Functional Transfer of Generalizing Representations in Biological and Artificial Neural Networks

Dissertation
der Mathematisch-Naturwissenschaftlichen Fakultät
der Eberhard Karls Universität Tübingen
zur Erlangung des Grades eines
Doktors der Naturwissenschaften
(Dr. rer. nat.)

vorgelegt von
Arne Fabian Nix
aus Mönchengladbach

Tübingen
2025

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation: 19.11.2025

Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter:	Prof. Dr. Fabian Sinz
2. Berichterstatter:	Prof. Dr. Matthias Bethge

Ich erkläre, dass ich die zur Promotion eingereichte Arbeit mit dem Titel “Functional Transfer of Generalizing Representations in Biological and Artificial Neural Networks” selbstständig verfasst, nur die angegebenen Quellen und Hilfsmittel benutzt und wörtlich oder inhaltlich übernommene Stellen als solche gekennzeichnet habe. Ich versichere an Eides statt, dass diese Angaben wahr sind und dass ich nichts verschwiegen habe. Mir ist bekannt, dass die falsche Abgabe einer Versicherung an Eides statt mit Freiheitsstrafe bis zu drei Jahren oder mit Geldstrafe bestraft wird.

Ort/Place, Datum/Date

Unterschrift/Signature

Acknowledgements

The completion of this thesis would not have been possible without the guidance, support, and encouragement of many individuals and organizations. I am deeply grateful to those who have contributed their time, expertise, and resources to help me throughout this journey. This section is dedicated to expressing my heartfelt appreciation to all those who played a significant role in shaping this work and supporting me during my academic pursuit.

First and foremost, I would like to express my deepest gratitude to my supervisor, Fabian Sinz. Your guidance and continuous support throughout my entire Ph.D. were invaluable, especially in times of struggle. Thank you for giving me the opportunity to conduct this research and for creating such an incredible and fun environment in your group. Working under your supervision has been an enriching and inspiring experience.

I am immensely thankful to Matthias Bethge, whose advice at crucial points had a significant impact on the shape this thesis ultimately took. Your insights and expertise have been instrumental in guiding my work. I would also like to thank Georg Martius and Robert Bamler, who completed my thesis committee and helped guide me in my research. Your thoughtful feedback and encouragement have been vital throughout this journey. Special thanks go to the IMPRS-IS graduate school and, in particular, to Leila Masri for their unwavering support and for fostering a vibrant and collaborative community of researchers. The sense of belonging and intellectual engagement within this community has been a cornerstone of my Ph.D. experience.

I am profoundly grateful to my friends and colleagues who supported me along the way. Mohammad, the best office mate one could wish for, inspired me to always stay curious. Creating the ML crash course together was one of the highlights of my Ph.D. journey. Suhas, from being my HiWi for a few days to becoming a trusted colleague and dear friend, provided invaluable advice and a willingness to listen. Dominik and Sara, although we only came together late in my Ph.D., made sharing an office a joy, and I will always cherish our time in India. The collaboration with Max contributed greatly to the success of my final project. Konsti brought infectious enthusiasm and made our collaboration memorable by allowing us to each bring our strengths. Shahd and Felix, my master's students, made it a pleasure to witness their growth and work alongside them. I would also like to thank Konstantin, Pavithra, Pawel, Edgar, Christoph, Veronika, Silke, and everyone else at SinzLab. Your camaraderie and support made my time in the lab enjoyable and productive. Special thanks to Mara, Michaela, and the rest of EckerLab for being wonderful colleagues. It was great fun to have you around and share ideas and experiences. Sascha, thank you for always being there with support and for offering great feedback at critical moments.

Finally, I extend my heartfelt thanks to my family: to my sisters, Ilka and Alina, for their unwavering belief in me and their constant encouragement; and to my parents,

Helga and Manfred, for their endless love and support. Your sacrifices and faith in my abilities have been my greatest source of strength.
To all those who have supported me directly or indirectly during this journey, I am forever grateful. This thesis is as much a testament to your contributions as it is to my efforts.

Abstract

In recent years, the field of deep learning has witnessed remarkable progress, driving the integration of these technologies into a wide range of real-world applications. This growing adoption of Deep Neural Networks (DNNs) is underpinned by an increasing confidence in their reliability and ability to generalize effectively across diverse tasks. However, the ability of the models to generalize to previously unseen inputs is poorly understood and frequently lacking, especially in smaller models. Generalization either has to be built into the architecture or it has to be learned from data. The former is limited by what we as developers can imagine, the latter is very costly and also limited by the quality of data available. Therefore it is desirable to consider systems that already have generalizing properties, especially on a representational level, and transfer those to a target system. As doing so would only make sense if the architecture is different between the two, the only option is to perform this transfer based on functional activity, i.e. to use functional transfer methods.

However, although there are numerous such methods proposed in the literature, it is unclear which of those are actually useful for transferring generalization. Thus, we aim with this thesis to understand how generalization generally manifests in learning systems and how it can be transferred specifically to artificial neural networks. We approach both points by considering application areas with functional transfer on two different sources for the generalizing representations.

In the first case, we use the recorded neuronal activity from the visual cortex of a macaque monkey as the teacher and successfully transfer generalization properties to a standard DNN student. This is achieved by training the DNN to jointly predict the monkey's neuronal responses with the actual classification task. We also introduce the attention readout architecture, a novel method for predicting neuron responses from DNN representations. This approach surpasses the current state-of-the-art in predicting responses in macaque area V4. This enables not only a deeper understanding of the visual cortex through more accurate in-silico experiments but also better functional transfer from neuronal activity in the future.

In the second case, the teacher is a DNN that has either learned or built-in properties that allow it to generalize better and thus are desirable to transfer to the student DNN. For this setting, we initially investigate fundamental transfer abilities for clearly defined invariances in a small, controlled environment with theoretical and empirical methods. The results show that established transfer methods cannot reliably transfer even simple invariances. Aiming to close this gap, we propose a novel method we call Orbit, as a method to capture and transfer invariance from teacher to student and demonstrate that it successfully solves this problem in our controlled environment. Building on the insights gained from Orbit and the corresponding analysis, we further propose a general framework with "Hard Augmentations for Robust Distillation

(HARD)" that extends existing functional transfer methods to handle invariances and other generalizing properties. We demonstrate its effectiveness beyond small-scale examples by outperforming state-of-the-art transfer methods on several tasks. Overall, we take a step towards understanding how generalization works and offer much insight into its transfer through functional methods. We reveal for the first time that traditional functional transfer methods are insufficient when it comes to the transfer of generalization and offer three new methods that successfully transfer invariance from artificial neural networks (Orbit) or robustness from neuronal data (Co-Training), as well as one more general method that can be applied generally for any setting (HARD).

Zusammenfassung

In den letzten Jahren wurden auf dem Gebiet des Deep Learning bemerkenswerte Fortschritte erzielt, welche die Integration dieser Technologien in eine Vielzahl von realen Anwendungen vorantreiben. Diese zunehmende Akzeptanz von DNNs wird durch das wachsende Vertrauen in ihre Zuverlässigkeit und ihre Fähigkeit zur effektiven Generalisierung über verschiedene Aufgaben hinweg untermauert. Allerdings ist die Fähigkeit der Modelle zur Generalisierung auf zuvor unbekannte Eingaben schlecht verstanden und häufig unzureichend, insbesondere bei kleineren Modellen. Die Generalisierung muss entweder in die Architektur eingebaut oder aus Daten gelernt werden. Ersteres ist begrenzt durch das, was wir uns als Entwickler vorstellen können, letzteres ist sehr kostspielig und auch durch die Qualität der verfügbaren Daten begrenzt. Daher ist es wünschenswert, Systeme zu betrachten, die bereits generalisierende Eigenschaften haben, insbesondere auf der Ebene von Representationen, und diese auf ein Zielsystem zu übertragen. Da dies nur dann sinnvoll ist, wenn sich die Architektur der beiden Systeme unterscheidet, besteht die einzige Möglichkeit darin, diese Übertragung auf der Grundlage funktionaler Aktivitäten durchzuführen, d. h. funktionale Transfermethoden zu verwenden. Obwohl in der Literatur zahlreiche solcher Methoden vorgeschlagen werden, ist unklar, welche davon tatsächlich für den Transfer der Generalisierung nützlich sind. Daher wollen wir in dieser Arbeit verstehen, wie sich Generalisierung generell in lernenden Systemen manifestiert und wie sie speziell auf künstliche neuronale Netze übertragen werden kann. Wir nähern uns beiden Punkten, indem wir Anwendungsbereiche mit funktionalem Transfer auf zwei verschiedene Quellen für die generalisierenden Repräsentationen betrachten.
Im ersten Fall verwenden wir die aufgezeichnete neuronale Aktivität aus dem visuellen Kortex eines Makaken-Affen als Lehrer und übertragen erfolgreich Generalisierungseigenschaften auf einen Standard-DNN-Schüler. Dies wird erreicht, indem DNN trainiert wird, die neuronalen Antworten des Affen gemeinsam mit der eigentlichen Klassifizierungsaufgabe vorherzusagen. Wir stellen auch die Attention-Readout Architektur vor, eine neuartige Methode zur Vorhersage von Neuronenantworten aus DNN-Repräsentationen. Dieser Ansatz übertrifft den derzeitigen Stand der Technik bei der Vorhersage von Reaktionen im Gehirnbereich V4 von Makaken. Dies ermöglicht nicht nur ein tieferes Verständnis des visuellen Kortex durch genauere In-silico-Experimente, sondern auch eine bessere funktionelle Übertragung der neuronalen Aktivität in der Zukunft.
Im zweiten Fall ist der Lehrer ein DNN, das entweder erlernte oder eingebaute Eigenschaften hat, die es ihm erlauben, besser zu generalisieren und daher wünschenswert sind, sie auf den Schüler DNN zu übertragen. Für dieses Setting untersuchen wir zunächst in einer kleinen, kontrollierten Umgebung mit theoretischen und empirischen Methoden grundlegende Transferfähigkeiten für klar

definierte Invarianzen. Die Ergebnisse zeigen, dass etablierte Transfermethoden selbst einfache Invarianzen nicht zuverlässig übertragen können. Um diese Lücke zu schließen, schlagen wir eine neuartige Methode vor, die wir Orbit nennen, als eine Methode zur Erfassung und Übertragung von Invarianzen vom Lehrer zum Schüler, und zeigen, dass sie dieses Problem in unserer kontrollierten Umgebung erfolgreich löst. Aufbauend auf den mit Orbit gewonnenen Erkenntnissen und der entsprechenden Analyse schlagen wir ein allgemeines Framework mit „Hard Augmentations for Robust Distillation (HARD)“ vor, das bestehende funktionale Transfermethoden erweitert, um Invarianzen und andere verallgemeinernde Eigenschaften zu behandeln. Wir demonstrieren seine Effektivität über kleine Beispiele hinaus, indem wir die modernsten Transfermethoden bei mehreren Aufgaben übertreffen.
Insgesamt machen wir einen Schritt hin zum Verständnis, wie Generalisierung funktioniert, und bieten viele Einblicke in ihre Übertragung durch funktionale Methoden. Wir zeigen zum ersten Mal, dass traditionelle funktionale Transfermethoden unzureichend sind, wenn es um den Transfer von Generalisierung geht, und bieten drei neue Methoden an, die erfolgreich Invarianz aus künstlichen neuronalen Netzen (Orbit) oder Robustheit aus neuronalen Daten (Co-Training) übertragen, sowie eine allgemeinere Methode, die generell für jede Umgebung angewendet werden kann (HARD).

Table of Contents

Introduction 1

Intelligent beings, be they humans or other animals, have the remarkable ability to quickly adapt to novel circumstances and environments. A cow on the beach will immediately be recognized as a cow even if the observer has only ever seen cows in grassy fields. This ability to *generalize* from previously learned concepts to new scenarios is crucial not only for vision but throughout many aspects of intelligence. Despite enormous progress in Artificial Intelligence (AI) through the emergence of deep learning, many AI systems still struggle with this. At the same time, AI becomes more and more omnipresent, with applications ranging from generating a funny image [Ramesh et al., 2021] over recognizing skin-cancer [Naqvi et al., 2023] all the way to controlling nuclear fusion [Degrave et al., 2022]. This also means that having a reliable and thus generalizing system is crucial for the success and safety of those applications. Some modern systems achieve remarkable results in this regard, but this usually comes with incredibly large scales of both training data size and most importantly number of parameters. Therefore, such large-scale Machine Learning (ML) systems are usually similarly hard to use in many real-world scenarios as it would be to use a real brain. This brings us to the question: *Could we somehow capture the generalization abilities from these systems to instill the same into more easily applicable systems?*

Following Sinz et al. [2019], there are three different levels on which this can be achieved, all of which are orthogonal to each other. First, is the implementational level, where the goal is to match or reproduce the biological structures found in the brain. Second, is the computational level, which hopes to shape the representations towards generalization by constraining the learning process with multiple tasks that share features. And finally, the representational level, which we will focus on in this thesis.

The idea for achieving generalization on the representational level is that the representations inside of the intelligent system – the *teacher* –, e.g. the population activity of an area in the macaque visual cortex or the activations from an intermediate layer inside a foundation model, encode their inputs in a generalizing way. That means achieving a representation in the target system – the *student* – which mirrors that of the teacher would ensure generalization for the student as well. Obviously, copying over the parameters from teacher to student would ensure the same thing, but this naive approach is not applicable to most of the interesting applications, where teacher and student are structurally distinct. Thus, a careful approach of *functional transfer* that aligns the teacher and student's representation is needed to ensure generalization in the student down the line.

Many such functional transfer methods exist [Hinton and Dean, 2015, McClure and Kriegeskorte, 2016, Zagoruyko and Komodakis, 2017]. However, it is unclear which of

these, if any, would be appropriate for the transfer of generalization. This thesis aims to clarify this issue by evaluating established methods for transferring generalizing representations and establishing new methods to address shortcomings of the existing methods. We will start with an overview of fundamental concepts (Chapter 2) that are going to be discussed throughout the later chapters. This includes background on representations, generalization, and how these aspects are currently expressed in biological vision as well as computer vision. In addition, this chapter will formally introduce the notion of functional transfer before going into its various forms and applications in the following chapters. The core of this thesis is structured in two strands:
Strand ❶ focuses on the transfer of knowledge from biological representations captured in the form of neuronal recordings to an artificial neural network and the challenges and insights particular to this setting. The first publication we discuss in this strand (Section 3.2) explores a novel co-training approach to incorporate neuronal responses and analyzes the generalization benefits achieved through this approach. The second work (Section 3.3) addresses the challenging problem of aligning higher brain areas with the representations of a Deep Neural Network (DNN) by introducing a novel architecture responsible for the alignment.
Strand ❷ approaches functional transfer on a more fundamental level. We focus on the transfer methods in a simplified framework that restricts teacher representations to originate from a (robust) DNN. The initial work in this strand (Section 4.2) conducts a broad empirical and theoretical analysis of established transfer methods' ability to capture generalization in the form of invariance. The results of this analysis show a severe inability of these methods to transfer generalizing representations. Following this, we offer a solution to this problem by extracting the teacher's invariance in an intermediate step and enforcing it on each layer's representation in the transfer. We continue this line of research in another paper (Section 4.3), which refines the previous method to learn the teacher's generalization abilities as data augmentations, transferring them effectively when combined with other transfer methods.
The thesis concludes in Chapter 5 with a discussion of the presented works. We connect the two strands and elaborate on their impact on the fields of machine learning and neuroscience. Finally, we summarize the challenges and questions that remain open after this thesis, to be addressed in the future.

1.1 List of Publications and Contributions

This section presents my publications and their associated contributions. For projects detailed in the thesis chapters, I provide specific information about my role and those of my co-authors. I also include publications from additional projects I contributed to during my PhD.

1.1.1 Publications Included in this Thesis

The following publications represent the core research of this thesis. For each, I detail the contributions using a modified version of the Contributor Roles Taxonomy (CRediT) [Allen et al., 2014, 2019, Holcombe, 2019]. These modifications, inspired by

[Bashiri, 2024], better reflect the nature of methodology-focused papers that comprise this work.

- Shahd Safarani, Arne Nix, Konstantin Willeke, Santiago Cadena, Kelli Restivo, George Denfield, Andreas Tolias, and Fabian Sinz. Towards robust vision by multi-task learning on monkey visual cortex. *Advances in Neural Information Processing Systems*, 34:739–751, 2021.
  Initial idea: FS, SS, AN, KW
  Methodology and theoretical development: SS, AN, KW, FS
  Software: SS, AN, KW
  Method validation and experiments: SS, AN, KW
  Data collection and preprocessing: SC, KR, GD
  Figures and visualization: SS, AN
  Writing (original draft): SS, AN, KW, FS
  Writing (review and editing): all authors
  Funding acquisition: FS and AT

- Paweł A. Pierzchlewicz, Konstantin Friedrich Willeke, Arne Nix, Pavithra Elumalai, Kelli Restivo, Tori Shinn, Cate Nealley, Gabrielle Rodriguez, Saumil Patel, Katrin Franke, Andreas S. Tolias, and Fabian H. Sinz. Energy guided diffusion for generating neurally exciting images. *Advances in Neural Information Processing Systems*, 36, 2024.
  Initial idea: (diffusion) FS, KW, PP **(attention readout)** FS, KW, AN
  Methodology and theoretical development: PP, KW, AN, FS
  Software: PP, KW, AN
  Method validation and experiments: diffusion method mainly PP with help from KW, Attention readout mainly KW & AN with help from PP, PE
  Data collection and preprocessing: KR, TS, CN, GR, SP
  Figures and visualization: PP, AN, PE, KW
  Writing (original draft): PP, AN, PE, KW
  Writing (review and editing): all authors
  Funding acquisition: FS, AT

- Arne Nix, Suhas Shrinivasan, Edgar Y Walker, and Fabian Sinz. Can Functional Transfer Methods Capture Simple Inductive Biases? In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 10703–10717. PMLR, 2022a.
  Initial idea: AN, FS
  Methodology and theoretical development: AN, FS, EW, SS
  Software: AN
  Method validation and experiments: AN
  Data collection and preprocessing: publicly available data was used
  Figures and visualization: AN
  Writing (original draft): AN, SS, FS
  Writing (review and editing): all authors

Funding acquisition: FS

- Arne F Nix, Max F Burg, and Fabian H Sinz. Hard: Hard augmentations for robust distillation. *arXiv preprint arXiv:2305.14890*, 2023.
  Initial idea: AN, FS
  Methodology and theoretical development: AN, FS, MB
  Software: AN
  Method validation and experiments: AN
  Data collection and preprocessing: publicly available data was used
  Figures and visualization: AN
  Writing (original draft): AN, MB, FS
  Writing (review and editing): all authors
  Funding acquisition: FS

1.1.2 Other Publications

The following publications are a result of a project I contributed to during the PhD, but lie outside of the scope of this thesis. For these publications, I will only explain my individual contribution.

- Konstantin F Willeke, Kelli Restivo, Katrin Franke, Arne F Nix, Santiago A Cadena, Tori Shinn, Cate Nealley, Gabby Rodriguez, Saumil Patel, Alexander S Ecker, Fabian H Sinz, and Andreas S Tolias. Deep learning-driven characterization of single cell tuning in primate visual area V4 unveils topological organization. May 2023.
  Contributions: AN contributed to discussions about the methodology, model development, as well as writing (review and editing).

- Arne Nix, Max F Burg, and Fabian H Sinz. Leading by example: Guiding knowledge transfer with adversarial data augmentation. In *NeurIPS 2022 Workshop on Synthetic Data for Empowering ML Research*, 2022b.
  Contributions: This work is an earlier version of [Nix et al., 2023], presented at NeurIPS workshop. Attribution is identical to this publication.

Background 2

This chapter introduces the key terminology, notation, and related work that will be relevant for understanding the work presented in subsequent chapters. While reading this chapter is not required to understand the remainder of this thesis, we believe understanding and connecting the individual works will be easier with the foundations we present here.

2.1 What are Generalizing Representations?

2.1.1 Learning Functions and Neural Networks

In machine learning, we are first and foremost interested in functions mapping a given input from domain $\mathcal{X}$ to an output in domain $\mathcal{Y}$.

$$f : \mathcal{X} \to \mathcal{Y} \tag{2.1}$$

The function may be a real-world system like the brain which we can only observe and collect noisy data from. In that case, the input could be a stimulus that is presented to sensory neurons and the output could be neuronal activity patterns or behavior. Alternatively, the function may be an artificial neural network that we define as a composition of L simple layer functions:

$$f(\boldsymbol{x}; \boldsymbol{\theta}) = (f^{(L)} \circ ... \circ f^{(1)})(\boldsymbol{x}) \tag{2.2}$$

The neural network is parameterized by a set of parameters $\boldsymbol{\theta} = \{\boldsymbol{\theta}^{(L)}, ..., \boldsymbol{\theta}^{(1)}\}$, which we will omit in most cases to simplify notation. A single layer at level l is commonly a non-linear function with parameters $\boldsymbol{\theta}^{(l)}$, and is defined as:

$$f^{(l)} : \mathcal{X}^{(l-1)} \to \mathcal{X}^{(l)} \tag{2.3}$$

Each layer's output domain $\mathcal{X}^{(l)}$ matches the subsequent layer's input domain – denoted *feature space* – and we set $\mathcal{X}^{(0)} = \mathcal{X}$ and $\mathcal{X}^{(L)} = \mathcal{Y}$, so the composition of all layers forms the neural network function. The entire network function is oftentimes optimized to minimize an objective or loss function $\mathcal{L}$ that is a scalar score to given parameters $\boldsymbol{\theta}$ and data $\boldsymbol{x} \in \mathcal{X}$ in unsupervised learning or $(\boldsymbol{x}, y) \in (\mathcal{X} \times \mathcal{Y})$ in supervised learning.

2.1.2 Representations

As we have seen above, the functions we are considering throughout this work can take a variety of forms and can be arbitrarily complex, to the point that we can no

longer understand its inner workings. Thus, in order to analyze and relate different functions, we employ a unifying view by regarding all these functions and systems through *representations*. The term "representation" has a multitude of different definitions, which can vary across fields, or even within the same field [Baker et al., 2022]. The research community in machine learning consistently highlights the importance of representations [Bengio et al., 2013, Goodfellow et al., 2016], but fails to give a formal definition for the term. Thus, we formulate our own definition here, influenced by the definition in representation theory [Teleman, 2005], consistent with the definitions in neuroscience [Baker et al., 2022] and usage in machine learning [Bengio et al., 2013, Goodfellow et al., 2016]. We define a representation as a tensor that describes elements from the real world like objects, words, etc. This tensor is the output of a transformation on the input vector space we chose for the element. Thus, a representation would, for example, be the output of a function on the pixel values of an object's image [Krizhevsky et al., 2012] or on a one-hot encoding of a word [Mikolov, 2013].
Often in machine learning the representations are studied and used because of the element they represent. For example, the representation of a word is interesting by itself as it can be compared in the representational space to other words or used in a downstream application such as translation. But a representation can also be studied to understand the function that produced the representation. In that case, it is often not sufficient to look at the representation of a single element, but of an entire set of elements. How the function acts upon these elements in relation to each other can characterize the function. So for the purpose of this thesis, we denote the representation of a DNN at a specific layer l as the output $f^{(:l)}(\boldsymbol{X}) = (f^{(l)} \circ ... \circ f^{(1)})(\boldsymbol{X})$ the network computes for a given set of inputs $\boldsymbol{X}$ after that layer. Similarly, we define the neuronal representation of a given brain area as the measured response (spike count measured through electrophysiology) of all recorded neurons when presented with a specific set of stimuli.
The degree to which a certain representation describes a specific function is determined by its *data diet* [Konkle and Alvarez, 2022], i.e. by the data points that are used to generate the representation. We hypothesize that for more complex functions, a larger variety in the data diet is necessary to fully capture the function's properties. This idea will become crucial when we focus on generalizing representations (Section 2.1.4) and their transfer (Section 2.3).

2.1.3 Generalization

The most common goal in the discipline of machine learning is to learn a function that has the ability to perform well on previously unseen data. This particular aim describes the process of *generalization*, which is generally measured by splitting the available data into *training* data that is used to adapt the system and its parameters and perform the actual learning on and *validation* or *test* data to evaluate how well the adapted system generalizes. The evaluation data will often be drawn from the same distribution as the training data, in which case we speak of *in-distribution (ID)* evaluation. The more challenging setting would be where the evaluation data does not originate from the same distribution as the training data, which describes

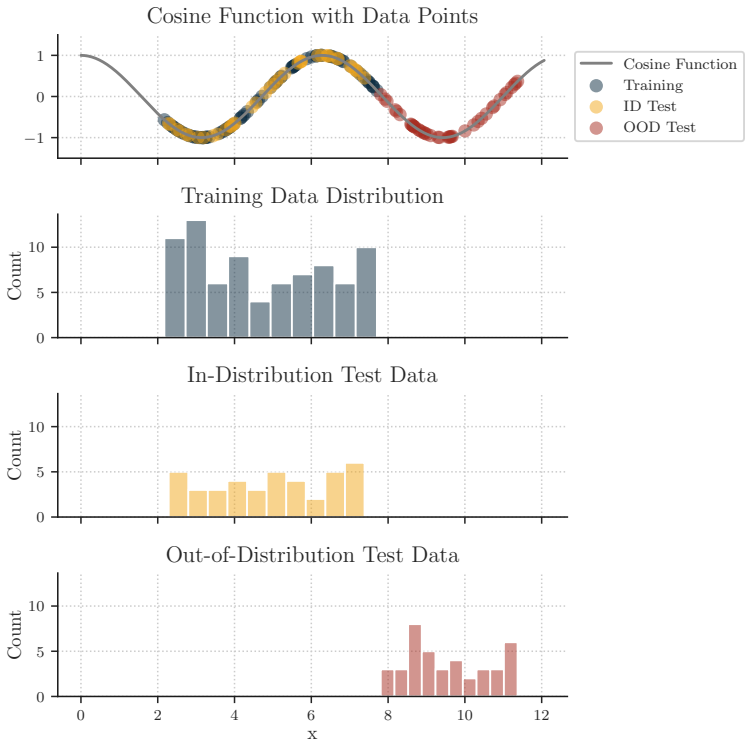


Figure 2.1: An illustrative example of the data distributions for training, ID and OOD evaluation. Here the ID test points evaluate interpolation and only OOD test points are located so that they test the periodicity of the function. Thus, a model would need to capture the periodic properties of the function to extrapolate well to the OOD setting.

out-of-distribution (OOD) evaluation. In the example from Chapter 1, an ID sample would be a new cow on a different patch of grass, and an o.o.d. sample would be a cow on the beach. The ID setting often makes generalization easier, since the system is essentially performing *interpolation* between the examples it observed in the training data in order to make a prediction. The OOD setting on the other hand often requires *extrapolation* beyond the given training data, or interpolation of a larger extend (see Figure 2.1).

