bethge, M., and Wichmann, F. A. Inducing a human-like shape bias leads to emergent human-level distortion robustness in CNNs. *Journal of Vision*, 2019b.
- Geirhos, R., Jacobsen, J.-H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., and Wichmann, F. A. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2: 665–673, 2020a.

- Geirhos, R., Meding, K., and Wichmann, F. A. Beyond accuracy: quantifying trial-by-trial behaviour of CNNs and humans by measuring error consistency. In *Advances in Neural Information Processing Systems*, 2020b.
- Geirhos, R., Narayanappa, K., Mitzkus, B., Bethge, M., Wichmann, F. A., and Brendel, W. On the surprising similarities between supervised and self-supervised models. In *NeurIPS Workshop on Shared Visual Representations in Human & Machine Intelligence*, 2020c.
- Geirhos, R., Narayanappa, K., Mitzkus, B., Thieringer, T., Bethge, M., Wichmann, F. A., and Brendel, W. Partial success in closing the gap between human and machine vision. In *Advances in Neural Information Processing Systems*, 2021.
- Geirhos, R., Narayanappa, K., Mitzkus, B., Thieringer, T., Bethge, M., Wichmann, F. A., and Brendel, W. The bittersweet lesson: data-rich models narrow the behavioural gap to human vision. *Journal of Vision*, 2022.
- Ghodrati, M., Farzmaadi, A., Rajaei, K., Ebrahimpour, R., and Khaligh-Razavi, S.-M. Feedforward object-vision models only tolerate small image variations compared to human. *Frontiers in Computational Neuroscience*, 8:74, 2014.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, 2014.
- Goodman, B. and Flaxman, S. European Union regulations on algorithmic decision-making and a “right to explanation”. *AI Magazine*, 38(3):50–57, 2017.
- Gregory, R. L. Will seeing machines have illusions? *Machine Intelligence IV*, 1967.
- Grundkiewicz, R. and Junczys-Dowmunt, M. Near human-level performance in grammatical error correction with hybrid machine translation. *arXiv preprint arXiv:1804.05945*, 2018.
- Güçlü, U. and van Gerven, M. A. Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. *Journal of Neuroscience*, 35(27):10005–10014, 2015.
- Hall, T. and Yoon, E. Searching for new ideas in the curious things your customers do, 2017. URL <https://hbr.org/2017/04/searching-for-new-ideas-in-the-curious-things-your-customers-do>.
- Hamming, R. You and your research. *Transcription of the Bell Communications Research Colloquium Seminar*, 1986.
- Hartmann, P., Ramseier, A., Gudat, F., Mihatsch, M., Polasek, W., and Geisenhoff, C. Das Normgewicht des Gehirns beim Erwachsenen in Abhängigkeit von Alter, Geschlecht, Körpergröße und Gewicht. *Der Pathologe*, 15(3):165–170, 1994.
- Hasher, L., Goldstein, D., and Toppino, T. Frequency and the conference of referential validity. *Journal of Verbal Learning and Verbal Behavior*, 16(1):107–112, 1977.
- He, K., Zhang, X., Ren, S., and Sun, J. Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1026–1034, 2015.
- Hendrycks, D. and Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019.
- Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.
- Hoffer, E., Hubara, I., and Soudry, D. Fix your classifier: the marginal value of training the last weight layer. *arXiv preprint arXiv:1801.04540*, 2018.

- Hornik, K., Stinchcombe, M., and White, H. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989.
- Huang, G.-B., Zhu, Q.-Y., and Siew, C.-K. Extreme learning machine: theory and applications. *Neurocomputing*, 70(1-3):489–501, 2006.
- Hubel, D. H. and Wiesel, T. N. Receptive fields of single neurones in the cat's striate cortex. *The Journal of Physiology*, pp. 574–591, 1959.