As a result, generalization remains one of the key challenges in machine learning. However, it is important to note that most generalization errors are "neither a failure to learn nor a failure to generalize at all, but instead, a failure to generalize in the intended direction." [Geirhos et al., 2020a] The intended direction could be arbitrarily defined but is generally defined as the direction that is deemed "correct" by humans, e.g. the ground-truth labels provided by annotators. Learning systems usually rely on certain features that they find in the inputs to generate an output (e.g. predicting the class of object present in an image). In order to achieve OOD model generalization in the intended direction, the features that the system relies on for predictions need to remain consistent after the distribution shift. This property of a system to rely on features that remain unchanged under distribution shift is the key to a system that is *robust* and generalizes well to OOD settings.

Although there are many different forms of robustness (e.g. robustness to label corruption, gradient noise, ...), this is what we mean with *robustness* in this thesis: *The ability of a learning system to generalize to shifts in the data distribution.*[1] This definition of robustness encompasses a variety of different scenarios that are each a research area of their own. There is robustness to domain shifts, which means that the data distribution is shifted to a new domain between training and evaluation. An area where especially neural network systems struggle with generalization is provided by (usually small and imperceptible) changes in the input that are optimized specifically to change the system's behavior. These are so-called *adversarial* examples, which can be found for many learning systems. Finally, the type of robustness we will be focusing on as a proxy for general generalization ability is the robustness to *common corruptions*, which describes changes in the input that transform an ID input into an OOD sample that could have been the result of a naturally occurring distribution shift. These input changes should therefore reflect corruptions caused in the data acquisition and prepossessing phase, as well as the environment itself [Hendrycks and Dietterich, 2019]. It is worth noting at this point, that these different forms of robustness – although all relevant aspects of generalization – may not be correlated in learning systems in practice [Hendrycks et al., 2022]. That is, achieving one aspect of robustness does not necessarily make the system to other aspects mentioned above. In general, it can be said that robustness is a very important aspect of machine learning, especially when deploying a system in the real world, where it would most certainly encounter unexpected scenarios. Additionally, the more responsibility is given to machine learning models, the more robustness becomes a safety concern [Michaelis et al., 2019] and trust in the system and its outputs grows more and more essential [Degrave et al., 2022, Naqvi et al., 2023]. Finally, it is also a matter of security if a system is vulnerable to malicious attacks that could exploit a lack of robustness.

2.1.4 Generalizing Representations

All this leads to the question of how generalization can be achieved for machine learning systems. The *"no-free-lunch theorem"* [Wolpert and Macready, 1997] essentially states that no single algorithm is universally better than all others across all possible problems. A consequence of this is that generalization to o.o.d. examples need a trade-off to be made. Generalization in "unimportant" directions is traded for generalization in the "important" directions. This is typically achieved by introducing prior assumptions—referred to as inductive biases [Hüllermeier et al., 2013]—on the space of learnable functions [Mitchell, 1980]. These biases guide the learning algorithm to select, based on the training data, a solution that generalizes effectively in the desired direction. This generalization will generally also carry over to representation associated with the learned function, which effectively means that the representation changes only minimally, or in a semantically meaningful way if the input changes along the "important" direction. For example, a generalizing object representation should not change when the background behind the object is changing, if it is to be useful for object recognition tasks. Given their importance and direct connection to the generalization abilities of a function, we focus on the

[1] We will be using the term robustness and OOD generalization interchangeably.

generalizing representations and their transfer as a means of instilling robustness into models that lack inductive biases and would otherwise struggle with generalization.

2.2 A Brief Overview of Generalization in Vision

2.2.1 Generalization in Biological Vision

The human brain not only serves as an inspiration for the development of DNNs, but also as a frame of reference when evaluating machine learning models. In supervised tasks, the ground truth labels are usually collected from human judgment, since we want our model to emulate human behavior in most cases. There are some instances, however, where this is not desirable, especially when human performance is not sufficient for the task at hand. These are for example safety-critical applications where we are aiming for super-human performance instead.
Nevertheless, we are looking at the human and by extension the animal brain to understand generalization, as those are the only examples of general intelligence we know. In this spirit, adversarial examples have often been presented as an example of superior generalization by humans over deep learning models [Szegedy et al., 2014, Goodfellow et al., 2014]. However, more recent work has shown that human behavior and primate neuronal responses are also susceptible to adversarial perturbations in images if the image is presented only for a short time and the perturbation is optimized appropriately [Elsayed et al., 2018, Guo et al., 2022, Veerabadran et al., 2023, Gaziv et al., 2024].
A similar trend is showing for more general corruptions of images, where early work [Geirhos et al., 2018] showed a gap in object classification performance between DNNs and humans when presented with images distorted by common corruptions. More recent results indicate that this performance gap is shrinking with more modern models [Geirhos et al., 2021]. Despite this, the error patterns of humans and DNNs still differ substantially [Eckstein et al., 2017, Rajalingham et al., 2018, Geirhos et al., 2020b, 2021, Malik et al., 2022], and only get more consistent when the model sees one to three orders of magnitude more data [Geirhos et al., 2021]. This may be related to the finding that humans are affected by contextual biases, but much less affected than DNNs when the context is missing [Biederman et al., 1982, Biederman, 1981, Oliva and Torralba, 2007, Castelhano and Heaven, 2011]. Additionally, humans are able to continually learn throughout their lifetime, use prior knowledge, and generalize it to different contexts as well as learn from a limited number of examples [Tenenbaum et al., 2011].
The key question that inspired much of this thesis is: *What makes the brain behave so differently to DNNs and in many cases much more robustly?* While the question is far from fully answered, certain characteristics are thought to play a partial role in explaining the robustness of the visual system. An increased reliance on shape over texture information in humans is a known difference to DNNs [Geirhos et al., 2019], and is attributed to be responsible for human's superior robustness. A related discrepancy is that humans are sensitive to a narrower range of frequencies compared to DNNs [Subramanian et al., 2024]. This insight helps neural network robustness when applied to DNNs [Chen et al., 2021, Li et al., 2023, Bu et al., 2023]. Another

known difference between the two systems is their architecture, especially the fact that the human visual system heavily relies on recurrent connections [Olshausen, 2013, Kietzmann et al., 2019], whereas most modern DNN architectures are exclusively feed-forward designs. Preventing humans from using their recurrent pathways significantly reduces their robustness [Elsayed et al., 2018], indicating their importance for generalization.

Although all these factors likely play a role, the only definitive conclusion is that our understanding of the brain's generalization abilities remains limited. Thus, the focus of this work is not to mimic the brain's architecture or design but to treat the brain as a black-box function for which we can only access its representations and to use those representations to improve existing machine learning models.

2.2.2 Generalization in Computer Vision

Robustness and generalization are crucial for neural networks to perform reliably in real-world applications, making them a significant focus in computer vision research [Drenkow et al., 2021]. Inductive biases (see Section 2.1.4) are foundational for generalization. Early convolutional neural networks (CNNs) [Fukushima, 1980] incorporated translation invariance, which remains a standard in computer vision [LeCun et al., 1998, Krizhevsky et al., 2012]. Similarly, L2-regularization imposes a Gaussian prior on model weights [Tikhonov, 1963]. However, as models grow larger and more complex, designing effective inductive biases becomes increasingly challenging, necessitating alternative strategies.

Data augmentation is a widely used technique to improve generalization by diversifying the training distribution. By modifying inputs, while preserving their labels, models are trained to ignore irrelevant variations and focus on robust features. Basic augmentations, like flips and crops, are standard in preprocessing pipelines [Lee et al., 2015]. More advanced methods, such as AutoAugment [Cubuk et al., 2018], MixUp [Zhang et al., 2017], and CutMix [Yun et al., 2019], apply augmentations guided by policies optimized for generalization. These policies determine the type, combination, and magnitude of augmentations, often requiring computationally expensive optimization [Cubuk et al., 2018, Ho et al., 2019, Tian et al., 2020, Bekor et al., 2024]. Recent methods like TrivialAugment [Müller and Hutter, 2021] bypass optimization entirely while achieving comparable results. For robustness specifically against OOD scenarios, notable augmentation techniques include AugMix [Hendrycks et al., 2020], DeepAugment [Hendrycks et al., 2021a], and PixMix [Hendrycks et al., 2022], which combine multiple augmentations to improve resilience to common corruptions, adversarial attacks, and model calibration. These methods enhance robustness by diversifying the training distribution and encouraging models to generalize across varying conditions.

Building on the idea of exposing models to challenging inputs, adversarial training [Szegedy et al., 2014, Goodfellow et al., 2014, Madry et al., 2018] directly targets perturbations crafted to exploit model vulnerabilities. Variants like domain adversarial training [Ganin et al., 2016] and adversarially optimized noise [Rusak et al., 2020] extend this approach to address domain shifts and a broader range of

Table 2.1: Distribution Shifts and Dataset Categories used to measure generalization performance in different scenarios. Adapted from [Liu et al., 2024]

Robustness	Distribution Shift	Source	Datasets
Natural	ID	Real-world	ImageNet-Val [Deng et al., 2009], ImageNet-V2 [Recht et al., 2019], ImageNet-Real [Beyer et al., 2020]
	OOD	Real-world	ImageNet-A [Hendrycks et al., 2021b], ImageNet-R [Hendrycks et al., 2021a]
		Synthesized	ImageNet-C [Hendrycks and Dietterich, 2019], Stylized-ImageNet [Geirhos et al., 2019], ImageNet-Sketch [Wang et al., 2019]
Adversarial	Adversarial	Adversary	White-box attacks, Transfer-based attacks

image corruptions, ensuring the model's robustness to both subtle and significant variations in data distribution.

Despite the effectiveness of data augmentation and adversarial training, especially in data-scarce scenarios [Islam et al., 2024], recent advances show that models without specialized augmentation can outperform these methods [Mahajan et al., 2018, Orhan, 2019, Radford et al., 2021, Geirhos et al., 2021, He et al., 2022, Oquab et al., 2023, Orhan, 2023]. This is largely attributed to three factors: (1) increased model parameters, (2) larger training datasets, and (3) new self-supervised training methods that, combined with extensive data, mitigate overfitting and fully leverage model capacity. Self-supervised approaches often produce robust representations by learning invariance to transformations like cropping [Oquab et al., 2023] or masking [He et al., 2022]. However, results are mixed; for example, SimCLR [Chen et al., 2020] shows strong generalization to certain corruptions, while other methods perform similarly to supervised approaches [Geirhos et al., 2020c]. While scaling models and data appear promising, systematic biases in large datasets persist [Geirhos et al., 2020a]. Something that is important to remember for the future and a reason why robustness is a key factor in *alignment* of AI [Ji et al., 2023]. We will discuss this topic in greater detail in Section 5.5.

2.2.3 Analyzing and Evaluating Generalization

To develop generalizing models, we need to be able to validate and determine whether a model is robust. In principle, we can quantify robustness by adapting the standard evaluation framework, by changing the data distribution to measure model performance on o.o.d. data. For image classification, this would, for example, imply computing classification accuracy on a new test set of o.o.d. images. In theory, there

are often infinitely many data distributions that are distinct from the original distribution and could be used for sampling the evaluation data. However, in practice, it is hard to find distributions that are challenging for established models and interesting generalization directions at the same time. Thus, there is a plethora of different test sets proposed for many tasks.

For image classification and models that were trained or fine-tuned on the ImageNet [Deng et al., 2009] training set, the variety of datasets for evaluating generalization are summarized in table 2.1. The listed datasets cover a large range of challenging data distributions, and while we will focus on the common corruptions of ImageNet-C [Hendrycks and Dietterich, 2019], we will eventually use all the test sets that cover "natural" robustness.

After quantifying the generalization abilities and identifying a robust model, the next step is to analyze the model to uncover the source of its robustness and ideally enhance it further. Analysis of DNN models is a challenging endeavor, and is mostly done by qualitatively analyzing the neural network features through visualization [Olah et al., 2017]. Along this line, multiple works investigate the origin of robustness in DNNs through feature visualization [Engstrom et al., 2019, Feather et al., 2023, Itazuri et al., 2019, Safarani et al., 2021]. Another form of analysis attempts to quantify the type of features a model is sensitive to by analyzing their frequency spectrum [Li et al., 2023, Liu et al., 2024, Yin et al., 2019]. The studies found that robust networks have a stronger bias towards low-frequency image components than their less robust counterpart.

2.3 Functional Transfer of Generalization

Transferring knowledge from a source to a target is a crucial aspect of machine learning, as it allows us to build on previous learnings and avoid redundant efforts in acquiring the same knowledge repeatedly. The most commonly associated way of doing this is *transfer learning* [Pratt, 1992], which describes the concept of training a *teacher* model on one dataset and transferring its knowledge to a *student* model that is intended to perform well on a separate task or domain. The transfer in this case commonly happens by copying all or a subset of the weights from teacher to student and finetuning the student on the new task. The training of the teacher model is often combined with large-scale pre-training in either supervised [Tan et al., 2018] or unsupervised fashion [He et al., 2019]. This approach of transfer learning has been successful for a long time across many machine learning disciplines, most notably computer vision [Donahue et al., 2014] and natural language processing [Devlin, 2018, Brown et al., 2020].

Despite this success, there are many scenarios where student and teacher don't share the same architecture, and copying the weights is not an option [McClure and Kriegeskorte, 2016, Beyer et al., 2021]. In these cases, *functional transfer* methods are used to capture the teacher's knowledge and instill it into the student. Formally, this means evaluating the teacher function f_t on a inputs x and training the student f_s to elicit similar representations, measured with some distance function

$\mathcal{D} : \mathcal{X}_\mathrm{t} \times \mathcal{X}_\mathrm{s} \to \mathbb{R}^+$:

$$\min_{\boldsymbol{\theta}_\mathrm{s}} \mathcal{D}(f_\mathrm{t}(\boldsymbol{x}), f_\mathrm{s}(\boldsymbol{x})) \tag{2.4}$$

The distance is used to optimize the student's parameters $\boldsymbol{\theta}_\mathrm{s}$.
This way of training the student promotes an alignment between the two representations [Sucholutsky et al., 2023] through the minimization of their *representational distance*. This brings with it four main aspects that vary across applications and need to fit together for a successful transfer:

1. **Which systems are used as teacher and student?** For the student f_s, which will be optimized to align with the teacher, the common choice is DNNs. The teacher f_t on the other hand can differ. For cognitive science, the teacher could be a human subject whose representations are probed through psychophysics experiments [Muttenthaler et al., 2024]. For Neuroscience, neural representations are recorded from human or animal brains through techniques like functional Magnetic Resonance Imaging (fMRI), electrophysiology, or two-photon microscopy [Li et al., 2019]. In ML, the teacher will be a neural network as well that can differ in architecture, size, or training data [Hinton and Dean, 2015].

2. **What representation is used for the transfer?** This means selecting the right brain area and neurons that are recorded or selecting the layer whose activation is used as the representation. This selection of a fitting representation can determine the scope of possible knowledge that can be transferred and can decide the outcome of an experiment early on.

3. **What data is used to generate the representations?** Selecting the right data is crucial for generating meaningful representations because the features extracted from data depend heavily on its quality and relevance. Hubel and Wiesel's [Hubel and Wiesel, 1962, Akler, 2022] groundbreaking discovery exemplifies this—when they accidentally presented a line at a specific angle to a cat's visual cortex, they uncovered that certain neurons are tuned to specific visual stimuli. This serendipitous finding highlights how the right input can unlock hidden patterns and insights in complex systems. Thus, the task of finding the right data diet to generate a representation will become central in this work.

4. **How do we transfer the representations?** This is mostly concerning the representation distance function $\mathcal{D}$. The choices we have available for this are highly dependent on the previous points and the representations resulting from this. For example, if we end up with representations that match in dimensionality and assume a direct relationship between the representations, then the best we can do is to directly minimize the distance between them. A prominent example of this would be standard knowledge distillation [Hinton and Dean, 2015]. If the matching is not as straightforward, other methods need to be employed to make a transfer possible. This can mean adding additional projections or pooling to align them [Zagoruyko and Komodakis, 2017, Lurz

et al., 2021], or it can mean that we need to compare representations along a different axis [McClure and Kriegeskorte, 2016].

Throughout this thesis, we will touch on all of these points and explore them in detail in the context of transferring generalizing representations. These questions form the foundation of our research and are essential building blocks for addressing our main objectives. Specifically, the primary goals of this research are to improve generalization through functional transfer by identifying and leveraging generalizing representations, enhance existing methods for functional transfer to make them more robust and efficient, deepen our understanding of the fundamental principles that govern generalization in machine learning, and gain deeper insights into the mechanisms and conditions under which functional transfer succeeds or fails. By addressing these goals, this work aims to provide both theoretical advancements and practical methodologies, advancing our understanding of generalization and functional transfer and ultimately pushing the boundaries of what is achievable in representation learning.

Functional Transfer of Neural Representations

3

As discussed in Chapter 2, the brain remains the only truly generalizing system we know. Unfortunately, our understanding of what enables the brain to generalize is still limited. Historically, neuroscience has inspired many advancements in artificial intelligence (see Section 2.2), with most of the transfer between neuroscience and machine learning occurring at the implementational level [Marr, 1982, Hassabis et al., 2017]. However, we currently lack the detailed understanding of brain structure needed to transfer functional generalization properties from biological systems to artificial ones [Sinz et al., 2019]. It has been argued that to truly understand the brain's inner workings and abilities, one must focus on its functional activity, especially at the level of entire neuron populations [Ebitz and Hayden, 2021, Pillow and Aoi, 2017]. Techniques such as fMRI [Gore et al., 2003], electrophysiology [Hodgkin and Huxley, 1952, Jun et al., 2017], and two-photon microscopy [Helmchen, 2009, Sofroniew et al., 2016] enable large-scale recordings of brain activity at various scales and granularities. These methods allow researchers to capture the brain's internal representations, facilitating their transfer to DNN models. In this chapter, we discuss two works that attempt this transfer of neural representations into an artificial neural network.

The first one (Section 3.2) imbues the representation from real neurons recorded in the visual cortex of rhesus macaque monkeys into an image classification model to enhance the robustness of the DNN to common corruptions. To align the internal representations of a DNN with those of a biological neural network, it is crucial to mitigate the misalignment with a small parameterized function commonly referred to as *readout*. A readout is a learned transformation that allows to transform the hidden state of a DNN into predicted spike counts of real neurons.

The second work we discuss here (Section 3.3) introduces a novel readout method that aims to improve this transformation such that it becomes more flexible and works better for brain areas higher in the brain.

3.1 Background: Neural System Identification

DNNs have been widely established as the state-of-the-art approach to learn encoding models of neural representations [Cadieu et al., 2014, Yamins et al., 2014, Cadena et al., 2019, Sinz et al., 2018, Zhang et al., 2019, McIntosh et al., 2016, Klindt et al., 2017, Kindel et al., 2019, Ecker et al., 2018, Cowley and Pillow, 2020, Burg et al., 2021, Batty et al., 2022, Bashiri et al., 2021, Antolík et al., 2016]. The model for this is typically composed of two components: a *core* and a *readout*. The core is simply a non-linear

feature extractor, for image stimuli, usually a Convolutional Neural Network (CNN), shared across all neurons in the recorded population and optimized either in *task-driven* or *data-driven* way. A task-driven core is pre-trained on a different task like object recognition [Guclu and van Gerven, 2015, Yamins et al., 2014, Yamins and DiCarlo, 2016, Cadena et al., 2019, 2022], whereas a data-driven core would be trained end-to-end on the neuronal response data and corresponding stimuli only. The second component, the readout, serves to transform the output features of the core to individual neuron responses, essentially aligning the core and neuronal representations. This transformation exists in many forms, but it usually employs some form of linear transformation with a rectifier non-linearity [Nair and Hinton, 2010]. The main difference across the readout variants is the nature of the restriction that is put on the linear transformation to reduce the number of parameters [Klindt et al., 2017, Cadena et al., 2019, Yamins et al., 2014, Lurz et al., 2021, Sinz et al., 2018, Guclu and van Gerven, 2015].
One example architecture for the readout is the *Gaussian readout* [Lurz et al., 2021], which first draws spatial coordinates for each neuron n from a bivariate Gaussian distribution $\mathcal{N}(\mu_n, \Sigma_n)$ with parameters μ_n, Σ_n. The distribution is learned through the reparametrization trick [Kingma and Welling, 2014] and has parameters separate for each neuron. The readout uses the inferred coordinates to select a feature vector via bilinear interpolation [Sinz et al., 2018] from that position in the core's output. Finally, a linear projection and rectifier function are applied to obtain the predicted firing rate conditioned on the input for each neuron.

3.2 Neural Co-Training with Monkey V1

This section is based on the following publication:

> Shahd Safarani, Arne Nix, Konstantin Willeke, Santiago Cadena, Kelli Restivo, George Denfield, Andreas Tolias, and Fabian Sinz. Towards robust vision by multi-task learning on monkey visual cortex. *Advances in Neural Information Processing Systems*, 34:739–751, 2021

Key takeaways from this section:

- We propose neural co-training as a novel approach to transfer knowledge from functional recordings to any task-performing DNN.
- Experiments with recordings of monkey V1 demonstrate a positive effect on generalization, increasing the model's robustness to image distortions despite training exclusively on undistorted images.
- Careful analysis reveals that the feature sensitivity of the monkey V1 co-trained model aligns with salient image features, offering evidence in line with existing theories about the role of V1 [Zhaoping and Li, 2014].

3.2.1 Motivation

The goal of this work is to transfer inductive biases from the brain to DNNs by transferring the representations that were recorded as responses of biological neurons to visual stimuli. It has previously been shown that such a transfer is possible and helpful for CNNs with neural recordings from humans [Fong et al., 2018], mice [Li et al., 2019], or monkeys [Federer et al., 2020]. Enforcing similarity to the brain representations especially benefited the network's generalization abilities to OOD inputs for object recognition. Concurrently, DNNs have established unprecedented benchmarks in modeling brain activity across visual cortical regions [Yamins and DiCarlo, 2016], emerging as the leading approach for predicting neural responses in area V1 [Cadena et al., 2019].
We aim to use these methods for encoding neural responses to align the internal representations of an image classification DNN with the neural representations. This method is based on the idea of Multi-task Learning (MTL) [Caruana, 1993] with image classification and prediction of neural responses and was proposed as the *neural co-training hypothesis* by Sinz et al. [2019], but remained previously untested to the best of our knowledge. We hypothesize that MTL with neural data enriches the shared representation by incorporating functional inductive biases, consequently enhancing the network's ability to generalize across OOD images and improve overall robustness. This approach offers a novel avenue towards functional transfer that only requires a small overhead of additional parameters (in the readout) to align the representations while being fully flexible with regard to representation format and task.

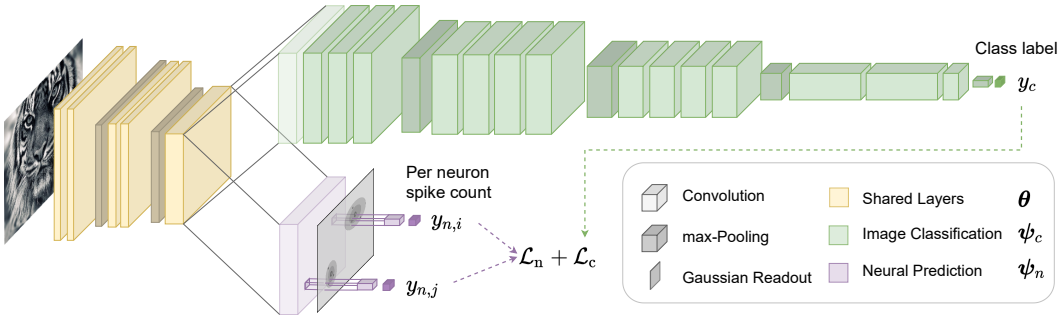


Figure 3.1: Overview of the co-training method. Showing the VGG-19 [Simonyan and Zisserman, 2015] architecture with upper layers split for MTL on image classification (green) and neural prediction (purple) respectively. This method allows us to learn representations that are shared (yellow) between neuronal prediction and image classification.

3.2.2 Methods

The base architecture we are considering in this work is a fully-convolutional variant of the VGG-19 model [Simonyan and Zisserman, 2015]. For the task of image classification, the model is trained on a gray-scale version of the Tiny ImageNet (TIN)[1] task, consisting of 100000 images from 200 ImageNet classes [Deng et al., 2009] downscaled to a resolution of 64 pixels. To allow a fair comparison of image classification performance, we use a VGG model trained using only a cross-entropy classification loss as our *Baseline*.
We additionally collected a set of neuronal responses from the macaque primary visual cortex (V1). The responses were recorded using a 32-channel depth electrode while presenting 24000 gray-scale ImageNet images as stimuli. Based on these recordings, we train a simple model that uses a Gaussian readout [Lurz et al., 2021] on top of VGG layer `conv-3-1` to predict neuronal responses. To obtain what we call the *Monkey Predictor* model, we train the model scratch on the neuronal recordings to optimize the Poisson loss.
Finally, to test the neural co-training hypothesis, we merge the two approaches and perform both image classification and neuronal response prediction in the same model, which we call *MTL Monkey*. The resulting architecture (see Figure 3.1) adds a secondary output with a Gaussian readout on top of layer `conv-3-1` of the VGG-19 model. Consequently, we optimize the model for both the image classification and the neuronal prediction objectives. To find a balance between the two task objectives, we use their corresponding negative log-likelihood and learn a noise parameter σ together for each task to scale its likelihood (following Kendall et al. [2018]), leading to the following combined objective function:

$$\min_{\boldsymbol{\theta},\boldsymbol{\psi}_n,\boldsymbol{\psi}_c,\sigma_c,\sigma_n} \frac{1}{\sigma_c^2}\mathcal{L}_c(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{\psi}_c) + \frac{1}{2\sigma_n^2}\mathcal{L}_n(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{\psi}_n) + \log\sigma_c + \log\sigma_n$$

where $\boldsymbol{\theta}$ are the shared parameters (below `conv-3-1`) and $\boldsymbol{\psi}_c$, σ_c and $\boldsymbol{\psi}_n$, σ_n are the task-specific parameters for $c$lassification and $n$eural prediction, respectively.

[1] https://www.kaggle.com/c/tiny-imagenet/overview

Table 3.1: Overview of the different models used in this study. Showing the data used in training for each and the color associated with each model in subsequent figures.

Model	Classification	Neural Prediction
Baseline	Clean TIN	–
Monkey Predictor	–	Monkey responses
Oracle	Noise augmented TIN	–
MTL-Oracle	Clean TIN	Oracle model responses
MTL-Monkey	Clean TIN	Monkey pred. responses
MTL-Shuffled	Clean TIN	Monkey pred. responses (shuffled)

3.2.3 Results

The main goal of our experiments is to test the neural co-training hypothesis, i.e. to answer whether MTL with neuronal responses can improve the generalization of DNNs. For this purpose, we follow the example of Hendrycks and Dietterich [2019] and evaluate the models' performance on a test set with 14 different common corruptions and compute a robustness score (see Manuscript 1 for details).

In a first experiment, we attempt to verify that MTL can capture robustness if it is present in the transferred representation. For this purpose, we train two models (see Table 3.1 for an overview). The first is an *Oracle* model that has its parameters up to layer `conv-3-1` trained on the same common corruptions that were applied to the robustness test set. The second model, we call *MTL-Oracle*, is optimized with the same co-training setup as MLT-Monkey, but with neuronal data generated by a Gaussian readout on top of conv-3-1 of the Oracle model. The results show improved robustness scores for MTL Oracle close to those of the Oracle model we see as an upper bound for the performance (Fig. 3.2 and Fig. 3.3A,B; dark blue and light blue). This demonstrates that MTL can improve the robustness, if the neuronal responses used for co-training are from a robust source, even though the model has not seen any corrupted inputs in training.

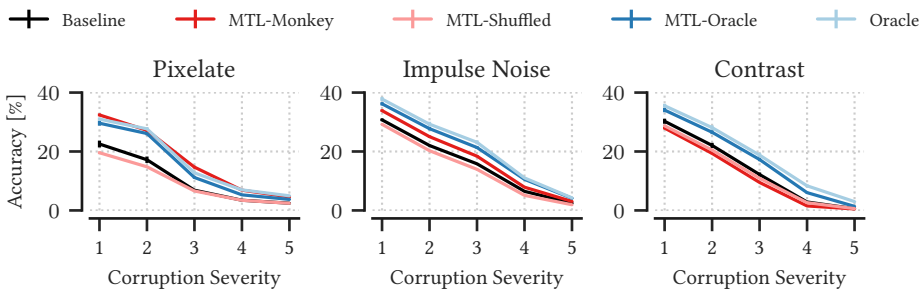


Figure 3.2: Exemplary classification results on TIN-TC. We show results on 3 corruption types with the best (left), median (center), and worst (right) robustness scores for MTL-Monkey across 5 increasing levels of severity each.

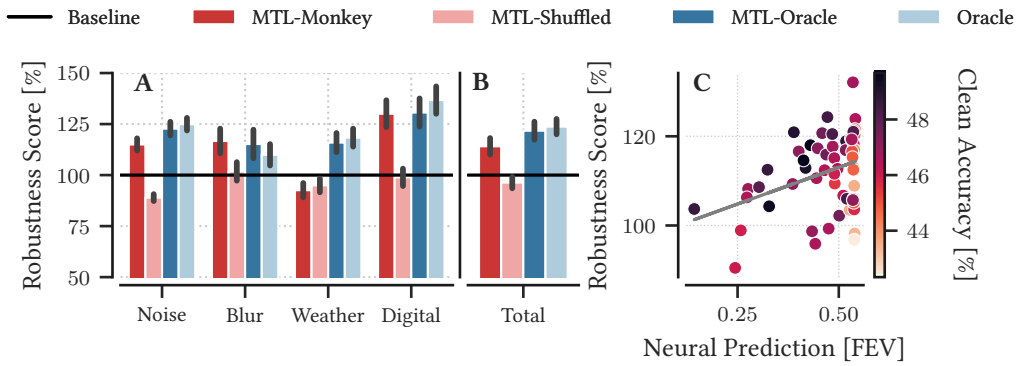


Figure 3.3: **A** Robustness scores for each model grouped by corruption category, as defined in Hendrycks and Dietterich [2019]. **B** Overall robustness scores for our 4 different models. **C** Robustness and neural prediction correlate positively for MTL-Monkey models across 12 different batch ratios and 5 random seeds per model (grey line: linear regression from neural performance to robustness). Neural prediction performance is measured as the fraction of explained variance (FEV), as described in Cadena et al. [2019]. A darker color indicates higher accuracy on the clean TIN test set.

After verifying that MTL works in principle, we set out to test it on real neuronal responses. However, to balance the neuronal prediction and image classification task and to remove the trial-to-trial variability, we use synthetic neuronal responses generated by passing the entire set of TIN training images through the Monkey Predictor model. The MTL Monkey model is trained on this data and the classification task shows increased robustness for 9 out of 14 corruption types (Figure 3.3A; red). The improvement is most prominent for distortions from the Noise, Blur, and Digital categories.

An additional control with MTL training on responses shuffled across images (*MTL-Shuffled*) shows no improvements (Figure 3.3A; light red). This suggests that the generalization benefits come from the original neural data, and not from noise that the neuronal prediction task introduces into the training. To further evaluate this hypothesis, we trained 60 models with a varying training focus for the neuronal prediction and image classification tasks. As intended, the resulting models cover a large range of neuronal prediction performance, clean classification accuracy, and robustness. A regression analysis on this set of models reveals that neural performance on real monkey V1 data is correlated to the network's robustness (Figure 3.3C; $p < 10^{-4}$ for both neural prediction and clean accuracy[2]). Overall, these findings align with prior research demonstrating a positive relationship between model robustness and "brain-like" characteristics [Dapello et al., 2020].

[2]t-test for both factors in a 2-factor linear regression, in which robustness (dependent variable) is predicted from clean test accuracy for image classification and performance on V1 prediction (independent variables).

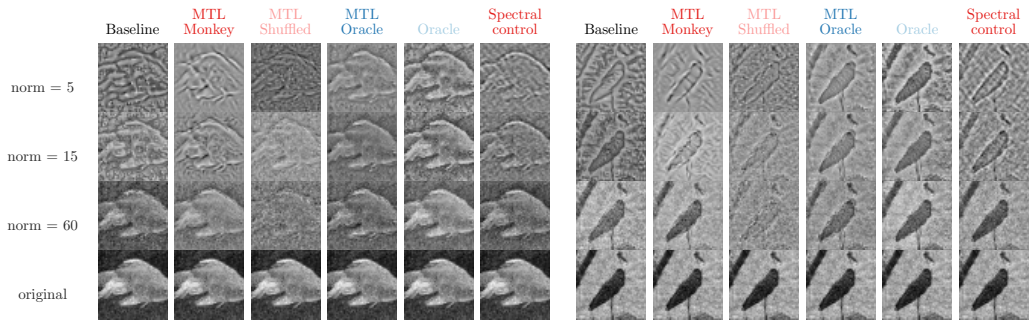


Figure 3.4: Qualitative results of our reconstruction analysis. Examples reconstructed using all 5 models from 4 noisy images (see the last row for original images) with Gaussian noise of severity level 2, under 3 norm constraints: 5, 15, and 60. Please refer to Manuscript 1 for details on the spectral control.

3.2.4 Analyzing the Robustness

After verifying that neural co-training improves robustness, our goal is to understand what caused the improvement in robustness and potentially improve future models through the insight we gain here. For this purpose, we introduce a novel method to visualize the image features the model is sensitive to in order to understand the difference in representation that is induced by the co-training with monkey V1 data. This method reconstructs an image by minimizing squared loss between activations $f(\boldsymbol{x}_0)$ the original image $\boldsymbol{x}_0$ produced in the neural network f and the activations of the reconstruction $f(\boldsymbol{x})$ while constraining the total norm of the image to $r < \|\boldsymbol{x}_0\|$.

$$\boldsymbol{x}^* = \text{argmin}_{\boldsymbol{x}} \|f(\boldsymbol{x}) - f(\boldsymbol{x}_0)\|^2 \quad \text{s.t. } \|\boldsymbol{x}\| \leq r.$$

The idea is to force the model to allocate contrast where it is necessary to recreate the activity of a given layer and thus visualize the sensitivities and invariances of this layer. We optimize the pixels based on the activations of layer `conv-3-1` for all five models *MTL-Monkey*, *Baseline*, *Oracle*, *MTL-Oracle* and *MTL-Shuffled*. Gaussian noise distortions are added to the original image x_0 in order to see if this is reflected in the reconstruction.

The results show qualitative differences between the models (see Figure 3.4). The robust models (MTL-Monkey, MTL-Oracle, and Oracle) seem to focus less on the noise and exhibit more content structure. Nevertheless, there is a noticeable difference between the robust models. MTL-Monkey puts slightly more emphasis on edges and object boundaries than Oracle and MTL-Oracle, which both generally preserve the image content as much as possible. Finally, we quantify this observed difference by comparing the reconstructed image's norm content with the binarized saliency map predicted by the Deep Gaze II model [Kummerer et al., 2017]. The results of this analysis show that MTL-Monkey is more sensitive to salient regions than Oracle and Baseline under low to mid-range norm constraints during reconstruction.

3.2.5 Discussion

The results show this approach to be a successful proof-of-concept for the neural co-training hypothesis [Sinz et al., 2019]. It further shows this method to be useful as a general method in scenarios of representational transfer. Although the achieved results are not competitive yet to state-of-the-art (SOTA) robustness, this still can serve as a conceptual step and an important method that can help align representations for purposes beyond robustness. Furthermore, the transfer of robustness from a robust "teacher" system could even be helpful for SOTA robust models down-the-line, since methods relying on data augmentations alone may not always be sufficient. Especially the fact that data augmentations may struggle with generalization to unseen corruptions [Vasiljevic et al., 2016, Geirhos et al., 2018] or require a carefully calibrated augmentation during training [Rusak et al., 2020], makes the neural co-training approach very appealing. As we demonstrated, our method results in generalization without data augmentation during training. Of course, a potential combination with data augmentations and inclusion of data from higher brain areas [Kietzmann et al., 2019] would likely further increase the usefulness of our method.
In addition to the introduced training framework, we also proposed a novel analytical tool to investigate hidden representations in neural networks, with a particular focus on their sensitivities. This approach helps uncover connections between neural network behavior and the primary visual cortex (V1), which has been linked to saliency in prior research [Zhaoping and Li, 2014], supported by [Li, 2002, Zhaoping and Zhe, 2015, Zhang et al., 2012, Wagatsuma et al., 2021]. Our current analysis is purely correlational and indicates that robust MTL-monkey models appear more sensitive to salient image content than both robust and non-robust models. If this focus on saliency is shown to be causal, it could suggest that the robustness of Oracle models and MTL-monkey models stems from distinct underlying mechanisms.

3.3 Attention Readout for Macaque Area V4

This section is based on the following publication:

> Paweł A. Pierzchlewicz, Konstantin Friedrich Willeke, Arne Nix, Pavithra Elumalai, Kelli Restivo, Tori Shinn, Cate Nealley, Gabrielle Rodriguez, Saumil Patel, Katrin Franke, Andreas S. Tolias, and Fabian H. Sinz. Energy guided diffusion for generating neurally exciting images. *Advances in Neural Information Processing Systems*, 36, 2024

Key takeaways from this section:

- We introduced a novel readout architecture to predict neural activity that is more powerful and can dynamically change its position based on the stimulus.
- The method outperforms the state-of-the-art Gaussian readout by 12% when applied to neurons of area V4 in the macaque visual cortex.
- The improved models, and the additionally introduced Energy-guided diffusion (EGG) method enable us to generate better-performing neurally exiting images.

3.3.1 Motivation

The previous section highlighted the importance of the readout component for modeling neuronal representations and aligning real neuronal responses with representation inside an artificial neural network. Beyond the neural co-training approach, this is crucial for any application that attempts to learn an accurate model of neuronal responses, which is useful to progress toward understanding the brain. Current readouts are generally static with respect to a change in stimulus [Klindt et al., 2017, Lurz et al., 2021]. This means that for each neuron the receptive field is fixed, independent of the presented stimulus. While this is a reasonable assumption for the primary visual cortex, it is questionable whether this assumption also holds for higher or mid-level areas like macaque V4. In particular for V4, research [Tolias et al., 2001] has shown that neurons are affected by attention effects and the receptive field can shift. Motivated by this observation, we proposed a novel readout architecture that dynamically adapts to the receptive field shift and thus allows more accurate prediction of neuronal activity in higher brain areas.

3.3.2 Method

As previously discussed, the model for neuronal predictions consists of a *core* and a *readout* component. For most settings, a four-layer CNN identical to [Lurz et al., 2021] trained in a data-driven fashion will be used for the core network. The readout is a novel architecture (see Figure 3.5 for an overview) that utilizes the attention

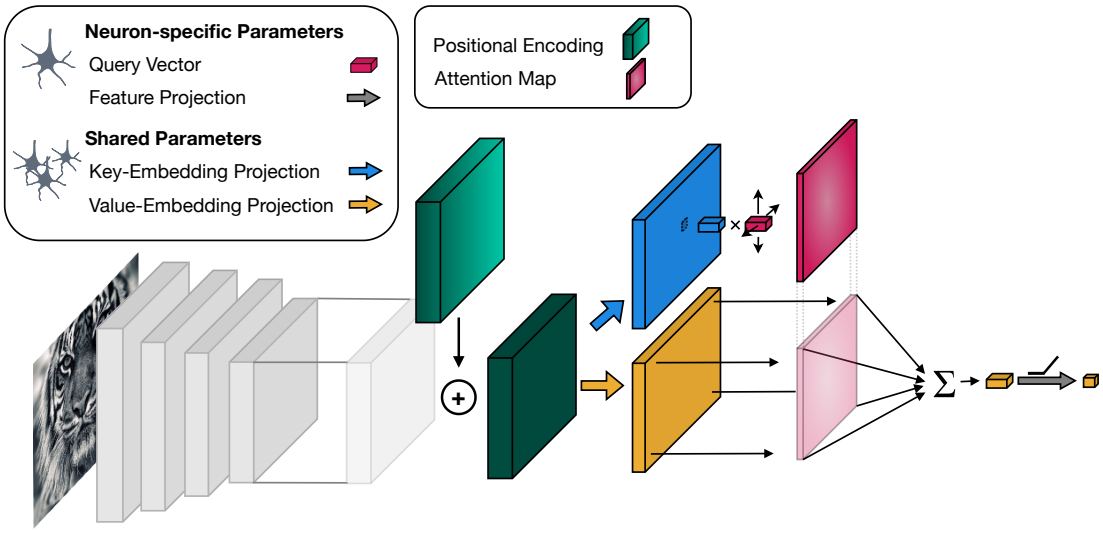


Figure 3.5: Schematic of the Attention Readout. Processing an image through the core (grey), followed by element-wise addition of the positional encoding (green), projection to obtain key-embedding (blue), and value-embedding (yellow). Dot-product computation with the query vector (magenta) and finally weighted sum of the value embeddings, followed by an affine projection and ELU non-linearity.

mechanism [Graves, 2013, Bahdanau et al., 2014, Vaswani et al., 2017]. It is applied to the core output $f(\boldsymbol{x})$ for a specific input $\boldsymbol{x}$. The core features are represented in a three-dimensional tensor that is further processed by adding a positional embedding encoding the location in width and height dimension for each feature vector. After applying an additional Layernorm [Xu et al., 2019] and flattening the spatial dimensions, the resulting output $\tilde{f}(\boldsymbol{x}) \in \mathbb{R}^{c\times(h\cdot w)}$ is presented to the attention mechanism. From this, we extract key and value representations through linear projections $\boldsymbol{V} \in \mathbb{R}^{c\times d_k}$ and $\boldsymbol{U} \in \mathbb{R}^{c\times d_v}$ that are shared across spatial positions and neurons. In contrast to standard attention [Vaswani et al., 2017], the query vectors in our approach are not a product of the input but are instead optimizable parameters $\boldsymbol{q}_n \in \mathbb{R}^{d_k}$ that are learned for each neuron. Key, query, and value are finally used together in a scaled dot-product attention [Vaswani et al., 2017], where the key embedding at each position is compared to each neuron's query vector.

$$\alpha_n = \mathrm{softmax}\left(\frac{\tilde{f}(\boldsymbol{x})^\top \boldsymbol{V}\boldsymbol{q}_n}{\sqrt{d_k}}\right) \tag{3.1}$$

The result is an attention map $\alpha_n \in \mathbb{R}^{h\times w\times 1}$ for each neuron that is normalized over the spatial dimensions and can be interpreted as the receptive field of that neuron. The attention map then is used to compute a weighted sum of the value embeddings. Leaving us with a single feature vector per neuron that is then further transformed into spike rate prediction $\hat{r}_n$ via a neuron-specific affine projection with ELU non-linearity [Clevert et al., 2015]. Combining this readout architecture with the data-driven core described above is referred to as the *attention model*.

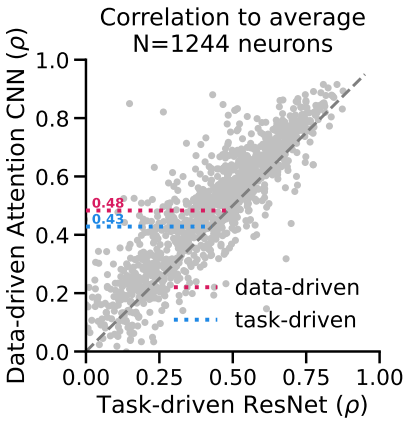


Figure 3.6: Correlation to average scores for 1,244 neurons. The Attention model (pink) shows a significant (as per the Wilcoxon signed rank test, p-value $= 6.79 \cdot 10^{-82}$) increase in the mean correlation to average in comparison to the Gaussian model (blue).

3.3.3 Results

To understand the impact of our model, we compared the attention model to the state-of-the-art approach which is a task-driven model [Willeke et al., 2023] with an adversarially robust ResNet50 as the core [He et al., 2015, Salman et al., 2020] and a Gaussian Readout [Lurz et al., 2021]. This approach will be referred to as *Gaussian model*. The models are trained on data from 1,244 Macaque V4 neurons from Willeke et al. [2023].

We compared the two models' predictions to the average of actual neuron responses across multiple image presentations (similar to Willeke et al. [2023]) by computing their correlation over the test set. Figure 3.6 shows the comparison between the Attention model and the Gaussian model for 1,244 individual neurons. The results show that the Attention model significantly outperforms the Gaussian model in predicting neuronal responses of macaque V4 cells on unseen natural and model-derived images. In total, it is 12% better compared to the Gaussian model (Wilcoxon signed-rank test, p-value $= 6.79 \cdot 10^{-82}$). We additionally show that the Attention model and Gaussian model show representational similarity and that the Attention model uses its ability to shift its receptive field. Finally, an ablation study comparing different choices of core and readout through test correlation (Table 3.2) underlines the superior performance of the Attention readout. In combination with the data-driven core, it outperforms all other setups and even when combined with the task-driven core, its performance is superior to the other readout architectures. Finally, the improved accuracy in the prediction of V4 neurons allows a more thorough analysis of their characteristics. For this purpose, we additionally introduce *energy guided diffusion (EGG)*. A novel technique to generate stimuli conditioned on specific neuron responses using a pre-trained diffusion model. We demonstrate that the method is both more efficient and generates stimuli that generalize better across models. For more details, please refer to Manuscript 2.

Table 3.2: Ablation study. Test correlation scores were compared for combinations of different cores and readouts. Bold indicates the best-performing model. Results show that the attention readout performs best independent of core choice.

Readout \ Core	Task-Driven	Data-Driven
Factorized	-	0.153
Gaussian	0.262	0.229
Attention	0.276	**0.294**

3.4 Discussion

The attention readout architecture described in this section allows us to model higher visual brain areas like V4 more accurately. This can enable many exciting opportunities for further research exploring the neurons of those areas. Analyses like the feature characterization provided by the EGG method will undoubtedly benefit from the improved accuracy of our model. Additionally, the success of the dynamic nature of the attention readout and the changing receptive fields found in the analysis already gives additional support to reports of attention effects in V4 [Tolias et al., 2001].
Finally, the attention readout demonstrates the successful alignment of the biological neuron representations and the internal representations of an artificial neural network. This is promising for an eventual application in our co-training framework, potentially enabling the integration of higher brain areas with the hope of further benefits in generalization.

Functional Transfer Methods 4

The focus of the previous chapter was on the specific case of transferring neuronal representations from recordings of brain activity. We discovered in the results of Section 3.2 that we successfully capture properties of robustness and imbue them into the student model. It remains unknown, however, if the achievements we see in Chapter 3 and in related works [Fong et al., 2018, Li et al., 2019, Federer et al., 2020] achieve their full potential. The reason for this is that we lack deeper insight into the mechanisms of transfer that were used in these works and the knowledge they can and cannot transfer.
In this chapter, we aim to close this gap of knowledge by focusing on the functional transfer methods themselves. This also means taking a step back from the brain as a teacher to achieve better control over the knowledge being transferred. We start by introducing the necessary background on equivariance (Section 4.1.1) and established functional transfer methods (Section 4.1.2). Then we discuss Manuscript 3 in Section 4.2, which analyzes established functional transfer methods both analytically and empirically. The findings show that transferring equivariance is hard and established functional transfer methods cannot guarantee an accurate transfer of equivariant representations. We further introduce Orbit, a novel approach to functional transfer that learns a model of the teacher's equivariance in an intermediate step and uses this to guide the student toward the same equivariance. Building upon our findings and the success of Orbit, we introduce hard augmentations for robust distillation (HARD) in Section 4.3. This method follows the idea of learning equivariance from the teacher but employs it only on the input level by generating augmentations in the teacher's equivariance directions. The result is an approach that is both scalable and effective in combination with other transfer methods, such as knowledge distillation.

4.1 Background

4.1.1 Equivariance

Symmetries, i.e. features of a system that remain unchanged under certain transformations, are present throughout nature. They can be found in all kinds of physical objects as well as the laws that govern the physical world [Feynman, 1966]. Thus it is not surprising that symmetries are often also present in the data we collect to train machine learning models. As the no-free-lunch theorem implies (see Section 2.1.4), generalization in machine learning models requires us to make trade-offs. Identifying and exploiting the symmetries in the data is a way to make smarter trade-offs in order to achieve generalization. Examples in computer vision

[Fukushima, 1980], natural language processing [Gordon et al., 2020], and molecular biology [Thiede et al., 2020] showcase the usefulness of symmetries throughout many application areas in machine learning.
A more general concept than symmetry is the notion of *equivariance* which describes the predictable nature of function outputs under certain transformations on the input side [Cohen, 2021]. The most famous example of equivariance in machine learning is certainly shift equivariance and its corresponding application in CNNs [Fukushima, 1980, LeCun et al., 1998]. Mathematically, this can formalized through symmetry groups [Cohen, 2021], where shift equivariance would be described through a group with elements $(m, n) \in \mathbb{Z}^2$ for shifts of all pixels by m and n positions in horizontal and vertical direction respectively. The group is defined with:

- Group operation: addition $(m, n) + (p, q) = (m + p, n + q)$
- Inverse elements: negative shifts $(m, n) + (-m, -n) = (0, 0)$
- Identity elements: $(0, 0)$

As introduced in Chapter 2, we assume a neural network is defined as composition of functions $f = f^{(L)} \circ ... \circ f^{(1)}$ where a single layer function $f^{(l)} : \mathcal{X}^{(l-1)} \to \mathcal{X}^{(l)}$ operates on the feature space $\mathcal{X}^{(l-1)}$. A group element $g \in G$ acts on $\boldsymbol{x}$ via a linear representation ρ_g. Then $f^{(l)}$ is G-equivariant, if and only if outputs have a action $\rho_g^{(l)}$ corresponding to input transformation $\rho_g^{(l-1)}$

$$f^{(l)}(\rho_g^{(l-1)}\boldsymbol{x}) = \rho_g^{(l)} f^{(l)}(\boldsymbol{x}) \tag{4.1}$$

for all $\boldsymbol{x} \in \mathcal{X}^{(l-1)}$ and any group element $g \in G$. In practice, this means for CNNs that shifts in the input will result in the same shift in the output of a layer. Additional pooling operations on top of convolutional layers will finally extend the equivariance to a full shift invariance, which exploits the fact that the object identity in images is invariant to spatial shifts of that object.
In general, it can be said that equivariances are useful if symmetries are known. It is however a challenging task to identify useful symmetries or to discover them from data. Thus, if we have a system that already uses equivariances, transferring those to another system is of great importance.