- Huber, L. S., Geirhos, R., and Wichmann, F. A. Out-of-distribution robustness: Limited image exposure of a four-year-old is enough to outperform ResNet-50. In *NeurIPS Workshop on Shared Visual Representations in Human & Machine Intelligence*, 2021.
- Hutton, D. The quest for artificial intelligence: A history of ideas and achievements. *Kybernetes*, 2011.
- Ilyas, A., Santurkar, S., Engstrom, L., Tran, B., and Madry, A. Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems*, 2019.
- Jaeger, H. Adaptive nonlinear system identification with echo state networks. In *Advances in Neural Information Processing Systems*, 2002.
- Jo, J. and Bengio, Y. Measuring the tendency of CNNs to learn surface statistical regularities. *arXiv preprint arXiv:1711.11561*, 2017.
- Kaplan, A. *The conduct of inquiry: Methodology for behavioural science*. Chandler Publishing Company, 1964.
- Karimi-Rouzbahani, H., Bagheri, N., and Ebrahimpour, R. Invariant object recognition is a personalized selection of invariant features in humans, not simply explained by hierarchical feed-forward vision models. *Scientific Reports*, 7(1):1–24, 2017.
- Kay, K. N. Principles for models of neural information processing. *NeuroImage*, 180:101–109, 2018.
- Kety, S. S. The general metabolism of the brain in vivo. In *Metabolism of the Nervous System*, pp. 221–237. Elsevier, 1957.
- Khaligh-Razavi, S.-M. and Kriegeskorte, N. Deep supervised, but not unsupervised, models may explain IT cortical representation. *PLoS Computational Biology*, 10(11):e1003915, 2014.
- Kheradpisheh, S. R., Ghodrati, M., Ganjtabesh, M., and Masquelier, T. Deep networks can resemble human feed-forward vision in invariant object recognition. *Scientific Reports*, 6(1):1–24, 2016a.
- Kheradpisheh, S. R., Ghodrati, M., Ganjtabesh, M., and Masquelier, T. Humans and deep networks largely agree on which kinds of variation make object recognition harder. *Frontiers in Computational Neuroscience*, 10:92, 2016b.
- Kienzle, W., Franz, M. O., Schölkopf, B., and Wichmann, F. A. Center-surround patterns emerge as optimal predictors for human saccade targets. *Journal of Vision*, 9(5):7–7, 2009.
- Kietzmann, T. C., McClure, P., and Kriegeskorte, N. Deep neural networks in computational neuroscience. *Oxford Research Encyclopedia of Neuroscience*, 2019a.
- Kietzmann, T. C., Spoerer, C. J., Sörensen, L. K., Cichy, R. M., Hauk, O., and Kriegeskorte, N. Recurrence is required to capture the representational dynamics of the human visual system. *Proceedings of the National Academy of Sciences*, 116(43):21854–21863, 2019b.
- Klayman, J. Varieties of confirmation bias. *Psychology of Learning and Motivation*, 32:385–418, 1995.

- Kornblith, S., Norouzi, M., Lee, H., and Hinton, G. Similarity of neural network representations revisited. In *International Conference on Machine Learning*, pp. 3519–3529. PMLR, 2019.
- Kornblith, S., Lee, H., Chen, T., and Norouzi, M. What’s in a loss function for image classification? *arXiv preprint arXiv:2010.16402*, 2020.
- Kriegeskorte, N. Deep neural networks: a new framework for modeling biological vision and brain information processing. *Annual Review of Vision Science*, 1:417–446, 2015.
- Kriegeskorte, N., Mur, M., and Bandettini, P. A. Representational similarity analysis—connecting the branches of systems neuroscience. *Frontiers in Systems Neuroscience*, 2: 4, 2008.
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2012.
- Kruskal, J. B. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–27, 1964.
- Kubilius, J., Bracci, S., and Op de Beeck, H. P. Deep neural networks as a computational model for human shape sensitivity. *PLoS Computational Biology*, 12(4):e1004896, 2016.
- Kubilius, J., Schrimpf, M., Kar, K., Rajalingham, R., Hong, H., Majaj, N., Issa, E., Bashivan, P., Prescott-Roy, J., Schmidt, K., et al. Brain-like object recognition with high-performing shallow recurrent ANNs. In *Advances in Neural Information Processing Systems*, 2019.
- Kuhn, T. S. *The structure of scientific revolutions*. University of Chicago press, 1962.
- Kümmerer, M., Theis, L., and Bethge, M. Deep Gaze I: Boosting saliency prediction with feature maps trained on ImageNet. In *International Conference on Learning Representations*, pp. 1–12, 2015.
- LeCun, Y., Bengio, Y., and Hinton, G. Deep learning. *Nature*, 521(7553):436, 2015.
- Lehky, S. R. and Sejnowski, T. J. Network model of shape-from-shading: neural function arises from both receptive and projective fields. *Nature*, 333(6172):452–454, 1988.
- Liao, Q. and Poggio, T. Bridging the gaps between residual learning, recurrent neural networks and visual cortex. *arXiv preprint arXiv:1604.03640*, 2016.
- Lighthill, J. Artificial intelligence: A general survey. In *Artificial Intelligence: a paper symposium*, pp. 1–21. Science Research Council London, 1973.
- Lillicrap, T. P. and Kording, K. P. What does it mean to understand a neural network? *arXiv preprint arXiv:1907.06374*, 2019.
- Lindsay, G. W. Convolutional neural networks as a model of the visual system: Past, present, and future. *Journal of Cognitive Neuroscience*, pp. 1–15, 2020.
- Lotter, W., Kreiman, G., and Cox, D. A neural network trained for prediction mimics diverse features of biological neurons and perception. *Nature Machine Intelligence*, 2(4):210–219, 2020.
- Lukoševičius, M. and Jaeger, H. Reservoir computing approaches to recurrent neural network training. *Computer Science Review*, 3(3):127–149, 2009.
- Ma, W. J. and Peters, B. A neural network walks into a lab: towards using deep nets as models for human behavior. *arXiv preprint arXiv:2005.02181*, 2020.
- Maass, W., Natschläger, T., and Markram, H. Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural Computation*, 14(11):2531–2560, 2002.

- Macke, J. H. and Wichmann, F. A. Estimating predictive stimulus features from psychophysical data: The decision image technique applied to human faces. *Journal of Vision*, 10(5):22–22, 2010.
- Marr, D. *Vision: A computational investigation into the human representation and processing of visual information*. W. H. Freeman and Company, San Francisco, 1982.
- Maslow, A. H. *The psychology of science: A Reconnaissance*. New York: Harper & Row, 1966.
- McClelland, J. L. and Rumelhart, D. E. *Parallel distributed processing*, volume 2. MIT press Cambridge, MA, 1986.
- McClelland, J. L., Hill, F., Rudolph, M., Baldridge, J., and Schütze, H. Placing language in an integrated understanding system: Next steps toward human-level performance in neural language models. *Proceedings of the National Academy of Sciences*, 117(42): 25966–25974, 2020.
- McDermott, D., Waldrop, M. M., Chandrasekaran, B., McDermott, J., and Schank, R. The dark ages of AI: a panel discussion at AAAI-84. *AI Magazine*, 6(3):122–122, 1985.
- Meding, K., Janzing, D., Schölkopf, B., and Wichmann, F. A. Perceiving the arrow of time in autoregressive motion. In *Advances in Neural Information Processing Systems*, 2019.
- Meding, K., Buschhoff, L. M. S., Geirhos, R., and Wichmann, F. A. Trivial or impossible—dichotomous data difficulty masks model differences (on ImageNet and beyond). In *International Conference on Learning Representations*, 2022.
- Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., and Galstyan, A. A survey on bias and fairness in machine learning. *arXiv preprint arXiv:1908.09635*, 2019.