4.1.2 Transfer Methods

Transferring properties like equivariance between neural networks requires a method that is flexible to capture the properties in the teacher network f_t and simultaneously restrictive enough to enforce it in the student network f_s. For all functional transfer methods we discuss here, we denote the optimization as minimizing the standard task loss $\mathcal{L}$ with an added term Ω that defines the transfer:

$$\min_{\boldsymbol{\theta}_\mathrm{s}} \mathcal{L}(\boldsymbol{x}; \boldsymbol{\theta}_\mathrm{s}) + \Omega(\boldsymbol{x}; \boldsymbol{\theta}_\mathrm{s}, \boldsymbol{\theta}_\mathrm{t}) \tag{4.2}$$

In the following, we will discuss some examples of different transfer methods.[1]

[1] Please note that we adapted the notation here and in Section 4.2 slightly compared to Manuscript 3 in order to unify notation with the rest of this thesis and ease understanding.

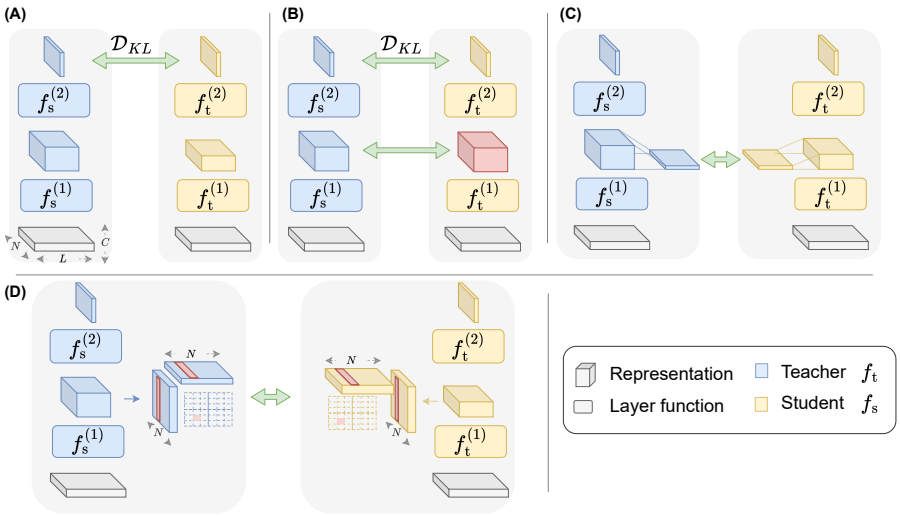


Figure 4.1: Overview of the transfer methods discussed in this work: **(A)** Knowledge Distillation, **(B)** Direct matching, **(C)** Attention transfer, **(D)** Representational Distance Learning.

Knowledge Distillation The most widely known functional transfer method in machine learning is Knowledge Distillation (KD) [Hinton and Dean, 2015]. Originally developed for classification tasks, it minimizes the Kullback-Leibler divergence between the softmax output of teacher and student (see Figure 4.1.A).

$$\Omega = p_{\mathrm{t}}(\boldsymbol{x}) \log \frac{p_{\mathrm{t}}(\boldsymbol{x})}{p_{\mathrm{s}}(\boldsymbol{x})} \tag{4.3}$$

Where $p_{\mathrm{t}}(\boldsymbol{x})_i = \frac{\exp(f_{\mathrm{t}}(\boldsymbol{x})_i/\tau)}{\sum_j \exp(f_{\mathrm{t}}(\boldsymbol{x})_j/\tau)}$ and $p_{\mathrm{s}}(\boldsymbol{x})_i = \frac{\exp(f_{\mathrm{s}}(\boldsymbol{x})_i/\tau)}{\sum_j \exp(f_{\mathrm{s}}(\boldsymbol{x})_j/\tau)}$ are the typical outputs of classification networks, with a scaling parameter $\tau \in \mathbb{R}^+$ as temperature control. For regression tasks, the equivalent would be *functional distance regularization* [Benjamin et al., 2019], which minimizes the Euclidean distance between the function values of the two tasks. Noteworthy here is that KD is very flexible and easy to apply, but limits the effect on the student network to the output level and additionally requires the two to perform the same task.

Direct Matching A larger effect on the student network can be achieved by minimizing the distance of student and teacher representations not only on the output level but also on the level of intermediate layers (see Figure 4.1.B). This *direct matching* approach is a very strong restriction on the student's representations and would therefore be expected to be most effective in transferring the teacher's properties. Applying this method requires the networks to be of identical size for all layers that are supposed to be matched, which would in most cases also allow to copy the weights as a simple alternative.

Attention Transfer If not all representations are of the same size, it may be possible to exploit the fact that most model representations maintain a structure with

a certain semantic interpretation. In the case of computer vision, for example, most hidden representations have a width, height, and channel dimension. Furthermore, most networks have a similar spatial extend (width and height) at a similar relative depth through the network. *Attention Transfer* uses this insight to match neural networks along those spatial dimensions by collapsing the channel dimension through summation or max-pooling. The collapse generates an attention map[2] for each representation with $A_{\mathrm{t}}^{(l)}(x) = \sum_{c=1}^{C_{\mathrm{s}}^{(l)}} |f_{\mathrm{t}}^{(l)}(\boldsymbol{x})_c| \in \mathbb{R}^{w \times h}$ and $A_{\mathrm{s}}^{(l)}$ defined analogously, as well as channel sizes $C_{\mathrm{s}}^{(l)}$ and $C_{\mathrm{t}}^{(l)}$. Those attention maps should be of comparable size for teacher and student (additional up-/down-sampling applied if not) and can be used to minimize the distance between the two representations:

$$\Omega^{(l)} = \left\| \frac{A_{\mathrm{t}}^{(l)}(\boldsymbol{x})}{\left\| A_{\mathrm{t}}^{(l)}(\boldsymbol{x}) \right\|_2} - \frac{A_{\mathrm{s}}^{(l)}(\boldsymbol{x})}{\left\| A_{\mathrm{s}}^{(l)}(\boldsymbol{x}) \right\|_2} \right\|_2^2$$

This method can be applied to more distinct teacher and student architectures but trades this increased flexibility for less precise matching due to the channel collapse.

Representational Distance Learning There is a plethora of methods that allow even more flexibility in comparing neural network representations than attention transfer. Kornblith et al. [2019] provides a comprehensive overview of these *representational similarity methods*. The general idea unifying most of these methods is to consider representations for an entire set (a batch) of inputs. Formally this means viewing flattened representations of neural network f_{t} at layer l for a batch of inputs $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \in \mathcal{X}^{(0)}$ as matrix $\boldsymbol{\Phi}_{\mathrm{t}}^{(l)} = [f_{\mathrm{t}}^{(l)}(\boldsymbol{x}_1), \ldots, f_{\mathrm{t}}^{(l)}(\boldsymbol{x}_N)]^\top$ and similarly $\boldsymbol{\Phi}_{\mathrm{s}}^{(l)} = [f_{\mathrm{s}}^{(l)}(\boldsymbol{x}_1), \ldots, f_{\mathrm{s}}^{(l)}(\boldsymbol{x}_N)]^\top$ for network f_{s}. The representation matrices then have dimensions $\mathbb{R}^{N \times d}$ and $\mathbb{R}^{N \times d'}$ for $\boldsymbol{\Phi}_s$ and $\boldsymbol{\Phi}_t$ respectively. Comparing the two matrices along the axis representing the different samples, allows us to compute a similarity even if $d \neq d'$.
Representational Distance Learning (RDL) [McClure and Kriegeskorte, 2016] is the best-known method applying this concept to functional transfer. The approach computes a *Representational Dissimilarity Matrix (RDM)* for each network, which is essentially a Gram matrix of the batch representations $\boldsymbol{\Phi}^{(l)}$ with additional normalization. Every entry (n, m) in this RDM represents the dissimilarity of samples $\boldsymbol{x}_n$ and $\boldsymbol{x}_m$ w.r.t. representation $\Phi^{(:l)}$. RDMs are computed for student and teacher respectively and then used to compute a representation distance between the two networks as the Frobenius norm between their RDMs:

$$\Omega^{(l)} = \frac{1}{N^2} \| \boldsymbol{\Phi}_t^{(l)} \boldsymbol{\Phi}_t^{(l)\top} - \boldsymbol{\Phi}_s^{(l)} \boldsymbol{\Phi}_s^{(l)\top} \|_F^2$$

Minimizing this distance gives a representational transfer method that is agnostic to both the task of each network and the shape of their representations.
We have now summarized some of the most important functional transfer methods. Although there exist countless other methods that work similarly or in some cases even better, we focus on the four presented above for our studies, as their

[2] not to be confused with the attention used in transformers [Vaswani et al., 2017]

performance is well established and they cover all levels of complexity from simple output-level transfer to individual layer matching. In the following sections, we will examine these methods, investigate their usefulness in transferring generalizing representations, and finally offer new methods that improve upon what we found.

4.2 Can Functional Transfer Methods Capture Simple Inductive Biases?

This section is based on the following publication:

> Arne Nix, Suhas Shrinivasan, Edgar Y Walker, and Fabian Sinz. Can Functional Transfer Methods Capture Simple Inductive Biases? In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 10703–10717. PMLR, 2022a

Key takeaways from this section:

- We evaluate functional transfer methods and show that they fail to transfer even basic shift equivariance.
- Theoretical analysis reveals that representational similarity methods cannot guarantee equivariance transfer.
- We propose a novel method that learns equivariance from a teacher network and regularizes the student to adopt it, successfully demonstrating this on shift and rotational equivariance.

4.2.1 Motivation

The goal of functional transfer methods varies across different applications. Applications in model compression [Bucilă et al., 2006, Hinton and Dean, 2015] might have a different aim than continual learning [Pan et al., 2020, Titsias et al., 2019, Benjamin et al., 2019] or neuroscience [Li et al., 2019]. The common denominator for all these examples is the transfer of "useful knowledge". However, it is often unclear how effective different methods are in achieving this. To understand this, we first need to define what "useful knowledge" could mean. In this thesis, the knowledge we focus on for transfer is inductive biases, which are not only useful but even essential for generalization in machine learning (see Section 2.1.4). One example of such an inductive bias is equivariance, as it can help a model with generalization in the corresponding symmetry directions. Since equivariances can be mathematically characterized and it is easy to induce with certain architecture choices in some cases, we choose to focus on this class of inductive biases for this analysis. Therefore, we want to go beyond previous efforts [Abnar et al., 2020] and conclusively answer: *Can functional transfer methods effectively transfer equivariance properties between a student and a teacher?*

4.2.2 Analysis

To answer the question posed in the previous section, we devised a simple and easily configurable task based on the MNIST-1D [Greydanus, 2020] dataset as an empirical

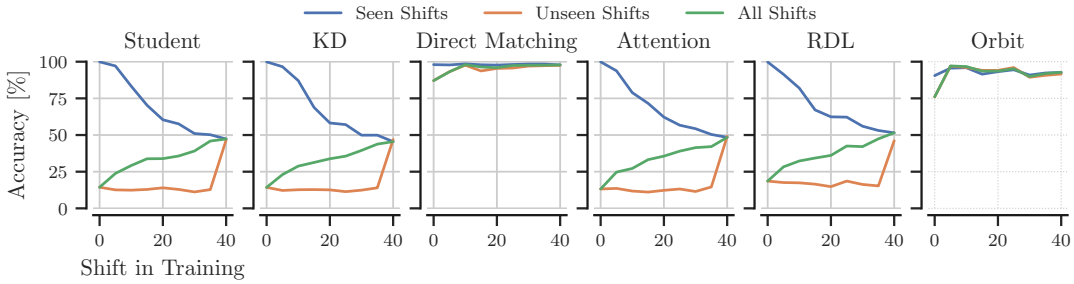


Figure 4.2: Test performance after transfer from a fully convolutional teacher to a fully connected student. We perform the transfer for different degrees of shift in the training data (x-axis) and evaluate seen (blue line) and unseen shifts (orange line) as well as both combined (green line). The results show that only direct matching and Orbit manage to transfer the teacher's shift equivariance successfully.

test. Using the scripts from Greydanus [2020], we procedurally generated 4000 synthetic training and 1000 synthetic test inputs, which represent digits 0 to 9 as 40-dimensional vectors. Through the generation procedure, we have fine-grained control over the shift of the digit representation along the length of the vector. We denote this as the *shift limit* s, which represents the upper limit of the range the shift is randomly sampled from for each input. The data used for training will only contain samples with shifts in the range of $[0, s]$. This allows us to split the evaluation data into *seen shifts* $[0, s]$ and *unseen shifts* $(s, 39]$, as well as *all shifts* $[0, 39]$. Each of these variants will be its own test set, allowing us to evaluate generalization to unseen shifts. We additionally standardize teacher and student to be small CNN and fully connected networks respectively. We train models for $s \in \{0, 10, 20, 30, 40\}$ so that as s increases, the *seen shifts* test set covers more shifts and the *unseen shifts* less. The baseline student network degrades on seen shifts for settings with more shifts in the training (larger s) (see Figure 4.2) and performance on unseen shifts stays consistently bad until $s = 40$ where all shifts are seen, demonstrating the lack of inductive bias in the plain fully connected student.

Knowledge Distillation Standard KD matches the final output of the student's network function to the teacher function. This means that the student should behave the same as the teacher if zero training loss is reached. It is tempting to think that this would also mean that the teacher's equivariance is transferred. Especially, since experiments by Abnar et al. [2020] demonstrate that there is a successful transfer of shift and scale invariance to some extent with KD. However, in a case where the training data is finite, it may happen that not all orbit elements of the symmetry group are contained in the training data and thus the student's output for these elements remains arbitrary. Our experiments explore this behavior by comparing the true generalization performance on seen and unseen shifts. The results clearly show that KD is not improving the student's performance over the baseline when evaluated on unseen shifts (see Figure 4.2). We hypothesize that this is due to insufficient strength of the constraints that KD imposes on the student during training. Simply matching on the output level is not enough.

Direct Matching Since KD appears to lack the constraint power to transfer equivariance, we subsequently explore the other extreme, where we employ direct matching to guide the output of each layer towards the teacher's activation at the same level. This means in practice that we minimize the Mean-Squared Error (MSE) to the teacher representations on all internal layers and the KL-Divergence on the final layer output. As Figure 4.2 demonstrates, direct matching achieves nearly perfect performance on unseen shifts, even if only a small part of the full orbit is present in the training data. However, this approach is rarely practically relevant since student and teacher architectures are required to share the same output shapes on all layers that are matched. In cases where the teacher and student are this similar, it may be more effective to perform the transfer on the parameter level instead.

Attention Transfer Attention transfer offers slightly more flexibility than direct matching by aligning teacher and student in the spatial dimension for each layer. Intuitively, this should be beneficial for a transfer of shift equivariance. However, we do not see this effect in practice (Figure 4.2), where the performance on unseen shifts does not perform over the baseline level. The likely explanation for this is that important information is lost during the collapse of the channel dimension. Making attention transfer a poor choice for transferring equivariance.

Representational Distance Learning RDL is even more flexible than attention transfer, while still allowing all layers to be guided by the teacher. The experiments show that it has shortcomings similar to all the other methods. This result is especially interesting to us, since RDL's flexibility allows for a teacher of arbitrary shape, opening the avenue towards using brain activity as the teacher signal [Li et al., 2019]. For this reason, we focused on RDL for our theoretical analysis and asked: Can RDL guarantee the transfer of equivariance?

Theoretical Analysis In standard RDL [McClure and Kriegeskorte, 2016], the optimal solution for the optimization would be $\mathbf{\Phi}_\mathrm{t}\mathbf{\Phi}_\mathrm{t}^\top = \mathbf{\Phi}_\mathrm{s}\mathbf{\Phi}_\mathrm{s}^\top$. This equality is true if and only if $\mathbf{\Phi}_\mathrm{t} = \boldsymbol{Q}\mathbf{\Phi}_\mathrm{s}$ with $\boldsymbol{Q} \in SO(N)$ [Co, 2013, theorem 7.3.11]. That means RDL guarantees that the student matches the teacher's representation up to an orthogonal transformation Q. This carries over to other representational similarity measures invariant to orthogonal transformation, which includes most mentioned by Kornblith et al. [2019]. In the limit of infinite data, this implies that not only are the representations but also the functions identical up to orthogonal transformation: Thus, we have $f_\mathrm{t} = \boldsymbol{Q} f_\mathrm{s}$ (slight abuse of notation). Based on this observation, we prove the following lemma in the Manuscript 3:

Lemma 1 *Given two representations $\mathbf{\Phi}_\mathrm{s}$ and $\mathbf{\Phi}_\mathrm{t}$ and corresponding functions with relationship $f_\mathrm{t} = \boldsymbol{Q} f_\mathrm{s}$ for some orthogonal $\boldsymbol{Q}$, the following holds: f_t is equivariant w.r.t. group representation $(\rho^{(0)}, \rho^{(1)})$ if and only if f_s is equivariant w.r.t. group representation $(\rho^{(0)}, \boldsymbol{Q}\rho^{(1)}\boldsymbol{Q}^\top)$.*

The consequence of this lemma is that representational similarity methods cannot guarantee a transfer of equivariance properties w.r.t. the same group representation on the input and output side. The student will be equivariant w.r.t. the transformed

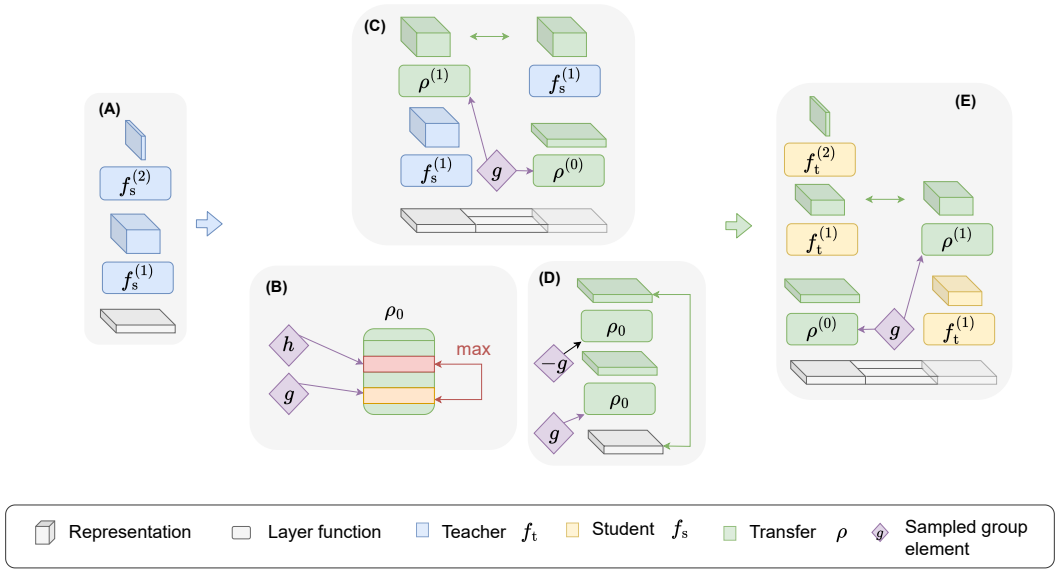


Figure 4.3: Overview of all steps involved in our orbit transfer method: **(A)** Training the teacher. **(B)** Maximizing dissimilarity between distinct group elements. **(C)** Learning the equivariance. **(D)** Encourage inverse elements to cancel each other out. **(E)** Transferring the learned equivariance to the student.

group representation $\boldsymbol{Q}\rho^{(1)}\boldsymbol{Q}^\top$ instead of the teacher's group representation $\rho^{(1)}$. Thus, RDL does transfer equivariance but might get a different linear representation of the group. The introduced orthogonal transformation $\boldsymbol{Q}$ may destroy shift invariance, since the supremum norm and, therefore, max-pooling is not invariant under rotation.

To summarize our theoretical and empirical findings, we are confident that we can answer the question *"Can functional transfer methods effectively transfer equivariance properties between a student and a teacher?"* with no. The methods cannot guarantee the transfer of the teacher's equivariance and therefore are insufficient for cases where this is a main objective. Building upon this insight, in the following section, we introduce a novel approach to functional transfer that is specifically designed to achieve this.

4.2.3 Method

Based on the results from the previous section, we hypothesize that matching the function of the entire network or even that of individual layers is too broad a task to reliably transfer specific equivariances. Thus, we propose a novel method – *Orbit transfer* – that decouples the equivariance property from the teacher function itself. The goal is for the student to fulfill $f_s(\rho_0 \boldsymbol{x}) = \rho_1(f_s(\boldsymbol{x}))$ after training, where ρ is the same group representation that the teacher is equivariant to, i.e. $f_t(\rho_0 \boldsymbol{x}) = \rho_1(f_t(\boldsymbol{x}))$. We approach this in two steps. First, we learn the group representation from the teacher and then we encourage the same equivariance in the student (see Figure 4.3 for an overview).

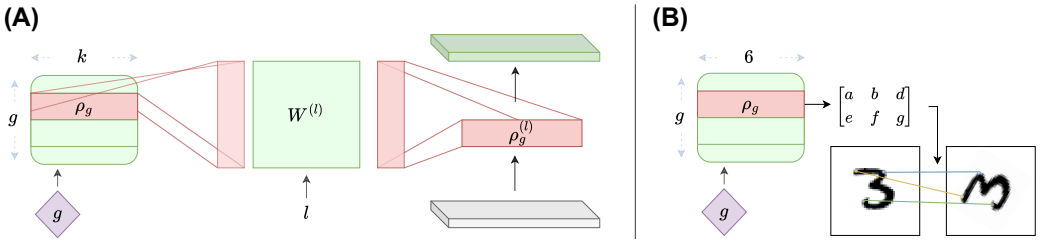


Figure 4.4: Illustration of the two parametrizations we propose for the equivariance model. **(A)** The group representation model first selects a filter of size k based on the group element g, then it applies the linear transformation $W^{(l)}$ to finally get the group representation that can be applied as a convolutional filter on the input x to get $\rho_g^{(l)} x$. **(B)** Each group element is an affine transformation (2D on input; 3D on internal representations) that is applied to the representation.

Learning the Equivariance from the Teacher To capture the teacher's equivariance, we learn a model[3] ρ that fulfills $f_{\mathrm{t}}(\rho_g^{(0)} \boldsymbol{x}) = \rho_g^{(1)} f_{\mathrm{t}}(\boldsymbol{x})$ for any given g and $\boldsymbol{x}$. This can be achieved by freezing the teacher and optimizing ρ's parameters to minimize:

$$\mathcal{L}_{\text{equiv}} = \frac{1}{H} \| f_{\mathrm{t}}(\rho_g^{(0)} \boldsymbol{x}) - \rho_g^{(1)} f_{\mathrm{t}}(\boldsymbol{x}) \|_2^2 \tag{4.4}$$

where H is the size of the output from layer function f_{t}.
To avoid a collapse to the trivial solution where ρ learns the identity function, we additionally minimize the cosine similarity (maximize dissimilarity) between representations of distinct group elements:

$$\mathcal{L}_{\text{group}} = \frac{1}{|G| \cdot (|G| - 1)} \sum_{g \in G} \sum_{h \in G \setminus \{g\}} |\cos(\rho_g^{(0)}, \rho_h^{(0)})| \tag{4.5}$$

We additionally encourage ρ to model a group where inverse group elements cancel each other out:

$$\mathcal{L}_{\text{inv}} = \| \rho_{-g}^{(0)} \rho_g^{(0)} \boldsymbol{x} - \boldsymbol{x} \|_2^2 \tag{4.6}$$

The final model ρ represents all group elements g of the equivariance orbit and optimizes these three objectives in addition to the standard cross-entropy loss, while the teacher's parameters are frozen:

$$\min_{\rho} \gamma_{\text{CE}} \mathcal{L}_{\text{CE}} + \gamma_{\text{equiv}} \mathcal{L}_{\text{eqiv}} + \gamma_{\text{group}} \mathcal{L}_{\text{group}} + \gamma_{\text{inv}} \mathcal{L}_{\text{inv}} \tag{4.7}$$

with hyper-parameters $\gamma_{\text{CE}}, \gamma_{\text{equiv}}, \gamma_{\text{group}}, \gamma_{\text{inv}} \in \mathbb{R}^+$.

[3]Since our goal is to learn the group representation the teacher is equivariant to, we use the same notation as above for the equivariance model: $\rho_g^{(l)}$ for layer l and group element g. With a slight abuse of notation, we also refer to the parameters of the equivariance model with the same symbol.

Table 4.1: MNIST (column 1 and 3) and MNIST-C (column 2 and 4) test results for four different transfer methods. The left two columns show the transfer results from a small CNN teacher to an Multilayer Perceptron (MLP) student. The right columns show analogous experiments between a ResNet18 teacher and a small Vision Transformer (ViT) student. The best-performing transfer is shown in bold for each column.

	CNN → MLP		ResNet18 → ViT	
Method	Centered	Translated	Centered	Translated
Teacher	99.0	93.4	99.6	90.3
Student	98.5	35.7	98.6	37.3
+ Augment	54.3	97.0	56.5	97.4
KD	**98.8**	41.1	98.8	41.2
Attention	98.4	31.9	—	—
RDL	98.6	31.9	**99.3**	59.6
Orbit	**98.8**	**95.2**	98.4	**84.0**

Group Representation Architecture The optimization of the group representation described above is intentionally kept agnostic to the choice of architecture. Important is only that the parametrization differentiates between different group elements and layers. We propose two different architectures for modeling equivariance here. The first group representation model learns convolutional filters for each group element g and a linear projection is applied to the filter that is shared across group elements, but different for each layer (see Figure 4.4 for more details). The second, more general model, learns an affine transformation matrix $\rho_g^{(l)} \in \mathbb{R}^{2\times 3}$ ($\mathbb{R}^{3\times 4}$ for internal representations) akin to spatial transformer models [Jaderberg et al., 2015] for each group element g and layer l and applies this transformation across spatial and feature dimensions.