- Mehrer, J., Spoerer, C. J., Kriegeskorte, N., and Kietzmann, T. C. Individual differences among deep neural network models. *Nature Communications*, 11(1):1–12, 2020.
- Michaelis, C., Mitzkus, B., Geirhos, R., Rusak, E., Bringmann, O., Ecker, A. S., Bethge, M., and Brendel, W. Benchmarking Robustness in Object Detection: Autonomous Driving when Winter is Coming. In *NeurIPS Workshop on Machine Learning for Autonomous Driving*, 2019.
- Minsky, M. and Papert, S. Perceptrons: an introduction to computational geometry. *The MIT Press, Cambridge, expanded edition*, 19(88):2, 1969.
- Mitchell, M. Why AI is harder than we think. *arXiv preprint arXiv:2104.12871*, 2021.
- Mitchell, T. M. *The need for biases in learning generalizations*. New Jersey: Department of Computer Science, Laboratory for Computer Science Research, Rutgers University, 1980.
- Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.
- Montavon, G., Samek, W., and Müller, K.-R. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, 73:1–15, 2018.
- Moravec, H. *Mind children: The future of robot and human intelligence*. Harvard University Press, 1988.
- Morcos, A. S., Raghu, M., and Bengio, S. Insights on representational similarity in neural networks with canonical correlation. In *Advances in Neural Information Processing Systems*, 2018.
- Muttenthaler, L. and Hebart, M. N. THINGSvision: a Python toolbox for streamlining the extraction of activations from deep neural networks. *bioRxiv*, 2021.



- Navon, D. Forest before trees: The precedence of global features in visual perception. *Cognitive Psychology*, 9(3):353–383, 1977.
- Nguyen, A., Yosinski, J., and Clune, J. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 427–436. IEEE, 2015.
- Noh, H., Hong, S., and Han, B. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1520–1528, 2015.
- Olah, C., Satyanarayan, A., Johnson, I., Carter, S., Schubert, L., Ye, K., and Mordvintsev, A. The building blocks of interpretability. *Distill*, 3(3):e10, 2018.
- Oord, A. v. d., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv:1807.03748*, 2018.
- O’Reilly, R. C., Wyatte, D., Herd, S., Mingus, B., and Jilk, D. J. Recurrent processing during object recognition. *Frontiers in Psychology*, 4:124, 2013.
- Orhan, A. E., Gupta, V. V., and Lake, B. M. Self-supervised learning through the eyes of a child. *arXiv preprint arXiv:2007.16189*, 2020.
- Papert, S. A. The summer vision project, 1966.
- Peissig, J. J. and Tarr, M. J. Visual object recognition: Do we know more now than we did 20 years ago? *Annual Review of Psychology*, 58:75–96, 2007.
- Ponce, C. R., Xiao, W., Schade, P. F., Hartmann, T. S., Kreiman, G., and Livingstone, M. S. Evolving images for visual neurons using a deep generative network reveals coding principles and neuronal preferences. *Cell*, 177(4):999–1009, 2019.
- Raghu, M., Gilmer, J., Yosinski, J., and Sohl-Dickstein, J. SVCCA: singular vector canonical correlation analysis for deep learning dynamics and interpretability. In *Advances in Neural Information Processing Systems*, 2017.
- Rahimi, A. and Recht, B. Weighted sums of random kitchen sinks: replacing minimization with randomization in learning. In *Advances in Neural Information Processing Systems*, 2008.
- Rahimi, A., Recht, B., et al. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, 2007.
- Raichle, M. E. and Gusnard, D. A. Appraising the brain’s energy budget. *Proceedings of the National Academy of Sciences*, 99(16):10237–10239, 2002.
- Rajalingham, R., Issa, E. B., Bashivan, P., Kar, K., Schmidt, K., and DiCarlo, J. J. Large-scale, high-resolution comparison of the core visual object recognition behavior of humans, monkeys, and state-of-the-art deep artificial neural networks. *Journal of Neuroscience*, 38(33):7255–7269, 2018.
- Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D. Y., Bagul, A., Langlotz, C., Shpanskaya, K. S., Lungren, M. P., and Ng, A. Y. CheXNet: Radiologist-level pneumonia detection on chest X-rays with deep learning. *arXiv:1711.05225*, 2017.
- Recht, B., Roelofs, R., Schmidt, L., and Shankar, V. Do ImageNet classifiers generalize to ImageNet? In *International Conference on Machine Learning*, pp. 5389–5400. PMLR, 2019.
- Rendsburg, L., Heidrich, H., and Von Luxburg, U. NetGAN without GAN: from random walks to low-rank approximations. In *International Conference on Machine Learning*, pp. 8073–8082. PMLR, 2020.
- Rosenblatt, F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386, 1958.

- Rosenthal, R. and Fode, K. L. *The problem of experimenter outcome-bias*. Series research in social psychology. Washington, DC: National Institute of Social and Behavioral Science, 1961.
- Rumelhart, D. E., Hinton, G. E., and McClelland, J. L. A general framework for parallel distributed processing. *Parallel distributed processing: Explorations in the microstructure of cognition*, 1(45-76):26, 1986a.
- Rumelhart, D. E., Hinton, G. E., and Williams, R. J. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986b.
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.
- Rust, N. C. and Movshon, J. A. In praise of artifice. *Nature Neuroscience*, 8(12):1647–1650, 2005.
- Saxe, A., Nelli, S., and Summerfield, C. If deep learning is the answer, then what is the question? *arXiv preprint arXiv:2004.07580*, 2020.
- Schönfelder, V. H. and Wichmann, F. A. Identification of stimulus cues in narrow-band tone-in-noise detection using sparse observer models. *The Journal of the Acoustical Society of America*, 134(1):447–463, 2013.
- Schrimpf, M., Kubilius, J., Hong, H., Majaj, N. J., Rajalingham, R., Issa, E. B., Kar, K., Bashivan, P., Prescott-Roy, J., Schmidt, K., et al. Brain-Score: which artificial neural network for object recognition is most brain-like? *BioRxiv*, pp. 407007, 2018.
- Schubert, L., Voss, C., Cammarata, N., Goh, G., and Olah, C. High-low frequency detectors. *Distill*, 6(1), 2021.
- Schönfelder, V. H. and Wichmann, F. A. Identification of stimulus cues in narrow-band tone-in-noise detection using sparse observer models. *The Journal of the Acoustical Society of America*, 134(1):447–463, 2013.
- Sejnowski, T. J. *What are the projective fields of cortical neurons?* Oxford, UK: Oxford University Press, 2006.
- Shen, S., Baevski, A., Morcos, A. S., Keutzer, K., Auli, M., and Kiela, D. Reservoir transformer. *arXiv preprint arXiv:2012.15045*, 2020.
- Shepard, R. N. The analysis of proximities: Multidimensional scaling with an unknown distance function. I. *Psychometrika*, 27(2):125–140, 1962.
- Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.
- Silvermintz, D. *Protagoras*. Bloomsbury Publishing, 2015.
- Smaldino, P. E. and McElreath, R. The natural selection of bad science. *Royal Society Open Science*, 3(9):160384, 2016.
- Smith, C. U. The use and abuse of metaphors in the history of brain science. *Journal of the History of the Neurosciences*, 2(4):283–301, 1993.
- Spoerer, C. J., McClure, P., and Kriegeskorte, N. Recurrent convolutional neural networks: a better model of biological object recognition. *Frontiers in Psychology*, 8:1551, 2017.
- Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. *arXiv:1312.6199*, 2013.
- Taigman, Y., Yang, M., Ranzato, M., and Wolf, L. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1701–1708, 2014.

- Tishby, N. and Zaslavsky, N. Deep learning and the information bottleneck principle. In *IEEE Information Theory Workshop (ITW)*, pp. 1–5. IEEE, 2015.