Training the Student to Have the Same Equivariance Finally, the learned equivariance ρ that was extracted from the teacher is used to train the student. For this, the same equivariance objective we defined above (Equation 4.4) is used again but applied to the student instead of the teacher and with the orbit model frozen.

$$\Omega^{(l)} = \| f_{\mathrm{s}}(\rho_g^{(0)}\boldsymbol{x}) - \rho_g^{(1)}(f_{\mathrm{s}}(\boldsymbol{x}))\|_2^2 \tag{4.8}$$

Additionally, $\rho_g^{(0)}$ is also used to augment data for the standard cross-entropy loss.

4.2.4 Results

Evaluating the Orbit transfer approach in our MNIST-1D setup shows that our method successfully captures the equivariance and transfers it to the student. The results outperform all other methods except for direct matching in generalization to unseen shifts (Figure 4.2). Orbit transfer also helps when we replace the pooling layer in the student with a linear layer, in contrast to direct matching which breaks down in this scenario.

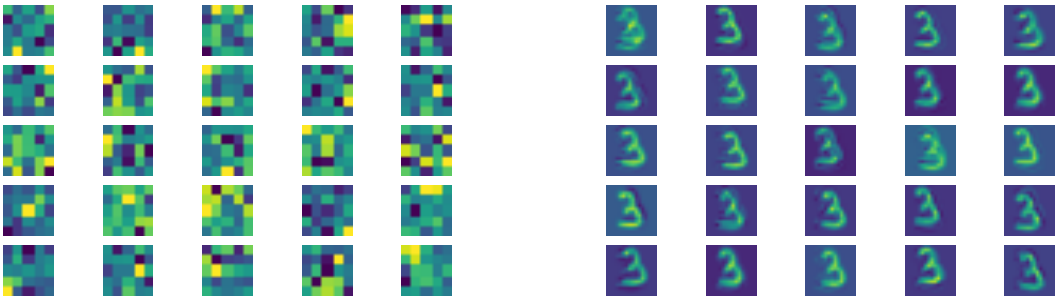


Figure 4.5: Quantiative analysis of the learned equivariance model. Rows 1-5: Kernels for all 25 linear representations of the group elements learned in the equivariance model. Rows 6-10: Kernels (in identical order) applied to an example input.

After the analysis on the MNIST-1D toy task, we move to a slightly more complex setup with MNIST [Deng, 2012]. Here we evaluate both a small setup with a three-layer CNN teacher and a fully connected student of the same depth as well as a larger setup with a ResNet-18 [He et al., 2015] as the teacher and a six-layer ViT [Dosovitskiy et al., 2020] as student. We train on the standard MNIST training set, and afterward evaluate both on the standard centered test set and a randomly translated version of it [Mu and Gilmer, 2019]. The results confirm the findings from the MNIST-1D setting. Shift equivariance is hard for both student networks as they have by default limited inductive bias. Also, established transfer methods again fail to transfer the equivariance, which shows in their lacking performance on the translated test set. Our Orbit transfer approach on the other hand shows strong results for both student-teacher pairings on both centered and translated test set (+59.5% for MLP, +46.7% for ViT).

Another benefit of our Orbit transfer method is that it allows us to analyze the transformations we learn from the teacher by inspecting both the learned kernels and their effect on the input images (Figure 4.5). Qualitative analysis reveals filters similar to what would be expected for kernels for perfect translation: A single non-zero position and all other entries close to zero. The impression becomes even clearer when we apply the filters to an example image, which shows that the filters shift the digit noticeably in different directions.

Finally, we test our method on a different form of equivariance. By using a Group-Convolutional Neural Network (G-CNN)[Cohen and Welling, 2016] as the teacher, we attempt to transfer equivariance w.r.t. rotations of multiples of 90 degrees to our MLP student. The training remains on the standard MNIST data and evaluation is done on the standard test set as well as a randomly rotated version of it. We again see the established methods struggle with transfer when evaluated on the rotated test images. Here orbit is modeled as learnable affine transformations in 3D, which is more general and harder to optimize. Nevertheless, Orbit performs remarkably well (+37.0%) on rotated images.

Table 4.2: Results on the MNIST test set with images in upright (column 1) or randomly rotated (column 2) orientation for four different transfer methods for a G-CNN teacher and an MLP student.

Method	Upright	Rotated
Teacher (G-CNN)	99.1	87.7
Student (MLP)	98.3	39.5
KD	98.7	46.3
RDL	**99.0**	45.6
Attention	98.3	41.1
Orbit	97.5	**76.5**

4.2.5 Discussion

While Orbit transfer demonstrates significant improvements over existing functional transfer approaches, several important challenges remain to be addressed. The optimization of affine transformations suffers from local minima, making the method potentially unstable and difficult to optimize effectively. This suggests the need for alternative parameterizations or optimization strategies. The computational cost of applying Orbit transfer presents a practical limitation, particularly when considering multiple layers simultaneously. Additionally, while the method provides clear interpretability at the input level through visualization of kernels and augmented input examples, understanding equivariance in hidden layers remains challenging. It is often unclear ahead of time how equivariance properties translate into these internal representations.
Looking forward, scaling the method to larger models and more complex datasets represents a crucial next step. This expansion would need to address both the computational challenges and optimization difficulties identified in our current implementation. Despite these limitations, Orbit transfer's success in transferring both translational and rotational equivariance suggests promise for transferring other important symmetries in deep learning applications.

4.3 Hard Augmentations for Robust Distillation

This section is based on the following publication:

> Arne F Nix, Max F Burg, and Fabian H Sinz. Hard: Hard augmentations for robust distillation. *arXiv preprint arXiv:2305.14890*, 2023

Key takeaways from this section:

- We introduce Hard Augmentations for Robust Distillation as a method to generate augmented or entirely synthetic data points for which the teacher and the student disagree.
- We show in a simple toy example that our augmentation framework solves the problem of transferring simple equivariances with KD.
- When applied to real-world tasks, our method outperforms even state-of-the-art data augmentations and since the augmented training inputs can be visualized, they offer a qualitative insight into the properties that are transferred from the teacher to the student.

4.3.1 Motivation

The work described in the previous section showed that current KD methods fail to transfer even simple equivariances between teacher and student (see Figure 4.2). However, carefully analyzing the results presented throughout the literature reveals that student and teacher after KD are separated by a larger gap in OOD performance than on ID data [Oquab et al., 2023, Beyer et al., 2021] (see appendix of Manuscript 4). This phenomenon is a generalization of the effects we saw for transferring equivariance. The knowledge that helps with generalization on OOD data is particularly hard to transfer. Based on these observations, and the insights from the Orbit transfer experiments, we hypothesize that the choice of input data is crucial for successful functional transfer and we illustrate the impact of training data by a simple toy example.
In this example, the goal is to transfer from the ground-truth teacher function $f_\mathrm{t}(x) = \cos(x)$ to a three layer MLP student $f_\mathrm{s}(x)$ with ReLU activations. The training data is chosen such that it does not capture the entirety of the teacher's periodicity (orange points in Figure 4.6A). Therefore the standard KD approach that tries to minimize the distance between the teacher's output and the student's output also fails to capture the periodicity and thus generalizes poorly. We further show that applying the equivalent of MixUp [Zhang et al., 2017] (Figure 4.6B,F) leads to better interpolation and additive Gaussian noise (Figure 4.6C,G) leads to extrapolation in a range limited by the chosen standard deviation. Clearly, we could improve interpolation and extrapolation by increasing the noise distribution's variance or shifting its mean, however, as we move to a high dimensional image input space ($x \in \mathbb{R} \rightarrow \boldsymbol{x} \in \mathbb{R}^N$) it becomes unclear how to heuristically select new samples that benefit generalization. Instead, we propose to optimize a parameterized augmentation

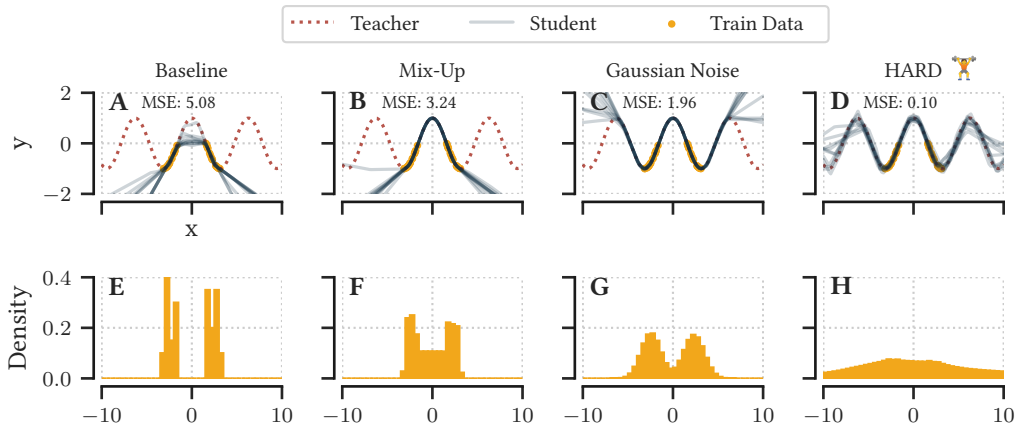


Figure 4.6: Motivating example for the HARD approach. Fitting the student, a three-layer ReLU MLP, to the teacher function, $\cos(x)$, for $10,000$ iterations. We show results for 10 random seeds (**A**-**D**) and the distribution of (augmented) training inputs as a normalized histogram (**E**-**H**). We compare a baseline (no augmentations) with Mixup, Gaussian noise, and an HARD-optimized noise distribution. We report MSE on 100 test inputs sampled from $\mathcal{U}_{[-10,10]}$.

to efficiently generate new, hard training samples on which the student lacks performance, as here the student could improve the most. In our toy example, we illustrate this by optimizing the Gaussian's parameters (mean and standard deviation) according to our augmentation framework *HARD Augmentations for Robust Distillation (HARD)*, which we will present in the next section. The resulting transfer leads to a much wider generalization (Figure 4.6D,H), showing that our approach can capture the desired inductive bias from the teacher by selecting the right input samples.

4.3.2 Method

Learning Hard Augmentations for Robust Distillation (HARD) The goal of this framework is to learn a parametrized augmentation model g_a that generates new input data points $\tilde{\boldsymbol{x}} = g_\mathrm{a}(\boldsymbol{x})$ in areas where teacher and student disagree. This "disagreement" will be measured by the *teacher-student loss* between student and teacher on augmented inputs $\tilde{x} \in \mathbb{R}^n$:

$$\mathcal{L}_{\tilde{\mathrm{s}}\leftrightarrow\tilde{\mathrm{t}}} = \mathcal{D}\left[f_\mathrm{s}(\tilde{\boldsymbol{x}}), f_\mathrm{t}(\tilde{\boldsymbol{x}})\right] \ . \tag{4.9}$$

We further guide the augmentations towards directions of useful knowledge through the *teacher-teacher loss*

$$\mathcal{L}_{\tilde{\mathrm{t}}\leftrightarrow\mathrm{t}} = \mathcal{D}\left[f_\mathrm{t}(\tilde{\boldsymbol{x}}), f_\mathrm{t}(\boldsymbol{x})\right] \ , \tag{4.10}$$

that is going to be minimal for augmentations the teacher is invariant to. We optimize the augmentor's parameters $\boldsymbol{\theta}_\mathrm{a}$ to maximize $\mathcal{L}_{\tilde{\mathrm{s}}\leftrightarrow\tilde{\mathrm{t}}}$ and minimize $\mathcal{L}_{\tilde{\mathrm{t}}\leftrightarrow\mathrm{t}}$ (Figure 4.7 bottom). Simultaneously, we optimize the student's parameters $\boldsymbol{\theta}_\mathrm{s}$ following the

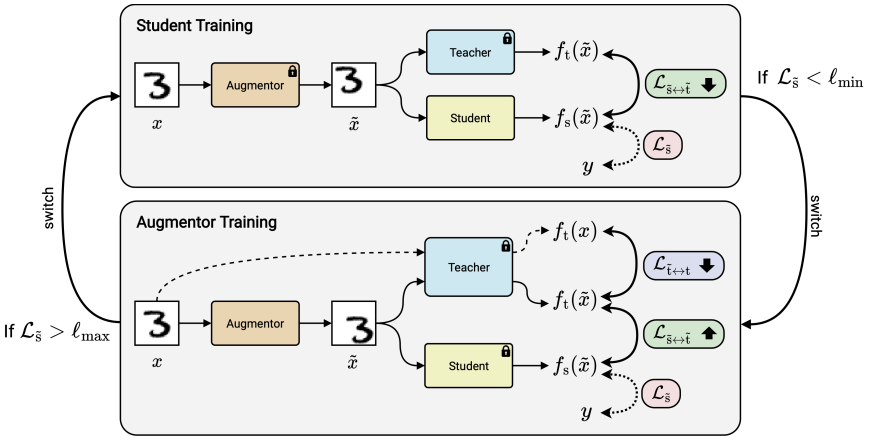


Figure 4.7: Overview of our task-agnostic HARD framework. Our approach switches between training the student to match the teacher and training the augmentor to generate new samples on which the student underperforms while maintaining high teacher performance. We optimize the augmentor and student in interchanging phases through a student-teacher loss $\mathcal{L}_{\tilde{s}\leftrightarrow\tilde{t}}$ and teacher-teacher loss $\mathcal{L}_{\tilde{t}\leftrightarrow t}$. We switch between the two phases by comparing the default loss $\mathcal{L}_{\tilde{s}}$ on augmented data to pre-defined thresholds.

standard KD convention to minimize the student-teacher distance on the augmented images (Figure 4.7 top).

$$\max_{\boldsymbol{\theta}_a} \lambda_s \mathcal{L}_{\tilde{s}\leftrightarrow\tilde{t}} - \lambda_t \mathcal{L}_{\tilde{t}\leftrightarrow t} \qquad \text{and} \qquad \min_{\boldsymbol{\theta}_s} \mathcal{L}_{\tilde{s}\leftrightarrow\tilde{t}} \,. \tag{4.11}$$

Here λ_s and λ_t trade off the loss terms and steer the augmentor more towards disagreement or invariance respectively. To avoid balancing between the teacher's and the augmentor's training speed, we switch back and forth between optimizing the student and optimizing the augmentor based on the task loss $\mathcal{L}_{\tilde{s}}$ evaluated on the augmented inputs.

The Augmentor Models Similar to the Orbit method, it is also important for this approach to choose a suitable model for the augmentations. The model needs to be flexible enough to allow for challenging augmentations, but simple enough to optimize. To ensure this, we base all our augmentors on a distribution that is learned via the reparametrization trick [Kingma and Welling, 2014].
In the simplest case – *HARD-Affine* – this is a distribution over the entries of an affine transformation matrix $\vartheta \in \mathbb{R}^{2\times 3}$. This allows for affine transformations of the coordinate grid of pixel locations [Jaderberg et al., 2015], i.e. shifts, rotations, scalings, and shears of images (Figure 4.8B).
HARD-Mix, an augmentor inspired by Mixup [Zhang et al., 2017] and Cutmix [Yun et al., 2019] augmentations, computes a patch-wise interpolation between images (Figure 4.8A). The weight of the interpolation is computed input-dependent by processing each image patch and computing a cross-attention [Vaswani et al., 2017] with a query vector sampled from a learned distribution. As it wouldn't make sense to expect an invariance to this kind of augmentation, no teacher-teacher loss is applied for this augmentor, and student and augmentor are optimized jointly.

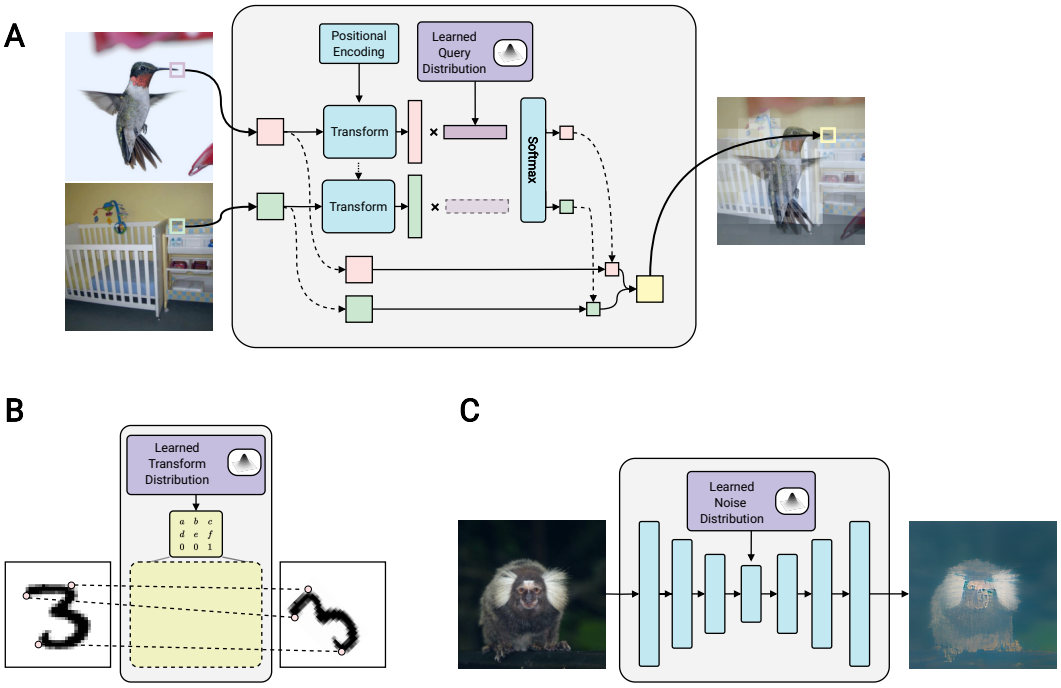


Figure 4.8: Illustration of the augmentor models used in our experiments. **(A)** HARD-Mix: Image-dependent patch-wise interpolation of multiple images. **(B)** HARD-Affine: Learned distribution of affine transformations in the pixel coordinates. **(C)** HARD-VAE: Finetuning (parts of) a pretrained VAE.

Finally, *HARD-VAE* is the augmentor with the least restrictions. It works by reconstructing an input image with a pre-trained Variational Autoencoder (VAE). The VAE's parameters are partially finetuned to modify the image during the reconstruction and additional noise from a learned distribution is added in the latent dimension. For the ImageNet experiments, this augmentor is based on the Residual-Quantized VAE (RQ-VAE) [Lee et al., 2022] model.

4.3.3 Results

Transferring Equivariance To confirm our hypothesis that selecting the right input data is the main hurdle to achieving transfer of generalization, we first reproduced the MNIST setup from Manuscript 3. With this, we test the transfer of shift equivariance using KD and the HARD-Affine augmentor. We observe improvements of +28.6% and +39.4% on the shifted test set compared to plain KD. Additionally, a qualitative analysis of the learned augmentations shows that the augmentor learns to shift the digits within the image. Comparing this result to Orbit transfer [Nix et al., 2022a], we see an improvement for the ViT student, but still, a large gap remains for the MLP student. However, the comparison is not totally fair, since HARD is way more efficient and flexible by only acting on the inputs as opposed to acting on every layer individually.

Table 4.3: MNIST (columns "Centered") and MNIST-C (columns "Shifted") test accuracies (mean and standard error of the mean across 4 random seeds). Compares KD without augmentation and our HARD-Affine method to Orbit transfer [Nix et al., 2022a], which also learns and transfers equivariances. The left two columns show the transfer results from a small CNN teacher to a MLP student. The right columns show analogous experiments between a ResNet18 teacher and a small ViT student. The best-performing transfer is shown in bold for each column. Examples of our HARD-Affine learned data augmentations are shown on the right. We include the controls *Random Affine* and *MNIST-C Shifts* (marked by italics).

	CNN → MLP		ResNet18 → ViT	
Method	Centered	Shifted	Centered	Shifted
Teacher only	99.0 ± 0.0	91.3 ± 0.5	99.5 ± 0.0	92.8 ± 0.5
Student only	98.4 ± 0.0	35.2 ± 0.7	98.3 ± 0.0	40.4 ± 0.8
+ Random Affine	*92.1 ± 0.6*	*81.0 ± 2.0*	*95.4 ± 0.3*	*90.4 ± 1.0*
+ MNIST-C Shifts	*98.1 ± 0.1*	*86.5 ± 0.3*	*98.5 ± 0.0*	*93.7 ± 0.2*
Orbit [Nix et al., 2022a]	**98.8**	**95.2**	98.4	**84.0**
KD	98.6 ± 0.0	40.3 ± 0.6	98.6 ± 0.1	44.7 ± 1.9
+ HARD🏋-Affine	98.6 ± 0.1	68.9 ± 2.5	**99.2** ± 0.0	**84.1** ± 2.3

Table 4.4: ID evaluation for ImageNet: reporting Top-1 accuracy in % on ImageNet-Validaton [Deng et al., 2009], ImageNet-ReaL [Beyer et al., 2020] and ImageNet-V2 [Recht et al., 2019] with KD from a robust ResNet50 [Hendrycks et al., 2021a] teacher to ResNet18 (columns 2-4) and ViT-S (columns 5-7) students.

	ResNet50 → ResNet18			ResNet50 → ViT-S		
	Val	ReaL	V2	Val	ReaL	V2
Teacher	75.8	83.1	63.7	75.8	83.1	63.7
Student	70.7	78.1	57.4	73.2	79.4	60.3
KD	70.7	78.7	58.1	75.3	82.8	62.9
+ HARD🏋-Affine	**71.6**	**79.5**	58.6	74.9	82.3	62.4
+ HARD🏋-Mix	71.4	79.4	58.6	75.7	83.0	63.3
+ HARD🏋-VAE	71.0	78.9	**58.7**	**75.8**	**83.1**	**63.5**

Table 4.5: OOD evaluation for ImageNet: Reporting Top-1 accuracy in % on ImageNet-A [Hendrycks et al., 2021b], ImageNet-R [Hendrycks et al., 2021a], ImageNet-Sketch [Wang et al., 2019] and ImageNet-Style [Geirhos et al., 2019] and mean-corruption-error on ImageNet-C (lower is better) [Hendrycks and Dietterich, 2019].

	ResNet50 → ResNet18					ResNet50 → ViT-S				
	Im-A	Im-R	Im-C ↓	Sketch	Style	Im-A	Im-R	Im-C ↓	Sketch	Style
Teacher	3.8	46.8	53.0	32.6	21.2	3.8	46.8	53.0	32.6	21.2
Student	1.6	30.0	88.1	18.4	4.4	**8.0**	26.3	78.1	13.8	6.6
KD	1.6	**40.2**	69.2	26.0	13.4	3.3	45.0	56.8	29.6	18.7
+ HARD🏋-Affine	1.5	38.2	73.1	24.9	10.4	3.4	40.8	62.2	26.2	14.5
+ HARD🏋-Mix	**1.8**	39.9	**68.8**	**26.1**	**13.7**	3.5	**45.4**	**56.2**	29.9	**19.2**
+ HARD🏋-VAE	1.7	39.5	72.5	25.8	12.1	3.4	**45.4**	57.4	**30.7**	18.1

Transfer on Natural Images After demonstrating the successful transfer of inductive biases in the form of shift equivariance, we moved the experiments toward mode realistic settings with natural images. Initial small-scale experiments on CIFAR10 [Krizhevsky et al.] showed success in the transfer of general test performance across architecture types, model sizes, and pre-training knowledge. We refer you to the manuscript for more details on these experiments.

For larger-scale experiments, we applied our approach to the ImageNet classification task [Deng et al., 2009]. We use a robustly trained ResNet50 [He et al., 2015, Hendrycks et al., 2021a] as the teacher for all experiments and ResNet18 and ViT-S [Dosovitskiy et al., 2020] as students to check the transfer across both architectures and model sizes. Following the example of [Oquab et al., 2023], we evaluate ID [Recht et al., 2019, Beyer et al., 2020] and OOD [Hendrycks and Dietterich, 2019, Hendrycks et al., 2021b,a, Geirhos et al., 2019, Wang et al., 2019] generalization. For a strong baseline, we added strong augmentations with Cutmix [Yun et al., 2019], Mixup [Zhang et al., 2017], and AugMix [Hendrycks et al., 2020] to the base KD setup. The standalone student we trained with lighter augmentations consisting of lower settings of CutMix, MixUp, and TrivialAugment [Müller and Hutter, 2021] instead of AugMix. More details on the setup can be found in Manuscript 4.

The results for the standalone student training show that the gap between teacher and student is much larger OOD than ID (average 4.4% ID, average 16.0% OOD). Standard KD closes the gap by a lot for ViT-S (+2.1% performance improvement compared to standalone on Val) on ID but barely makes a difference on the same test sets for the ResNet18 (+0% on Val). The same KD trained models evaluated on OOD show strong improvements for plain KD on OOD (up to 21.3% improvement on Im-C), which is unsurprising due to the strong augmentations we apply for this setup.

Our methods, HARD-Affine, HARD-Mix and HARD-VAE replace the heuristic augmentations of base KD setup with our augmentors that we optimize during the transfer. The approach shows strong performance improvements in the ID setting over the KD baseline of up to 0.9% for ResNet18 (HARD-Affine) and 0.6% for ViT-S (HARD-VAE). The HARD-VAE method even matches the teacher's performance on 2 out of 3 test sets for the ViT-S student. In the OOD setting, the HARD-Affine augmentor performs worse compared to the KD baseline, which is likely due to the

Figure 4.9: Example augmentations applied to images of the ImageNet validation set obtained from augmentor models in the ViT-S setting at the end of training.

lack of variety that affine augmentations offer in training compared to the AugMix augmentations of the baseline. HARD-VAE also doesn't reach the same levels for the ResNet18 student. However, HARD-Mix and HARD-VAE (for ViT-S) outperform plain KD on several test sets and are roughly on par for all others, across the board.

HARD additionally allows us to gain insight into KD and the transferred knowledge by visualizing the augmentations. Analyzing the augmentations from each augmentor model (Figure 4.9), we observe some interesting patterns. HARD-Affine seems to mainly focus on downscaling and shifting the image to crop away part of the input. The results from HARD-Mix show that the augmentor either merges two objects into the same picture (rows 1 and 4) or changes the style (row 2) or background (row 3). Finally, HARD-VAE seems to mainly change the style and brightness of the image, or perform some blurring of high-frequency features.

4.3.4 Discussion

The HARD framework demonstrates promise for improving knowledge distillation, though several important considerations emerge from our experiments. While our results show strong performance on moderate-sized models, scaling to larger teacher-student pairs presents both opportunities and challenges. The computational overhead of optimizing augmentors alongside student models may become significant with very large architectures, suggesting a need for more efficient optimization strategies.

Our results provide clear evidence that learned augmentations can outperform standard hand-crafted approaches, particularly for out-of-domain generalization. The interpretability of HARD offers valuable insights into which transformations best facilitate knowledge transfer, from simple affine transformations to complex style modifications. The varying effectiveness across augmentor types and architectural

pairs suggests that augmentation strategies should be carefully matched to specific distillation tasks.
Looking forward, diffusion models present an interesting direction for extending HARD's capabilities, potentially enabling more sophisticated augmentation strategies. However, this would need to be balanced against increased computational costs. Other promising directions include developing hybrid augmentation approaches that combine strengths of different augmentor types, and extending the framework beyond computer vision to establish HARD as a general-purpose distillation tool.

Discussion and Outlook 5

To highlight the contributions of this thesis and examine their connections or distinctions, we turn back to the four main aspects of functional transfer introduced in Chapter 2:

1. **Which systems are used as teacher and student?** Throughout this work, the student was generally chosen to be a DNN model that was optimized to align with the teacher's representation. For the teacher, we covered more variety. In Chapter 3, the teacher was a macaque brain, specifically its visual cortex that was probed through neurophysiological recordings. The works presented in Chapter 4 focused more on the other aspects of the transfer and thus chose DNNs as the teacher where we have total control over its inductive bias. Thus, regardless of which student was chosen, we always ensured a knowledge gap between teacher and student was present.

2. **What representation is used for the transfer?** The choice of representation is highly dependent on the system that is used for teacher and student. When considering brain recordings, we generally chose to represent them as spike counts. Additionally, it is important which brain areas were recorded. Here, we found that even an early area like V1 already contains generalization properties. Furthermore, our new attention readout architecture also enables higher brain areas to be used, as we successfully demonstrated by modeling the representations we recorded in macaque V4. When considering DNNs, the depth within the network can make a big difference, and we found that although it is beneficial to consider the representations on all layers, the output representation alone can already be sufficient for the transfer of generalization if the right data is presented.

3. **What data is used to generate the representations?** Choosing the right data as function input ended up being a crucial point in capturing generalization in representations. The experiments in Chapter 3 used a subset of images from the ImageNet training set to generate representations from both monkey neurons and DNN. However, we found that it was helpful to explore a larger range of inputs and we therefore generated synthetic representations through an intermediate model. Finally, both Orbit and HARD demonstrated the importance of exploring the input space. The results from those methods highlighted a fact that intuitively makes sense: Representations will behave arbitrarily in areas where the alignment did not happen. Thus, to ensure the transfer of generalization, you need to explore input space in these directions and match the representations there.

4. **How do we transfer the representations?** This aspect was the main focus throughout this thesis. In the first strand, we presented two new approaches for brain representations: neural co-training and the attention readout. In the second strand, we explored many different established methods and found that they are insufficient for transferring generalizing representations. Thus, we proposed two new methods: Orbit, a method that is very specialized to learning and transferring equivariance, and HARD, a framework of learning generalization directions in the input space that is orthogonal to most functional transfer methods. The methods demonstrate successful transfer of generalization for their respective application.

We addressed these questions in the pursuit of our primary research goals that we outlined at the beginning of this thesis (Section 2.3). Each goal is examined in detail in the corresponding sections that follow. We begin by discussing how this work contributes to improving generalization through functional transfer (Section 5.1), focusing on the identification and effective use of generalizing representations across tasks. Next, we explore advancements in functional transfer methods (Section 5.2), highlighting the robustness and efficiency achieved through our proposed approaches. Following this, we delve into the fundamental principles of generalization uncovered through this research (Section 5.3), offering new insights into its underlying mechanisms. Finally, we analyze the conditions and dynamics under which functional transfer succeeds or encounters limitations (Section 5.4), providing a comprehensive understanding of its potential and challenges. Together, these sections synthesize the theoretical and practical contributions of this work while situating them within the broader context of representation learning. However, several open questions remain and guide the path toward challenges that remain for future researchers to pursue (Section 5.5).

5.1 Generalization Through Functional Transfer

The main goal of this thesis was the improvement of generalization through functional transfer. This was approached through a variety of different avenues by the works presented in Manuscript 1, Manuscript 3 and Manuscript 4. The approach presented in Section 3.2 marks a proof-of-concept for neural co-training that demonstrates that the method can in principle improve generalization in a student using the representations of any generalizing teacher. Results achieved using recordings of macaque area V1 indicate the suitability of the brain as a teacher for this objective. However, the improvements are limited compared to SOTA methods like data augmentation and large-scale training. Nevertheless, this is a step towards instilling properties into the student that go beyond heuristics of data augmentations, and open up many possibilities for future exploration. Similarly, Orbit (Section 4.2) was not meant to deliver SOTA robustness, but instead should offer a deep insight into the mechanics of functional transfer of generalizing representations. The focus was on equivariance, which we believe should generalize to other forms of robustness, as the ideal network function should be invariant to the distribution shifts we are considering as OOD. Finally, HARD (Section 4.3) continues in the spirit of Orbit, but

makes it more generally applicable. We move from simple shift equivariance to OOD generalization on ImageNet. The results show that our method outperforms SOTA data augmentation techniques in combination with KD. It remains to be seen if the results we show with the ResNet50 teacher also generalize to larger robust models. If that is the case, this could open up possibilities to obtain efficient models that generalize well.

5.2 Improving Functional Transfer Methods

Each of the four manuscripts we include in this thesis introduced a novel transfer mechanism. Neural co-training (Section 3.2) and by extension its potential combination with the attention readout (Section 3.3) are designed to work with neuronal recordings, but can in principle be used to align any representation to those of a neural network. Therefore, our co-training method offers a flexible tool to transfer across arbitrarily distinct representations. Orbit's (Section 4.2) applicability on the other hand is limited due to its method needing a model of the equivariance for all layers of a network. However, HARD (Section 4.3) builds on the same principle as Orbit but only learns an invariance for the entire function in the input space. This seems to be limited to artificial neural networks at first glance since it needs to learn the augmentation model by backpropagating through the teacher. However, the experiments from Section 3.2 have shown that learning a model of the teacher representation as a prior step can be a valid alternative to using the teacher directly. If such a model is learned beforehand, the nature of the teacher is no longer relevant as we can simply use the "digital twin" we learned from it to train the augmentation model (inspired by Walker et al. [2019]). The resulting augmentations can even be applied in vivo to get real responses to the OOD stimuli through this. So far this application of the HARD framework is only theoretical and it remains to be shown that the digital twin is as reliable to find the generalization directions as the real brain.

5.3 Understanding Generalization

In addition to improving generalization, this work also offers a unique perspective into the mechanisms that impact generalization. Both Orbit and HARD learn the features the teacher is robust to. In the input space, these features are interpretable and allow us to identify what is transferred. We see emerging patterns in the augmentations that include shifts and rotations, texture changes, blurring, and obfuscation of objects. Most of these augmentations are generally expected to be useful for generalization, but the extent to which they are effective is rather surprising. In general, this analysis will provide a chance to understand the transfer for cases where we do not understand yet how the teacher generalizes. In this line of thought, we believe the analysis results from Section 3.2 were particularly interesting. We have not only found that for our co-trained models, neural prediction score and robustness are correlated, but also gained further hints towards the cause for this robustness improvement. Specifically, the analysis of the learned representations indicated that an emphasis on salient regions is a contributing factor for

generalization in macaque V1. This was achieved via the reconstruction analysis we proposed, which although not entirely novel [Feather et al., 2022], offers a unique perspective of the important features for a representation.
A follow-up analysis of our neurally co-trained model was done by Li et al. [2023]. They analyzed our model and compared it to the model from Li et al. [2019] which was regularized with mouse V1 data via RDL, as well as some baselines. Their findings suggest that models that have knowledge transferred from neuronal representations tend to be less sensitive to high-frequency components in the input. This is in line with other findings [Liu et al., 2024, Yin et al., 2019] and something that is also found for robust models trained with data augmentation and other methods to improve generalization. Overall, the insights from analyses conducted by us and Li et al. [2023] leave us with exciting directions for generalization research in both neuroscience and machine learning.

5.4 Understanding Functional Transfer

Beyond improving the transfer methods, an important goal in our work is to mechanisms understand functional transfer and explain its behavior. The most crucial factor in this is certainly to identify the knowledge that is transferred between teacher and student. Especially HARD attempts to help us understand this by explicitly learning the features in the input space that represent the gap of knowledge between teacher and student that the functional transfer is supposed to bridge. As we discussed in the previous section, the augmentations indicate some of the features that are being transferred which are the generalizing properties of the teacher.
We also investigated vanilla functional transfer methods. The most striking insight from this analysis was that they cannot guarantee to transfer an equivariance faithfully. The success of direct matching showed that internal representations are helpful for this problem, but the fact that RDL and attention transfer struggle with transfer indicates that the right amount of constraints is needed for the internal representations to be effective. One thing we hypothesized and confirmed with our experiments is that input is crucial for functional transfer. We need the inputs to cover a large range in order to transfer generalization knowledge. How large this range needs to be exactly is still an open question, as well as whether or not all generalization directions need to be covered in order for them to be transferred.

5.5 Remaining Challenges

We have shown multiple new methods for functional transfer, as well as plenty of insights into the nature of transfer and generalization. Nevertheless, there remains plenty to explore in all directions that we touched on in this thesis. In terms of functional transfer methods, the main challenge will be to scale them up towards bigger models, larger data, and higher brain areas. With the introduction of our attention readout, neural co-training is ready to be extended to capture more powerful representations. Similarly, HARD has the potential to enable the transfer of any imaginable type of generalization, however, a fitting augmentor model is needed

for this to succeed. Finding something that works flexibly and is powerful enough is a big challenge, where we believe generative models like diffusion [Dhariwal and Nichol, 2021] can be incredibly effective in generating augmented or entirely synthetic inputs. Furthermore, the combination of our two research stands remains open for future work. We already sketched a potential application of HARD using a digital twin model for transfer from neuronal recordings above and additionally combining this with neural co-training seems plausible as well.
Finally, with the increasing size and performance of today's machine learning models, one of the biggest challenges remains *alignment* [Soares and Fallenstein, 2014, Christian, 2021, Hendrycks et al., 2021c, Ji et al., 2023]. Alignment in AI has the goal to make AI systems behave in line with human intentions and values [Leike et al., 2018], and it can be broken down into four key objectives [Ji et al., 2023]: Robustness, Interpretability, Controllability, and Ethicality (RICE). Robustness is very much in line with the goals we pursued in this thesis, however, all four of those objectives are in some way related to what we want to achieve. Many recent efforts have attempted to achieve alignment through the utilization of human behavioral data [Ouyang et al., 2022, Rafailov et al., 2024]. Our efforts here are on a different scale and with a different modality of representations, but the training with behavioral data can be seen as a variant of functional transfer. Another indicator that highlights the importance of understanding functional transfer and its impact on generalizing representations.

5.6 Conclusion

In this thesis, we have explored key aspects of the functional transfer of generalizing representations, encompassing several critical dimensions. We conducted an in-depth analysis of existing functional transfer methods, developed improved methodologies for functional transfer, and investigated the potential for transfer from neuronal representations. Additionally, we examined the nature of generalization and the characteristics of generalizing representations.
The contributions of this work extend beyond the immediate scope of functional transfer, offering valuable perspectives and tools for advancing both neuroscience and machine learning. Through this interdisciplinary approach, we have made meaningful progress in understanding the mechanisms of generalization in biological systems and in enabling improved generalization in AI.
We hope that the methods and insights presented in this thesis will provide a robust foundation for future research at the intersection of neuroscience and machine learning, fostering further innovations and discoveries in these interconnected fields.

Bibliography

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International conference on machine learning*, pages 8821–8831. Pmlr, 2021.

Maryam Naqvi, Syed Qasim Gilani, Tehreem Syed, Oge Marques, and Hee-Cheol Kim. Skin cancer detection using deep learning—a review. *Diagnostics*, 13(11):1911, 2023.

Jonas Degrave, Federico Felici, Jonas Buchli, Michael Neunert, Brendan Tracey, Francesco Carpanese, Timo Ewalds, Roland Hafner, Abbas Abdolmaleki, Diego de Las Casas, et al. Magnetic control of tokamak plasmas through deep reinforcement learning. *Nature*, 602(7897):414–419, 2022.

Fabian H Sinz, Xaq Pitkow, Jacob Reimer, Matthias Bethge, and Andreas S Tolias. Engineering a less artificial intelligence. *Neuron*, 103(6):967–979, 2019.

Geoffrey Hinton and Jeff Dean. Distilling the Knowledge in a Neural Network. Technical report, 2015.

Patrick McClure and Nikolaus Kriegeskorte. Representational distance learning for deep neural networks. *Frontiers in Computational Neuroscience*, 10(DEC):131, dec 2016. ISSN 16625188. doi: 10.3389/fncom.2016.00131. URL `http://journal.frontiersin.org/article/10.3389/fncom.2016.00131/full`.

Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*, 2017. URL `https://github.com/szagoruyko/attention-transfer`.

Liz Allen, Jo Scott, Amy Brand, Marjorie Hlava, and Micah Altman. Publishing: Credit where credit is due. *Nature*, 508(7496):312–313, April 2014.

Liz Allen, Alison O'Connell, and Veronique Kiermer. How can we ensure visibility and diversity in research contributions? how the contributor role taxonomy (credit) is helping the shift from authorship to contributorship. *Learned Publishing*, 32(1): 71–74, 2019. doi: https://doi.org/10.1002/leap.1210. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/leap.1210`.

Alex O. Holcombe. Contributorship, not authorship: Use credit to indicate who did what. *Publications*, 7(3), 2019. ISSN 2304-6775. doi: 10.3390/publications7030048. URL `https://www.mdpi.com/2304-6775/7/3/48`.

Mohammad Bashiri. Latent variable and implicit models for neural system identification, March 2024.

Shahd Safarani, Arne Nix, Konstantin Willeke, Santiago Cadena, Kelli Restivo, George Denfield, Andreas Tolias, and Fabian Sinz. Towards robust vision by multi-task learning on monkey visual cortex. *Advances in Neural Information Processing Systems*, 34:739–751, 2021.

Paweł A. Pierzchlewicz, Konstantin Friedrich Willeke, Arne Nix, Pavithra Elumalai, Kelli Restivo, Tori Shinn, Cate Nealley, Gabrielle Rodriguez, Saumil Patel, Katrin Franke, Andreas S. Tolias, and Fabian H. Sinz. Energy guided diffusion for generating neurally exciting images. *Advances in Neural Information Processing Systems*, 36, 2024.

Arne Nix, Suhas Shrinivasan, Edgar Y Walker, and Fabian Sinz. Can Functional Transfer Methods Capture Simple Inductive Biases? In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 10703–10717. PMLR, 2022a.

Arne F Nix, Max F Burg, and Fabian H Sinz. Hard: Hard augmentations for robust distillation. *arXiv preprint arXiv:2305.14890*, 2023.

Konstantin F Willeke, Kelli Restivo, Katrin Franke, Arne F Nix, Santiago A Cadena, Tori Shinn, Cate Nealley, Gabby Rodriguez, Saumil Patel, Alexander S Ecker, Fabian H Sinz, and Andreas S Tolias. Deep learning-driven characterization of single cell tuning in primate visual area V4 unveils topological organization. May 2023.

Arne Nix, Max F Burg, and Fabian H Sinz. Leading by example: Guiding knowledge transfer with adversarial data augmentation. In *NeurIPS 2022 Workshop on Synthetic Data for Empowering ML Research*, 2022b.

Ben Baker, Benjamin Lansdell, and Konrad P Kording. Three aspects of representation in neuroscience. *Trends in cognitive sciences*, 26(11):942–958, 2022.

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

Constantin Teleman. Representation theory. *math.berkeley.edu*, 2005. URL `https://math.berkeley.edu/~teleman/math/RepThry.pdf`.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25, pages 1097–1105. Curran Associates, Inc., 2012. URL `https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf`.

Tomas Mikolov. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 3781, 2013.

Talia Konkle and George A Alvarez. A self-supervised domain-general learning framework for human ventral stream representation. *Nature communications*, 13(1): 491, 2022.

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(11):665–673, 2020a.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*, 2019.

Dan Hendrycks, Andy Zou, Mantas Mazeika, Leonard Tang, Bo Li, Dawn Song, and Jacob Steinhardt. Pixmix: Dreamlike pictures comprehensively improve safety measures. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16783–16792, 2022.

Claudio Michaelis, Benjamin Mitzkus, Robert Geirhos, Evgenia Rusak, Oliver Bringmanny, Alexander S. Eckery, Matthias Bethgey, and Wieland Brendely. Benchmarking Robustness in Object Detection: Autonomous driving when winter is coming, 2019. ISSN 23318422. URL `https://github.com/bethgelab/`.

David H Wolpert and William G Macready. No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, 1(1):67–82, 1997.

Eyke Hüllermeier, Thomas Fober, and Marco Mernberger. Inductive bias. *Encyclopedia of systems biology*, pages 1018–1019, 2013.

Tom M Mitchell. The need for biases in learning generalizations. 1980.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*. International Conference on Learning Representations, ICLR, dec 2014. URL `https://arxiv.org/abs/1312.6199v4`.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Gamaleldin Elsayed, Shreya Shankar, Brian Cheung, Nicolas Papernot, Alexey Kurakin, Ian Goodfellow, and Jascha Sohl-Dickstein. Adversarial examples that fool both computer vision and time-limited humans. *Advances in neural information processing systems*, 31, 2018.

Chong Guo, Michael J Lee, Guillaume Leclerc, Joel Dapello, Yug Rao, Aleksander Madry, and James J DiCarlo. Adversarially trained neural representations may already be as robust as corresponding biological neural representations. *arXiv preprint arXiv:2206.11228*, 2022.

Vijay Veerabadran, Josh Goldman, Shreya Shankar, Brian Cheung, Nicolas Papernot, Alexey Kurakin, Ian Goodfellow, Jonathon Shlens, Jascha Sohl-Dickstein, Michael C Mozer, et al. Subtle adversarial image manipulations influence both human and machine perception. *Nature Communications*, 14(1):4933, 2023.

Guy Gaziv, Michael Lee, and James J DiCarlo. Strong and precise modulation of human percepts via robustified anns. *Advances in Neural Information Processing Systems*, 36, 2024.

Robert Geirhos, Carlos R. Medina Temme, Jonas Rauber, Heiko H. Schütt, Matthias Bethge, and Felix A. Wichmann. Generalisation in humans and deep neural networks. *Advances in Neural Information Processing Systems*, 2018-Decem: 7538–7550, aug 2018. URL `http://arxiv.org/abs/1808.08750`.

Robert Geirhos, Kantharaju Narayanappa, Benjamin Mitzkus, Tizian Thieringer, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. Partial success in closing the gap between human and machine vision. *Advances in Neural Information Processing Systems*, 34:23885–23899, 2021.

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.

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.

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*, 33:13890–13902, 2020b.

Girik Malik, Dakarai Crowder, and Ennio Mingolla. Extreme image transformations affect humans and machines differently. *Biological Cybernetics*, 117:331 – 343, 2022. URL `https://api.semanticscholar.org/CorpusID:255186202`.

Irving Biederman, Robert J Mezzanotte, and Jan C Rabinowitz. Scene perception: Detecting and judging objects undergoing relational violations. *Cognitive psychology*, 14(2):143–177, 1982.

Irving Biederman. On the semantics of a glance at a scene. In *Perceptual organization*, pages 213–253. Routledge, 1981.

Aude Oliva and Antonio Torralba. The role of context in object recognition. *Trends in cognitive sciences*, 11(12):520–527, 2007.

Monica S Castelhano and Chelsea Heaven. Scene context influences without scene gist: Eye movements guided by spatial associations in visual search. *Psychonomic bulletin & review*, 18:890–896, 2011.

Joshua B Tenenbaum, Charles Kemp, Thomas L Griffiths, and Noah D Goodman. How to grow a mind: Statistics, structure, and abstraction. *science*, 331(6022):1279–1285, 2011.

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*, May 2019. URL `https://openreview.net/forum?id=Bygh9j09KX`.

Ajay Subramanian, Elena Sizikova, Najib Majaj, and Denis Pelli. Spatial-frequency channels, shape bias, and adversarial robustness. *Advances in Neural Information Processing Systems*, 36, 2024.

Guangyao Chen, Peixi Peng, Li Ma, Jia Li, Lin Du, and Yonghong Tian. Amplitude-phase recombination: Rethinking robustness of convolutional neural networks in frequency domain. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 458–467, 2021.

Zhe Li, Josue Ortega Caro, Evgenia Rusak, Wieland Brendel, Matthias Bethge, Fabio Anselmi, Ankit B Patel, Andreas S Tolias, and Xaq Pitkow. Robust deep learning object recognition models rely on low frequency information in natural images. *PLOS Computational Biology*, 19(3):e1010932, 2023.

Qingwen Bu, Dong Huang, and Heming Cui. Towards building more robust models with frequency bias. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4402–4411, 2023.

Bruno A Olshausen. 20 years of learning about vision: Questions answered, questions unanswered, and questions not yet asked. In *20 Years of Computational Neuroscience*, pages 243–270. Springer, 2013.

Tim C. Kietzmann, Courtney J. Spoerer, Lynn K. A. Sörensen, Radoslaw M. Cichy, Olaf Hauk, and Nikolaus 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. ISSN 0027-8424. doi: 10.1073/pnas.1905544116. URL `https://www.pnas.org/content/116/43/21854`.

Nathan Drenkow, Numair Sani, Ilya Shpitser, and Mathias Unberath. A systematic review of robustness in deep learning for computer vision: Mind the gap? *arXiv preprint arXiv:2112.00639*, 2021.

Kunihiko Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4):193–202, 1980. ISSN 03401200. doi: 10.1007/BF00344251.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11): 2278–2324, 1998.

Andrei N Tikhonov. Solution of incorrectly formulated problems and the regularization method. *Sov Dok*, 4:1035–1038, 1963.

Chen-Yu Lee, Saining Xie, Patrick Gallagher, Zhengyou Zhang, and Zhuowen Tu. Deeply-Supervised Nets. In Guy Lebanon and S. V. N. Vishwanathan, editors, *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, volume 38 of *Proceedings of Machine Learning Research*, pages 562–570, San Diego, California, USA, 09–12 May 2015. PMLR. URL `https://proceedings.mlr.press/v38/lee15a.html`.

Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. AutoAugment: Learning Augmentation Policies from Data. *Cvpr 2019*, (Section 3): 113–123, may 2018. doi: 10.48550/arxiv.1805.09501. URL `https://arxiv.org/abs/1805.09501v3`.

Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond Empirical Risk Minimization. *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, oct 2017. doi: 10.48550/arxiv.1710.09412. URL `https://arxiv.org/abs/1710.09412v2`.

Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6023–6032, 2019.

Daniel Ho, Eric Liang, Xi Chen, Ion Stoica, and Pieter Abbeel. Population based augmentation: Efficient learning of augmentation policy schedules. In *International conference on machine learning*, pages 2731–2741. PMLR, 2019.

Keyu Tian, Chen Lin, Ming Sun, Luping Zhou, Junjie Yan, and Wanli Ouyang. Improving auto-augment via augmentation-wise weight sharing. *Advances in Neural Information Processing Systems*, 33:19088–19098, 2020.

Tom Bekor, Niv Nayman, and Lihi Zelnik-Manor. Freeaugment: Data augmentation search across all degrees of freedom. *arXiv preprint arXiv:2409.04820*, 2024.

Samuel G Müller and Frank Hutter. Trivialaugment: Tuning-free yet state-of-the-art data augmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 774–782, 2021.

Dan Hendrycks, Norman Mu, Ekin Dogus Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. In *International Conference on Learning Representations*, 2020.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The Many Faces of Robustness: A Critical Analysis of Out-of-Distribution Generalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8320–8329, 2021a. ISBN 9781665428125. doi:

10.1109/ICCV48922.2021.00823. URL https://github.com/hendrycks/imagenet-r.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=rJzIBfZAb.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, Victor Lempitsky, Urun Dogan, Marius Kloft, Francesco Orabona, Tatiana Tommasi, and Al Ganin. Domain-Adversarial Training of Neural Networks. Technical report, 2016.

Evgenia Rusak, Lukas Schott, Roland S. Zimmermann, Julian Bitterwolf, Oliver Bringmann, Matthias Bethge, and Wieland Brendel. A simple way to make neural networks robust against diverse image corruptions. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12348 LNCS:53–69, jan 2020. ISSN 16113349. doi: 10.48550/arxiv.2001.06057. URL https://arxiv.org/abs/2001.06057v5.

Tauhidul Islam, Md Sadman Hafiz, Jamin Rahman Jim, Md Mohsin Kabir, and MF Mridha. A systematic review of deep learning data augmentation in medical imaging: Recent advances and future research directions. *Healthcare Analytics*, page 100340, 2024.

Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens Van Der Maaten. Exploring the limits of weakly supervised pretraining. In *Proceedings of the European conference on computer vision (ECCV)*, pages 181–196, 2018.

A Emin Orhan. Robustness properties of facebook's resnext wsl models. *arXiv preprint arXiv:1907.07640*, 2019.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023.

A Emin Orhan. Scaling may be all you need for achieving human-level object recognition capacity with human-like visual experience. *arXiv preprint arXiv:2308.03712*, 2023.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

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*, 2020c.