- Torralba, A. and Efros, A. A. Unbiased look at dataset bias. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.
- van Bergen, R. S. and Kriegeskorte, N. Going in circles is the way forward: the role of recurrence in visual inference. *Current Opinion in Neurobiology*, 65:176–193, 2020.
- Vlasic, B. and Boudette, N. E. Self-driving Tesla was involved in fatal crash, U.S. says, 2016. URL <https://www.nytimes.com/2016/07/01/business/self-driving-tesla-fatal-crash-investigation.html>.
- Walker, E. Y., Sinz, F. H., Cobos, E., Muhammad, T., Froudarakis, E., Fahey, P. G., Ecker, A. S., Reimer, J., Pitkow, X., and Tolias, A. S. Inception loops discover what excites neurons most using deep predictive models. *Nature Neuroscience*, 22(12):2060–2065, 2019.
- Wang, L. P. and Wan, C. R. Comments on "the extreme learning machine". *IEEE Transactions on Neural Networks*, 19(8):1494–1495, 2008.
- Watson, A. B., Barlow, H., and Robson, J. G. What does the eye see best? *Nature*, 302(5907):419–422, 1983.
- Werbos, P. Beyond regression: new tools for prediction and analysis in the behavioral sciences. *Ph. D. dissertation, Harvard University*, 1974.
- Wichmann, F. A. and Jäkel, F. *Methods in Psychophysics*, pp. 1–42. John Wiley & Sons, Inc, 2018.
- Wichmann, F. A., Graf, A. B., Bühlhoff, H. H., Simoncelli, E. P., and Schölkopf, B. Machine learning applied to perception: Decision images for gender classification. In *Advances in Neural Information Processing Systems*, 2005.
- Wichmann, F. A., Drewes, J., Rosas, P., and Gegenfurtner, K. R. Animal detection in natural scenes: Critical features revisited. *Journal of Vision*, 10(4):6–6, 2010.
- Wolpert, D. H. and Macready, W. G. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997.
- Xu, Y. and Vaziri-Pashkam, M. Limits to visual representational correspondence between convolutional neural networks and the human brain. *Nature Communications*, 12(1): 1–16, 2021.
- Yamins, D., Hong, H., Cadieu, C., and DiCarlo, J. Hierarchical modular optimization of convolutional networks achieves representations similar to macaque IT and human ventral stream. In *Advances in Neural Information Processing Systems*, 2013.
- Yamins, D. L. and DiCarlo, J. J. Using goal-driven deep learning models to understand sensory cortex. *Nature Neuroscience*, 19(3):356–365, 2016.
- Yamins, D. L., Hong, H., Cadieu, C. F., Solomon, E. A., Seibert, D., and DiCarlo, J. J. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, 111(23):8619–8624, 2014.
- Yovel, Y., Franz, M. O., Stilz, P., and Schnitzler, H.-U. Plant classification from bat-like echolocation signals. *PLoS Computational Biology*, 4(3):e1000032, 2008.
- Zador, A. M. A critique of pure learning and what artificial neural networks can learn from animal brains. *Nature Communications*, 10(1):1–7, 2019.
- Zhang, X., Zhou, X., Lin, M., and Sun, J. ShuffleNet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6848–6856, 2018.

Zhou, Z. and Firestone, C. Humans can decipher adversarial images. *Nature Communications*, 10(1):1334, 2019.

Zhuang, C., Yan, S., Nayebi, A., Schrimpf, M., Frank, M. C., DiCarlo, J. J., and Yamins, D. L. Unsupervised neural network models of the ventral visual stream. *Proceedings of the National Academy of Sciences*, 118(3), 2021.

Zimmermann, R. S., Borowski, J., Geirhos, R., Bethge, M., Wallis, T. S. A., and Brendel, W. How well do feature visualizations support causal understanding of CNN activations? In *Advances in Neural Information Processing Systems*, 2021.