Jiaming Ji, Tianyi Qiu, Boyuan Chen, Borong Zhang, Hantao Lou, Kaile Wang, Yawen Duan, Zhonghao He, Jiayi Zhou, Zhaowei Zhang, et al. Ai alignment: A comprehensive survey. *arXiv preprint arXiv:2310.19852*, 2023.

Chang Liu, Yinpeng Dong, Wenzhao Xiang, Xiao Yang, Hang Su, Jun Zhu, Yuefeng Chen, Yuan He, Hui Xue, and Shibao Zheng. A comprehensive study on robustness of image classification models: Benchmarking and rethinking. *International Journal of Computer Vision*, pages 1–23, 2024.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, June 2009.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*, pages 5389–5400. PMLR, 2019.

Lucas Beyer, Olivier J Hénaff, Alexander Kolesnikov, Xiaohua Zhai, and Aäron van den Oord. Are we done with imagenet? *arXiv preprint arXiv:2006.07159*, 2020.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. *CVPR*, 2021b.

Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pages 10506–10518, 2019.

Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2(11):e7, November 2017. doi: 10.23915/distill.00007. https://distill.pub/2017/feature-visualization.

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.

Jenelle Feather, Guillaume Leclerc, Aleksander Mądry, and Josh H McDermott. Model metamers reveal divergent invariances between biological and artificial neural networks. *Nature Neuroscience*, 26(11):2017–2034, 2023.

Takahiro Itazuri, Yoshihiro Fukuhara, Hirokatsu Kataoka, and Shigeo Morishima. What do adversarially robust models look at? *arXiv preprint arXiv:1905.07666*, 2019.

Dong Yin, Raphael Gontijo Lopes, Jon Shlens, Ekin Dogus Cubuk, and Justin Gilmer. A fourier perspective on model robustness in computer vision. *Advances in Neural Information Processing Systems*, 32, 2019.

Lorien Y Pratt. Discriminability-based transfer between neural networks. *Advances in neural information processing systems*, 5, 1992.

Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. A survey on deep transfer learning. In *Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part III 27*, pages 270–279. Springer, 2018.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. arxiv e-prints, art. *arXiv preprint arXiv:1911.05722*, 2019.

Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 647–655, Bejing, China, 22–24 Jun 2014. PMLR. URL `https://proceedings.mlr.press/v32/donahue14.html`.

Jacob Devlin. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020. URL `https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf`.

Lucas Beyer, Xiaohua Zhai, Amélie Royer, Larisa Markeeva, Rohan Anil, and Alexander Kolesnikov. Knowledge distillation: A good teacher is patient and consistent. jun 2021. URL `https://arxiv.org/abs/2106.05237v1http://arxiv.org/abs/2106.05237`.

Ilia Sucholutsky, Lukas Muttenthaler, Adrian Weller, Andi Peng, Andreea Bobu, Been Kim, Bradley C Love, Erin Grant, Iris Groen, Jascha Achterberg, et al. Getting aligned on representational alignment. *arXiv preprint arXiv:2310.13018*, 2023.

Lukas Muttenthaler, Lorenz Linhardt, Jonas Dippel, Robert A Vandermeulen, Katherine Hermann, Andrew Lampinen, and Simon Kornblith. Improving neural network representations using human similarity judgments. *Advances in Neural Information Processing Systems*, 36, 2024.

Zhe Li, Wieland Brendel, Edgar Walker, Erick Cobos, Taliah Muhammad, Jacob Reimer, Matthias Bethge, Fabian Sinz, Zachary Pitkow, and Andreas Tolias. Learning from brains how to regularize machines. In *Advances in Neural Information Processing Systems*, pages 9529–9539, 2019.

David H Hubel and Torsten N Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1): 106, 1962.

Howard Akler. David hubel: The scientist. *Defining Moments Canada: NobelCanadian*, 2022. URL `definingmomentscanada.ca/nobelcanadian/david-hubel/the-scientist/`.

Konstantin-Klemens Lurz, Mohammad Bashiri, Konstantin Willeke, Akshay Jagadish, Eric Wang, Edgar Y. Walker, Santiago A Cadena, Taliah Muhammad, Erick Cobos, Andreas S. Tolias, Alexander S Ecker, and Fabian H. Sinz. Generalization in data-driven models of primary visual cortex. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=Tp7kI90Htd`.

David Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information.* Henry Holt and Co., Inc., New York, NY, USA, 1982. ISBN 0716715678.

Demis Hassabis, Dharshan Kumaran, Christopher Summerfield, and Matthew Botvinick. Neuroscience-inspired artificial intelligence. *Neuron*, 95(2):245–258, 2017.

R Becket Ebitz and Benjamin Y Hayden. The population doctrine in cognitive neuroscience. *Neuron*, 109(19):3055–3068, 2021.

Jonathan W Pillow and Mikio C Aoi. Is population activity more than the sum of its parts? *Nature neuroscience*, 20(9):1196–1198, 2017.

John C Gore et al. Principles and practice of functional mri of the human brain. *The Journal of clinical investigation*, 112(1):4–9, 2003.

Alan L Hodgkin and Andrew F Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of physiology*, 117(4):500, 1952.

James J Jun, Nicholas A Steinmetz, Joshua H Siegle, Daniel J Denman, Marius Bauza, Brian Barbarits, Albert K Lee, Costas A Anastassiou, Alexandru Andrei, Çağatay Aydın, et al. Fully integrated silicon probes for high-density recording of neural activity. *Nature*, 551(7679):232–236, 2017.

Fritjof Helmchen. Two-photon functional imaging of neuronal activity. *In vivo optical imaging of brain function*, page 428, 2009.

Nicholas James Sofroniew, Daniel Flickinger, Jonathan King, and Karel Svoboda. A large field of view two-photon mesoscope with subcellular resolution for in vivo imaging. *elife*, 5:e14472, 2016.

Charles F Cadieu, Ha Hong, Daniel L K Yamins, Nicolas Pinto, Diego Ardila, Ethan A Solomon, Najib J Majaj, and James J DiCarlo. Deep neural networks rival the representation of primate IT cortex for core visual object recognition. *PLoS Comput. Biol.*, 10(12):e1003963, December 2014.

Daniel L K 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. *Proc. Natl. Acad. Sci. U. S. A.*, 111(23):8619–8624, June 2014.

Santiago A Cadena, George H Denfield, Edgar Y Walker, Leon A Gatys, Andreas S Tolias, Matthias Bethge, and Alexander S Ecker. Deep convolutional models improve predictions of macaque V1 responses to natural images. *PLoS Comput. Biol.*, 15(4):e1006897, April 2019.

Fabian H Sinz, Alexander S Ecker, Paul G Fahey, Edgar Y Walker, Erick Cobos, Emmanouil Froudarakis, Dimitri Yatsenko, Xaq Pitkow, Jacob Reimer, and Andreas S Tolias. Stimulus domain transfer in recurrent models for large scale cortical population prediction on video. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, pages 7199–7210, Red Hook, NY, USA, December 2018. Curran Associates Inc.

Xiaofeng Zhang, Zhangyang Wang, Dong Liu, and Qing Ling. DADA: Deep Adversarial Data Augmentation for Extremely Low Data Regime Classification. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2019-May:2807–2811, may 2019. ISSN 15206149. doi: 10.1109/ICASSP.2019.8683197. URL `https://arxiv.org/abs/1809.00981v1`.

Lane T McIntosh, Niru Maheswaranathan, Aran Nayebi, Surya Ganguli, and Stephen A Baccus. Deep learning models of the retinal response to natural scenes. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pages 1369–1377, Red Hook, NY, USA, December 2016. Curran Associates Inc.

D A Klindt, A S Ecker, T Euler, and M Bethge. Neural system identification for large populations separating "what" and "where". In *Advances in Neural Information Processing Systems*, pages 4–6, 2017.

William F Kindel, Elijah D Christensen, and Joel Zylberberg. Using deep learning to probe the neural code for images in primary visual cortex. *J. Vis.*, 19(4):29, April 2019.

Alexander S Ecker, Fabian H Sinz, Emmanouil Froudarakis, Paul G Fahey, Santiago A Cadena, Edgar Y Walker, Erick Cobos, Jacob Reimer, Andreas S Tolias, and Matthias Bethge. A rotation-equivariant convolutional neural network model of primary visual cortex. September 2018.

Benjamin R Cowley and Jonathan W Pillow. High-contrast "gaudy" images improve the training of deep neural network models of visual cortex. June 2020.

Max F Burg, Santiago A Cadena, George H Denfield, Edgar Y Walker, Andreas S Tolias, Matthias Bethge, and Alexander S Ecker. Learning divisive normalization in primary visual cortex. *PLoS Comput. Biol.*, 17(6):e1009028, June 2021.

Eleanor Batty, Josh Merel, Nora Brackbill, Alexander Heitman, Alexander Sher, Alan Litke, E J Chichilnisky, and Liam Paninski. Multilayer recurrent network models of primate retinal ganglion cell responses. July 2022.

Mohammad Bashiri, Edgar Walker, Konstantin-Klemens Lurz, Akshay Jagadish, Taliah Muhammad, Zhiwei Ding, Zhuokun Ding, Andreas Tolias, and Fabian Sinz. A flow-based latent state generative model of neural population responses to natural images. *Adv. Neural Inf. Process. Syst.*, 34:15801–15815, December 2021.

Ján Antolík, Sonja B Hofer, James A Bednar, and Thomas D Mrsic-Flogel. Model constrained by visual hierarchy improves prediction of neural responses to natural scenes. *PLoS Comput. Biol.*, 12(6):e1004927, June 2016.

U. Guclu and M. A. J. van Gerven. Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. *Journal of Neuroscience*, 35(27):10005–10014, July 2015. doi: 10.1523/jneurosci.5023-14.2015. URL `https://doi.org/10.1523/jneurosci.5023-14.2015`.

Daniel L K Yamins and James J DiCarlo. Using goal-driven deep learning models to understand sensory cortex. *Nat. Neurosci.*, 19(3):356–365, March 2016.

Santiago A Cadena, Konstantin F Willeke, Kelli Restivo, George Denfield, Fabian H Sinz, Matthias Bethge, Andreas S Tolias, and Alexander S Ecker. Diverse task-driven modeling of macaque V4 reveals functional specialization towards semantic tasks. May 2022.

Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, 2010.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*. International Conference on Learning Representations, ICLR, dec 2014. doi: 10.48550/arxiv.1312.6114. URL `https://arxiv.org/abs/1312.6114v10`.

Li Zhaoping and Zhaoping Li. *Understanding vision: theory, models, and data*. Oxford University Press, USA, 2014.

Ruth C. Fong, Walter J. Scheirer, and David D. Cox. Using human brain activity to guide machine learning. *Scientific Reports*, 8(1):5397, Mar 2018. doi: 10.1038/s41598-018-23618-6. URL `https://doi.org/10.1038/s41598-018-23618-6`.

Callie Federer, Haoyan Xu, Alona Fyshe, and Joel Zylberberg. Improved object recognition using neural networks trained to mimic the brain's statistical properties. *Neural Networks*, 131:103–114, 2020. ISSN 18792782. doi: 10.1016/j.neunet.2020.07.013.

Richard A. Caruana. Multitask Learning: A Knowledge-Based Source of Inductive Bias. In *Machine Learning Proceedings 1993*. 1993. doi: 10.1016/b978-1-55860-307-3.50012-5.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491, 2018.

Joel Dapello, Tiago Marques, Martin Schrimpf, Franziska Geiger, David Cox, and James J DiCarlo. Simulating a primary visual cortex at the front of CNNs improves robustness to image perturbations. In *Advances in Neural Information Processing Systems (NeurIPS 2020)*, volume 33, pages 13073–13087. Curran Associates, Inc., 2020.

Matthias Kummerer, Thomas S. A. Wallis, Leon A. Gatys, and Matthias Bethge. Understanding low- and high-level contributions to fixation prediction. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

Igor Vasiljevic, Ayan Chakrabarti, and Gregory Shakhnarovich. Examining the impact of blur on recognition by convolutional networks. *arXiv preprint arXiv:1611.05760*, 2016.

Zhaoping Li. A saliency map in primary visual cortex. *Trends in cognitive sciences*, 6 (1):9–16, 2002.

Li Zhaoping and Li Zhe. Primary visual cortex as a saliency map: a parameter-free prediction and its test by behavioral data. *PLoS Comput Biol*, 11(10):e1004375, 2015.

Xilin Zhang, Li Zhaoping, Tiangang Zhou, and Fang Fang. Neural activities in v1 create a bottom-up saliency map. *Neuron*, 73(1):183–192, 2012.

Nobuhiko Wagatsuma, Akinori Hidaka, and Hiroshi Tamura. Correspondence between monkey visual cortices and layers of a saliency map model based on a deep convolutional neural network for representations of natural images. *Eneuro*, 8 (1), 2021.

A S Tolias, T Moore, S M Smirnakis, E J Tehovnik, A G Siapas, and P H Schiller. Eye movements modulate visual receptive fields of V4 neurons. *Neuron*, 29(3):757–767, March 2001.

Alex Graves. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*, 2013.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 2017-Decem, pages 5999–6009, 2017.

Jingjing Xu, Xu Sun, Zhiyuan Zhang, Guangxiang Zhao, and Junyang Lin. Understanding and improving layer normalization. *Advances in Neural Information Processing Systems*, 32, 2019.

Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 2015.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-Decem:770–778, dec 2015. ISSN 10636919. doi: 10.1109/CVPR.2016.90. URL `https://arxiv.org/abs/1512.03385v1`.

Hadi Salman, Andrew Ilyas, Logan Engstrom, Ashish Kapoor, and Aleksander Madry. Do adversarially robust ImageNet models transfer better? July 2020.

Richard P Feynman. Symmetry in physical laws. *The Physics Teacher*, 4(4):161–174, 1966.

Jonathan Gordon, David Lopez-Paz, Marco Baroni, and Diane Bouchacourt. Permutation equivariant models for compositional generalization in language. *Iclr*, (2019):1–12, sep 2020.

Erik Henning Thiede, Truong Son Hy, and Risi Kondor. The general theory of permutation equivarant neural networks and higher order graph variational encoders. 2020. URL `http://arxiv.org/abs/2004.03990`.

Taco S. Cohen. *Equivariant convolutional networks*. PhD thesis, University of Amsterdam, 2021. URL `https://dare.uva.nl`.

Ari S Benjamin, David Rolnick, and Konrad P Kording. Measuring and regularizing networks in function space. In *7th International Conference on Learning Representations, ICLR 2019*, 2019.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of Neural Network Representations Revisited. *36th International Conference on Machine Learning, ICML 2019*, 2019-June:6156–6175, may 2019. URL https://arxiv.org/abs/1905.00414v4.

Cristian Bucilă, Rich Caruana, and Alexandra Niculescu-Mizil. Model compression. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, volume 2006, pages 535–541, 2006. ISBN 1595933395. doi: 10.1145/1150402.1150464.

Pingbo Pan, Siddharth Swaroop, Alexander Immer, Runa Eschenhagen, Richard E Turner, and Mohammad Emtiyaz Khan. Continual deep learning by functional regularisation of memorable past. In *Advances in Neural Information Processing Systems*, volume 2020-Decem, 2020.

Michalis K Titsias, Jonathan Schwarz, Alexander G. de G. Matthews, Razvan Pascanu, and Yee Whye Teh. Functional Regularisation for Continual Learning with Gaussian Processes. 2019. URL http://arxiv.org/abs/1901.11356.

Samira Abnar, Mostafa Dehghani, and Willem Zuidema. Transferring Inductive Biases through Knowledge Distillation. 2020. URL https://github.com/samiraabnar/Reflect.http://arxiv.org/abs/2006.00555.

Sam Greydanus. Scaling down Deep Learning. nov 2020. URL https://arxiv.org/abs/2011.14439v3http://arxiv.org/abs/2011.14439.

Tomas B. Co. Matrix Analysis. In *Methods of Applied Mathematics for Engineers and Scientists*, pages 99–146. Cambridge University Press, oct 2013. ISBN 9781139020411. doi: 10.1017/cbo9781139021821.005.

Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *Advances in Neural Information Processing Systems*, volume 2015-Janua, pages 2017–2025. Neural information processing systems foundation, jun 2015. URL https://arxiv.org/abs/1506.02025v3.

Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. 2020. URL https://github.com/http://arxiv.org/abs/2010.11929.

Norman Mu and Justin Gilmer. MNIST-C: A Robustness Benchmark for Computer Vision. jun 2019. doi: 10.5281/zenodo.3237938. URL https://arxiv.org/abs/1906.02337v1http://arxiv.org/abs/1906.02337.

Taco S. Cohen and Max Welling. Group Equivariant Convolutional Networks. *33rd International Conference on Machine Learning, ICML 2016*, 6:4375–4386, feb 2016. URL `https://arxiv.org/abs/1602.07576v3`.

Doyup Lee, Chiheon Kim, Saehoon Kim, Minsu Cho, and Wook-Shin Han. Autoregressive image generation using residual quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11523–11532, 2022.

Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. CIFAR-10 (Canadian Institute for Advanced Research). URL `http://www.cs.toronto.edu/$\sim$kriz/cifar.html`.

Edgar Y Walker, Fabian H Sinz, Erick Cobos, Taliah Muhammad, Emmanouil Froudarakis, Paul G Fahey, Alexander S Ecker, Jacob Reimer, Xaq Pitkow, and Andreas S Tolias. Inception loops discover what excites neurons most using deep predictive models. *Nat. Neurosci.*, 22(12):2060–2065, December 2019.

Jenelle Feather, Guillaume Leclerc, Aleksander Mądry, and Josh H McDermott. Model metamers illuminate divergences between biological and artificial neural networks. May 2022.

Prafulla Dhariwal and Alex Nichol. Diffusion models beat GANs on image synthesis. May 2021.

Nate Soares and Benja Fallenstein. Aligning superintelligence with human interests: A technical research agenda. *Machine Intelligence Research Institute (MIRI) technical report*, 8, 2014.

Brian Christian. The alignment problem: Machine learning and human values. *Perspectives on Science and Christian Faith*, 73:245–247, 12 2021. doi: 10.56315/PSCF12-21Christian.

Dan Hendrycks, Nicholas Carlini, John Schulman, and Jacob Steinhardt. Unsolved problems in ml safety. *arXiv preprint arXiv:2109.13916*, 2021c.

Jan Leike, David Krueger, Tom Everitt, Miljan Martic, Vishal Maini, and Shane Legg. Scalable agent alignment via reward modeling: a research direction. *arXiv preprint arXiv:1811.07871*, 2018.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36, 2024.

Appendix

Acronyms

AI Artificial Intelligence. 9, 19, 61

CNN Convolutional Neural Network. 24, 25, 36, 41, 45, 46, 52

DNN Deep Neural Network. , 10, 14, 17, 18, 23, 25, 57

fMRI functional Magnetic Resonance Imaging. 21, 23

G-CNN Group-Convolutional Neural Network. 46, 47

HARD HARD Augmentations for Robust Distillation. 49, 58

ID in-distribution. 14–16, 19, 48, 52, 53

KD Knowledge Distillation. 37, 41, 42, 48, 50–54, 59

ML Machine Learning. 9, 21

MLP Multilayer Perceptron. 45–49, 51, 52

MSE Mean-Squared Error. 42, 49

MTL Multi-task Learning. 25

OOD out-of-distribution. 15, 16, 18, 19, 25, 48, 53, 58, 59

RDL Representational Distance Learning. 38, 42, 43

RDM Representational Dissimilarity Matrix. 38

SOTA state-of-the-art. 30, 58, 59

TIN Tiny ImageNet. 26

VAE Variational Autoencoder. 51

ViT Vision Transformer. 45, 46, 51–54

Manuscripts

This section contains the publications discussed in chapters 2 & 3.

Towards robust vision by multi-task learning on monkey visual cortex

Shahd Safarani,[1,*] Arne Nix,[1-2] Konstantin Willeke,[1-2] Santiago A. Cadena,[2-3]
Kelli Restivo,[4-5] George Denfield,[6] Andreas S. Tolias,[4-5] Fabian H. Sinz[1-5,]**
[1] Institute for Bioinformatics and Medical Informatics, University Tübingen, Germany
[2] International Max Planck Research School for Intelligent Systems, Tübingen, Germany
[3] Institute for Computer Science, University of Göttingen, Germany
[4] Department for Neuroscience, Baylor College of Medicine, Houston, TX, USA
[5] Center for Neuroscience and Artificial Intelligence, Baylor College of Medicine, Houston, TX, USA
[6] Columbia University, Department of Psychiatry, New York, USA

[*]shahdsaf@hotmail.com, [**]sinz@cs.uni-goettingen.de

Abstract

Deep neural networks set the state-of-the-art across many tasks in computer vision, but their generalization ability to simple image distortions is surprisingly fragile. In contrast, the mammalian visual system is robust to a wide range of perturbations. Recent work suggests that this generalization ability can be explained by useful inductive biases encoded in the representations of visual stimuli throughout the visual cortex. Here, we successfully leveraged these inductive biases with a multi-task learning approach: we jointly trained a deep network to perform image classification and to predict neural activity in macaque primary visual cortex (V1) in response to the same natural stimuli. We measured the out-of-distribution generalization abilities of our resulting network by testing its robustness to common image distortions. We found that co-training on monkey V1 data indeed leads to increased robustness despite the absence of those distortions during training. Additionally, we showed that our network's robustness is often very close to that of an Oracle network where parts of the architecture are directly trained on noisy images. Our results also demonstrated that the network's representations become more brain-like as their robustness improves. Using a novel constrained reconstruction analysis, we investigated what makes our brain-regularized network more robust. We found that our monkey co-trained network is more sensitive to content than noise when compared to a Baseline network that we trained for image classification alone. Using DeepGaze-predicted saliency maps for ImageNet images, we found that the monkey co-trained network tends to be more sensitive to salient regions in a scene, reminiscent of existing theories on the role of V1 in the detection of object borders and bottom-up saliency. Overall, our work expands the promising research avenue of transferring inductive biases from biological to artificial neural networks on the representational level, and provides a novel analysis of the effects of our transfer.

1 Introduction

Although machine learning algorithms have witnessed enormous progress thanks to the recent success of deep learning methods [1], current state-of-the-art deep models [2–4] still fall behind the generalization abilities of biological brains [5, 6]. This includes a lack of robustness to image

35th Conference on Neural Information Processing Systems (NeurIPS 2021).

corruptions as pointed out by Hendrycks and Dietterich [7], who measured a network's performance on 15 different image corruptions applied to the ImageNet [8] test-set. Studies with similar image corruptions have proven to severely decrease performance in classification networks while having a smaller impact on human perception [9], suggesting that the ability to *extrapolate* is weak in these networks compared to the mammalian visual system. This gap in extrapolation has previously been attributed to differences in feature representations [10, 11] and internal strategies for decision making [12] between humans and CNNs.

Historically, neuroscience has inspired many innovations in artificial intelligence [13, 14], and most of this transfer between neuroscience and machine learning happens on the implementational level [15, 13]. However, we currently know too little about the structure of the brain at the level of detail needed to transfer functional generalization properties from biological to artificial systems [5]. To transfer functional inductive biases from the brain to deep neural networks (DNNs), it may thus be better to consider the representational level by capturing biological feature representations in the responses of biological neurons to visual input – abstracting away from the implementational level. In fact, deep neural networks have set new standards in capturing brain activity across multiple areas in the visual cortex [16], becoming the state-of-the-art for neural response prediction of the primary visual cortex (area V1, [17], but also see Marques et al. [18] for biologically inspired models that perform similarly well). Recent work has shown that CNNs, which were fitted to V1 neural data, can generalize well to other neurons, animals, and stimuli [19, 20]. Vice versa, prior work suggests that enforcing brain-like representations in CNNs via neural data from humans [21], mice [22], or monkeys [23] can have beneficial effects on the generalization abilities of these networks to stimuli outside their training distribution for object recognition.

Our work expands on this line of research by exploring the extrapolation capabilities of multi-task learning models (MTL; [24]) trained on image classification and prediction of neural responses from monkey V1. This approach was proposed as the *neural co-training hypothesis* by Sinz et al. [5], but remained untested to the best of our knowledge. We implement MTL via a shared representation between image classification and neural response prediction (Fig. 1). Our hypothesis is that MTL with neural data regularizes the shared representation to inherit good functional inductive biases from neural data, and improves the network's generalization abilities to out-of-distribution images, thus rendering it more robust. We empirically test this idea using common corruptions on tiny ImageNet (TIN)[1]. We show that MTL on monkey V1 has a positive effect on generalization as it increases the model's robustness to image distortions, even though it is trained on undistorted images only. We compare our model with a robust Oracle model to quantify what performance improvement can be expected given that only parts of the network are shared during MTL. Subsequently, we develop a constrained reconstruction based method to analyze learned sensitivities and invariances in the feature representation of the different models. We find that the robust models qualitatively exhibit different feature sensitivities than a standard classification model. Finally, we show that the feature sensitivity of the monkey V1 co-trained model is related to salient image features, consistent with existing theories about the role of V1 [25]. Overall, our results support the neural co-training hypothesis and further expand the scope of prior results, by exploring the relationship between brain-like representations and robustness. To the best of our knowledge, we are the first to analyze learned feature representations of neurally co-trained models and thus take a step towards a semantic understanding of what makes biological vision robust.

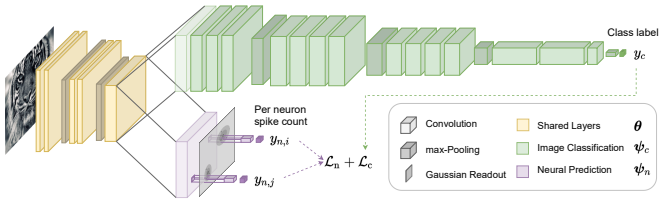


Figure 1: VGG-19 architecture for MTL on image classification and neural prediction.

2 Neural multi-task learning

Data The images for the classification task and the neurophysiological experiments were selected from the ImageNet dataset [8]. For the classification task, we used a grayscale version of TIN[1]. The tiny ImageNet dataset (TIN) is a subset of ImageNet containing 100000 training images of 200 classes (500 for each class) downsized to images of size 64x64. In addition, each class has 50 validation images and 50 test images. For neural prediction, we used neurophysiological recordings of 458 neurons from the primary visual cortices (area V1) of two fixating awake macaque monkeys, recorded with a 32-channel depth electrode during 15 (monkey 1) and 17 (monkey 2) sessions. In each session, approximately 1000 trials of 15 images were presented – each image for 120ms. We extracted the spike count from 40ms to 160ms after image onset. The image set presented to the monkey consists of 24075 images from 964 categories – 25 images per category. Of those, 24000 were designated to model training and 75 to testing. For each training trial, a new subset of 15 images was randomly sampled from the training set. Test images were displayed in fixed order in 5 test trials consisting of 15 images each. Each of the test trials were randomly interleaved among training trials and repeated 40-50 times per session. All images were converted to gray-scale and presented at 420×420 px, covering 6.7° visual angle for the monkey, resulting in 63 pixels per degree (ppd). For model training, images were downscaled and cropped to 64×64 pixels, corresponding to 14.0 ppd.

Architecture All our experiments were based on a variant of the VGG-19 architecture [26] with batch normalization layers [27] after every convolutional layer (Figure 1). To allow for arbitrary image sizes, we made the network fully convolutional by replacing the fully connected readout by three convolutional layers with dropout of 0.5 after the first two, and a final max-pooling operation and softmax [28]. We predicted neural responses by feeding the output of the convolutional layer `conv-3-1`, shown by Cadena et al. [17] to be optimal for predicting V1 responses, into a *Gaussian readout* [19] yielding a spike count prediction per neuron and image.

Models We use a VGG-19, like we described it above, trained on grayscale TIN to serve as the *Baseline* for image classification in our experiments. To prepare our neural co-training, similar to Li et al. [22], we first trained a *Monkey Predictor* model on the image-response pairs of our recorded neural data. We then used that model to predict neural responses for all input images of the TIN classification dataset. These predicted responses served as the basis neural dataset we used in our MTL approach. This allowed us to balance the amount of data we have for each task and it removed trial-to-trial variability in the neural data.

Since co-training only affects the shared representation up to layer `conv-3-1`, we cannot expect the network to be as robust as a network where all layers are trained on data augmented with the image distortions. To explore the limits on robustness resulting from sharing lower layers only, we trained a classification model with a 1:1 mixture of clean and distorted images drawn from the pool of 14 ImageNet-C [7] corruptions (cf. Figure 9 for examples). To push the robustness to the frozen part, we added a second loss that penalizes the Euclidean distance between the outputs of layer `conv-3-1` for the same image augmented with different corruptions – similar to Chen et al. [29]. We then froze all layers up to `conv-3-1`, re-initialized the rest, and re-trained the remaining network on clean data only. We refer to this model as the *Oracle* since it has access to the image distortions during training – unlike our MTL models.

To demonstrate that MTL can in principle transfer robustness properties without showing distorted images in training, we generated neural responses from our Oracle model for all images of the *clean* TIN dataset by freezing the Oracle model and training a Gaussian readout on top of layer `conv-3-1` for 10 epochs to predict V1 data. Then, we trained a model on the resulting neural responses alongside clean image classification using MTL. We call this model *MTL-Oracle*. This model also gives us a realistic "upper bound" for our MTL experiments.

For our main experiment, we trained MTL with the neural responses generated from the Monkey Predictor model, and refer to it as *MTL-Monkey*. This model has never seen distorted images at any point during training. To demonstrate that MTL-Monkey has an effect beyond introducing noise into the training, we perform a control experiment which we refer to as *MTL-Shuffled*. For this, we train a model on the same neural data but with shuffled responses across images for all neurons. An overview of all models used in this study can be found in Table 1.

[1] https://www.kaggle.com/c/tiny-imagenet/overview

Table 1: Overview of the different models that we use in this study.

Model	Classification	Neural Prediction
Baseline	Clean TIN	–
Monkey Predictor	–	Monkey responses
Oracle	Noise augmented TIN	–
MTL-Oracle	Clean TIN	Oracle model responses
MTL-Monkey	Clean TIN	Monkey predictor responses
MTL-Shuffled	Clean TIN	Monkey predictor responses (shuffled)

Training We used cross-entropy loss for single task *image classification* and Poisson loss for single task *neural prediction*. For *multi-task training*, the challenge was to find the optimal balance between the two objectives that achieves a reasonable performance on each task individually, and allows both tasks to benefit from each other by learning common representations. To put both objectives on the same scale, we used their corresponding negative log-likelihood and learned their balance through trainable observation noise parameters σ [30]. This yields a combined loss of $\frac{1}{\sigma_c^2}\mathcal{L}_{\text{CE}}(\boldsymbol{\theta}, \boldsymbol{\psi}_c) + \frac{1}{2\sigma_n^2}\mathcal{L}_{\text{MSE}}(\boldsymbol{\theta}, \boldsymbol{\psi}_n) + \log \sigma_c + \log \sigma_n$ where $\boldsymbol{\theta}$ are the shared parameters and $\boldsymbol{\psi}_c$, σ_c and $\boldsymbol{\psi}_n$, σ_n are the task-specific parameters for $c$lassification and $n$eural prediction, respectively. The classification objective $\mathcal{L}_{\text{CE}}$ was the standard cross-entropy, analogous to the single-task case. For MTL on neural data, we used mean-squared error $\mathcal{L}_{\text{MSE}}$ because the targets are predictions from the network trained on neural data and not the original noisy neural responses. For optimization, we accumulated the gradients over the different losses to optimize the shared parameters $\boldsymbol{\theta}$ in a single combined gradient step. By definition, the two loss components would contribute equally to the learning process.

We standardized all pixel values with the mean and standard deviation of the training set, and augmented the images by random cropping, horizontal flipping, and rotations in a range of 15° for classification. Optimization was performed using stochastic gradient descent with momentum in all classification-related cases, and Adam for single task neural prediction [31]. We used a batch-size of 128 and weight decay with a factor of $5 \cdot 10^{-4}$ throughout all our experiments. The initial learning rate was determined for each task individually and reduced by a (task-specific) factor via an adaptive learning rate schedule. The schedule reduces the learning rate depending on the validation performance – classification performance in the case of MTL – when the rate of improvement is not above 10^{-4} for 5 consecutive epochs. The training was stopped when we reach either five learning rate reduction steps or a maximum number of epochs, that we defined for each task. This training setup was determined via prior hyper-parameter searches on the validation-set. We repeated every experiment with five different random initializations. Error bars were obtained by bootstrapping (250 repetitions).

3 Results

Our goal is to test whether co-training on neural data can lead to improved extrapolation abilities. To this end, we evaluated our model's robustness on distorted copies of the TIN validation-set – used as a test-set in our experiments – following the corruption paradigm of Hendrycks and Dietterich [7]. We reproduced the distortions with an on-the-fly implementation [32], dropped *glass blur* because it is computationally expensive, and refer to our resulting test-set as *TIN-TC*. We quantified the robustness for each of the remaining 14 noise types and five levels of corruption severity separately, and computed a summary robustness score adopted from Hendrycks and Dietterich [7]: $\frac{1}{14}\sum_{c=1}^{14} \overline{A}_c^{\text{robust}} / \overline{A}_c^{\text{Baseline}}$, where $\overline{A}_c = \frac{1}{5}\sum_{l,s=1}^{5} A_{l,c,s}$ denotes the mean accuracy on corruption c across levels l and seeds s.

MTL can successfully transfer robustness Comparing the robustness of the MTL-Oracle model on TIN-TC to the robustness of the single-task *Baseline* model trained on clean TIN only, we saw clear signs of successful transfer (Fig. 2 and Fig. 3A,B) although the MTL network has never seen the image distortions of TIN-TC. In fact, the MTL-Oracle performed close to the Oracle in most cases.

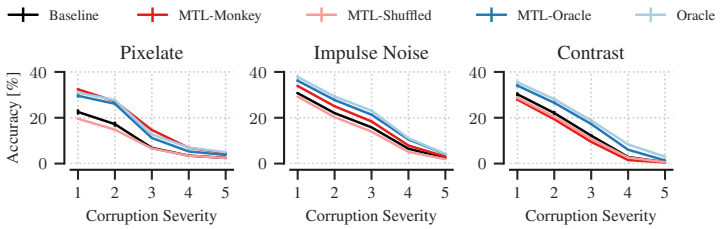


Figure 2: Exemplary classification results on TIN-TC, showing 3 corruption types with the best (left), median (center) and worst (right) robustness scores for MTL-Monkey across 5 increasing levels of severity each.

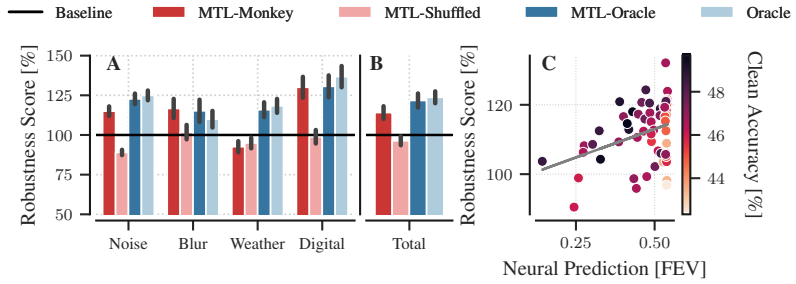


Figure 3: **A** Robustness scores for each model grouped by corruption category, as defined in Hendrycks and Dietterich [7]. **B** Overall robustness scores for our 5 different models. **C** Robustness and neural prediction correlate positively for MTL-Monkey models across 12 different batch-ratios and 5 random seeds per model (grey line: linear regression from neural performance to robustness). Neural prediction performance is measured as the fraction of explained variance (FEV), as described in Cadena et al. [17]. A darker color indicates higher accuracy on the clean TIN test-set.

Co-training with monkey V1 increases robustness. The results of MTL-Oracle show that MTL with neural responses from a robust network in response to undistorted images successfully transfers robustness properties. Furthermore, MTL-Monkey generalized better to the TIN-TC image distortions than the Baseline model, similar to MTL-Oracle, despite the fact that MTL-Monkey has not seen distorted images at any stage during the training process. We found increased robustness for 9/14 image corruptions. This improvement is mainly observed across 3 groups of distortions: *Noise*, *Blur* and *Digital* (Fig. 3A), whereas MTL-Monkey did not exceed the Baseline performance for the *Weather* group. The shuffled control did not provide any benefits (Fig. 2 and Fig. 3A,B), suggesting that the improved robustness of our monkey network is, in fact, due to the original neural data.

The more "brain-like" the neural network, the better it generalizes to image distortions. If features in the neural data affect the robustness, we would expect that the robustness of MTL-Monkey correlates positively with its neural prediction performance on real monkey V1 data. To test this hypothesis, we created a pool of MTL-Monkey models with varying neural performance by altering the amount of neural data introduced during co-training. We ran experiments with different ratios of neural data to image classification that was presented to the network before each backward pass. We ran experiments for ratios $1:15, 1:10, 1:7, 1:5, 1:4, 1:3, 1:2, 1:1, 2:1, 3:1, 4:1$, and $5:1$ with five seeds each, giving us 60 models to plot in Figure 3-C. We found that both the model's test accuracy on clean images and its neural performance on real monkey V1 data improved the network's robustness (Figure 3C; $p < 10^{-4}$ for both neural prediction and clean accuracy[2]).

[2]t-test for both factors in a 2-factor linear regression, in which robustness (dependent variable) is predicted from clean test accuracy for image classification and performance on V1 prediction (independent variables).

Analysis for MTL-Shuffled showed a slight connection between robustness and neural performance ($p = 0.034$ for neural prediction and $p < 10^{-13}$ for clean accuracy). When comparing the regression coefficient of neural prediction in the case of MTL-Monkey and MTL-Shuffled, we found that the influence of neural performance on robustness is two orders of magnitude larger for real neural data $b_{monkey} = 54.72$ than for the shuffled version $b_{shuffled} = 0.26$. Overall, our results are consistent with previous work finding a positive correlation between model robustness and "brain-likeness" [33].

4 Analysis

In the previous section, we showed that our MTL approach can transfer robustness properties and that improved robustness correlates with more brain-like representations learned from monkey V1 data. The aim of this section is to understand the representational differences compared to other models that could be responsible for the increased robustness of our MTL-Monkey model. To this end, we visualize which image features the networks are sensitive to, using a novel resource constrained image reconstruction from a given layer across all of the models used in this study. The rationale behind a constrained reconstruction is to put a resource limitation on the total power of an image, thereby force the reconstruction to put contrast in the image wherever it is necessary to recreate the activity of a given layer, and thus visualize the sensitivities and invariances of this layer (see Figure 4). Specifically, given a noisy target image $\mathbf{x}_0$, we computed the corresponding activations $f(\mathbf{x}_0)$ of a particular layer and reconstructed the original image by minimizing the squared loss between the target activations and the activations from the reconstructed image $\ell(\mathbf{x}_0, \mathbf{x}) = \|f(\mathbf{x}) - f(\mathbf{x}_0)\|^2$ subject to a norm constraint

$$\mathbf{x}^* = \text{argmin}_{\mathbf{x}} \|f(\mathbf{x}) - f(\mathbf{x}_0)\|^2 \text{ s.t.} \|\mathbf{x}\|^2 \leq r^2.$$

Note that, if $\|\mathbf{x}_0\| \leq r$, one trivial optimal solution is $\mathbf{x} = \mathbf{x}_0$. However, if $\|\mathbf{x}_0\| > r$, the constraint becomes active and the reconstruction has to choose where to put power in the image (see Figure 4). This can be seen more formally if we approximate the loss function with a second order Taylor approximation $\ell(\mathbf{x}_0, \mathbf{x}) \approx \frac{1}{2}(\mathbf{x} - \mathbf{x}_0)^\top H (\mathbf{x} - \mathbf{x}_0)$ around the optimal solution $\mathbf{x}_0$. Using the local approximation in the optimization problem and solving for $\mathbf{x}$ using Lagrange multipliers

$$\max_\gamma \min_{\mathbf{x}} \frac{1}{2}(\mathbf{x} - \mathbf{x}_0)^\top H (\mathbf{x} - \mathbf{x}_0) + \gamma \cdot \frac{1}{2}\left(\|\mathbf{x}\|_2^2 - r^2\right) \text{ s.t. } \gamma \geq 0$$

yields $\mathbf{x} = (H + \gamma I)^{-1} H \mathbf{x}_0$ where γ denotes the Lagrange multiplier chosen such that $\|\mathbf{x}^*\| = r$ if the constraint is active. To see that this preferentially reconstructs images along directions where the loss in activation space is more sensitive, consider $\mathbf{x}$ and $\mathbf{x}_0$ in the eigenbasis U of the Hessian $H = U\gamma U^\top$, which we denote by $\mathbf{v} = U^\top \mathbf{x}$ and $\mathbf{v}_0$, respectively. In that space, the solution is

$$\mathbf{v} = (\Lambda + \gamma I)^{-1} \Lambda \mathbf{v}_0 \quad \text{or} \quad v_i = \frac{\lambda_i}{\lambda_i + \gamma},$$

which means that directions with $\lambda_i \gg 0$, i.e. high curvature or strong sensitivity, stay as they are while directions with small λ_i, i.e. low curvature or "invariance", get diminished. How strongly they are diminished depends on the Lagrange multiplier γ, or, equivalently, the norm constraint.

In our analysis, we optimized the pixels based on the activations of layer `conv-3-1` – the co-trained layer – for all five models *MTL-Monkey*, *Baseline*, *Oracle*, *MTL-Oracle* and *MTL-Shuffled*. We found SGD with a learning rate of 5 to be most suitable to reconstruct from all the MTL models, and the Adam optimizer with a learning rate of 0.01 best for the Baseline and Oracle models. We always optimized for 8000 steps per image and model for each norm constraint.

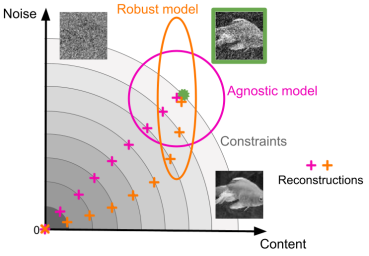


Figure 4: An illustration of the reconstruction process. We reconstruct a noise corrupted target image (green) from a given model under a resource constraint (gray circles). An agnostic model (dark pink) would be, by definition, equally sensitive to: The image content and noise. In contrast, a robust model (orange) would be more sensitive to content and less to noise. Resource constraints that do not allow for a full reconstruction force the optimization to put more image power towards directions to which the model is more sensitive.

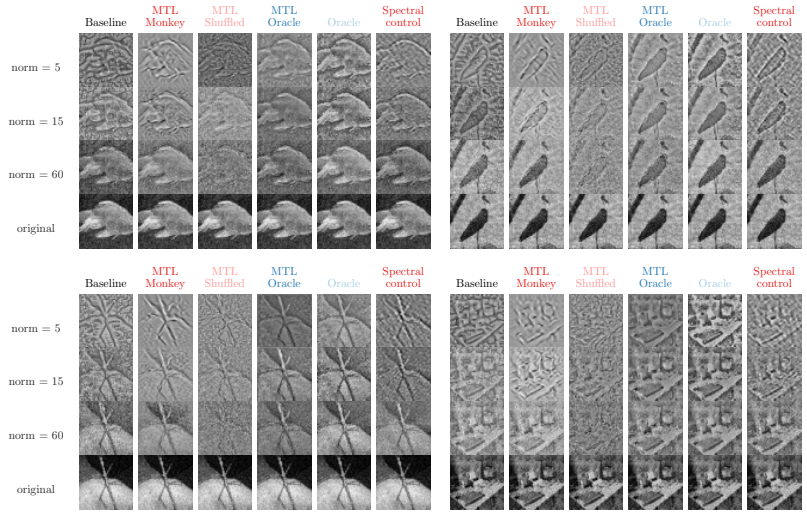


Figure 5: Reconstruction examples from all the 5 models of 4 noisy images (see last row for original images) with Gaussian noise of severity level 2, under 3 norm constraints: 5, 15 and 60. See main text for details on the spectral control.

Reconstructions show qualitative differences between models Reconstructions from test images distorted with Gaussian noise under three different norm constraints qualitatively show that the models are sensitive to different features (Figure 5). When looking at a mid-range norm constraint (norm=15), where we expected the difference between the models to be the largest (see Figure 4), we found that the Baseline model seems to be sensitive to distortions present in the original image. This manifests itself in a stronger noise component in the background of the image reconstructed from the Baseline model compared to other models. The robust networks (MTL-Monkey, Oracle and MTL-Oracle), on the other hand, were less sensitive to these perturbations and exhibited more content structure for that norm constraint (see appendix for more reconstructions). However, when comparing our MTL-Monkey model with the other robust networks, we noticed that the Oracle and MTL-Oracle models generally preserved the original image content as much as possible, while reconstructions from the MTL-Monkey model seem to put slightly more emphasis on edges and object boundaries.

MTL-Monkey's sensitivities cannot be fully explained by frequency filtering One simple mechanism that would make a network more robust against noise types with high-frequency perturbations would be to change the frequency sensitivity towards low-pass components. To assess whether this might be the case for the MTL-Monkey network, we used a simple "spectral control", where we transferred the Fourier amplitude spectrum of the reconstructed image to the original noisy image. This enforces the norm constraint on the original image, and exactly matches the frequency content. If the MTL-Monkey model were simply changing the frequency sensitivity, we would expect the reconstruction and the spectral control to match. However, this does not seem to be the case. The spectral control exhibited more noise in the background and showed less edge enhancing compared to the MTL-Monkey reconstruction. Thus, while changed frequency sensitivity might play a role, there might be additional effects that lead to more noise suppression and more edge enhancement.

MTL-Monkey exhibits increased sensitivity to salient image regions Our constrained reconstruction analysis exposes image content that is relevant to recreate the responses of a particular layer. In this final section, we try to characterize this content in terms of known perceptual mechanisms of mammalian vision. Motivated by the potential role of V1 in bottom-up saliency [25], we specifically investigated the relation between salient regions of an image and the regions that get emphasized in

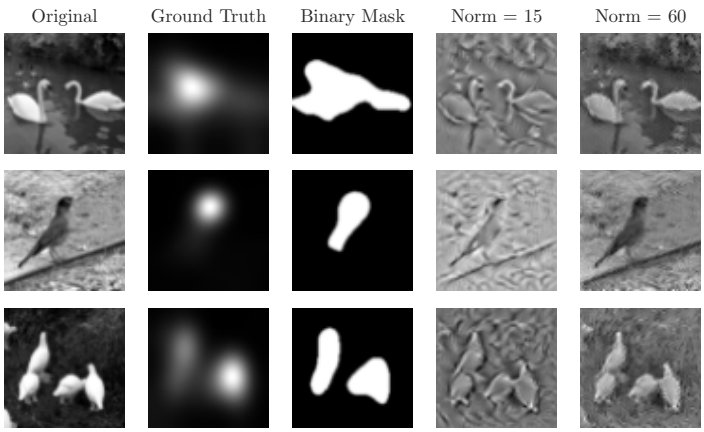


Figure 6: Examples from the ImageNet images used for testing our neural saliency hypothesis in the MTL-Monkey model. In addition to the original image, we show the DeepGaze predicted saliency map and the resulting binarized mask for computing the norm ratio as well as the MTL-Monkey reconstructions with norm constraints (15 and 60).

the reconstructions from different models. Salient regions of an image often include boundaries and shapes of central objects [25, chapter 5]. It could be possible that focusing on more salient regions of an image improves the robustness of a model to corruptions as a correlation between shape-like feature detection and robustness has been reported before [10, 32]. From our qualitative observation of the reconstruction examples (Figure 5), we noticed that the reconstructions from the MTL-Monkey model seem to often enhance the central object in a scene by emphasizing edges and boundaries compared to the reconstructions from our Oracle models. To investigate whether the MTL-Monkey model is more sensitive to salient regions compared to other models, we collected 70 undistorted images with structured background from ImageNet. We used grayscale versions of these images as targets for the reconstruction from our MTL-Monkey, Oracle, and Baseline models. Additionally, we used the DeepGaze II model [34] to predict the saliency maps of these images. To define a binary saliency mask, we took the predicted density map, sorted all resulting pixel values in a descending order, and selected all pixels up to a cumulative sum of 0.7 as "salient" (see section A). Afterwards, we resized the target image and the binarized mask to 64x64 pixels, which is the standard size used for our models (see Figure 6). To quantify how much contrast is spent on the salient region $\text{salient}(I)$ compared to the entire image I, we computed the ratio ϱ of the norm of the salient region against the full image's norm:

$$\varrho(I) = \frac{\|\text{salient}(I)\|_2^2}{\|I\|_2^2}.$$

We then computed the difference $\varrho\left(I_r\right) - \varrho\left(I_o\right)$ between the norm ratio of each reconstructed image I_r and the original image I_o. To put the values on a common scale, we normalized this difference by its maximally achievable value $1 - \varrho\left(I_o\right)$ [inspired by 12]:

$$\bar{\varrho}_r = \frac{\varrho\left(I_r\right) - \varrho\left(I_o\right)}{1 - \varrho\left(I_o\right)}. \tag{1}$$

Our results show that the MTL-Monkey is more sensitive to salient regions than the Oracle and Baseline models across images under low to mid-range norm constraints (Figure 7). And as the norm approaches the norm of the full image, the ratios for both models become more equal to those of the MTL-Monkey model, as expected (right). In comparison to the Baseline and Oracle models, the spectral control seems to be closer to the diagonal. However, in most cases the MTL-Monkey model emphasized the salient regions more strongly, supporting our previous observation that frequency

filtering cannot fully explain the sensitivities of the co-trained model (see section 4). We want to stress that our analysis is purely correlational at this point: Robust MTL monkey models seem to be more sensitive to salient image content than other –robust and non-robust– models. However, if the focus on salient features turns out to be causal, it would open up the possibility that the Oracle models and the MTL-Monkey model are robust due to different reasons.

5 Related work

A number of previous studies also transferred useful inductive biases from biological to artificial neural networks. For instance, Arai et al. [35] utilized neural data –recorded from the superior colliculus (SC) in monkeys– to regularize the network's hidden layers for predicting saccadic eye movements accurately and improving the model's generalization abilities. Moreover, Rezai et al. [36] used the prediction of responses from the primate middle temporal area (MT) to train the lower layers of a deep CNN for visual odometry via multi-task learning, which is the main approach in our work. Although Arai et al. [35] and Rezai et al. [36] leverage the idea of co-training, their main focus is on neuroscientific modeling in contrast to our work. We directly focus on improving machine learning models, which is more in line with other related examples from the literature. Fong et al. [21] used fMRI data of the human brain's activity to guide the training of neural networks on a classification task, obtaining general performance improvements for CNNs. Other works used neural data from animals to regularize the training of artificial networks on image classification, such as in Li et al. [22], through representational distance learning (RDL, [37]). Li and colleagues used mouse V1 to regularize CNNs towards a more brain-like representation. They used the CIFAR-10 and CIFAR-100 datasets, and evaluated robustness on Gaussian noise corruptions. Similar to our findings, they observed an increase in robustness when comparing their network to the Baseline while the model regularized by the shuffled neural data did not yield similar benefits. They also observed increased robustness to adversarial attacks. We extend their work by using a larger set of 14 distinct types of corruption to evaluate the robustness on tiny ImageNet. Federer et al. [23] used neural data from the primary visual cortex of macaque monkeys for regularization, and reported an improvement in test accuracy and an increased robustness to label corruption.

Finally, previous work, summarized by Yamins and DiCarlo [16], found a strong correlation between how well a neural network performs on a classification task and how well it predicts neural responses. Recent work by Dapello et al. [33] also found that robust networks tend to be better at predicting V1 responses in macaque monkeys than non-robust models. They also reported a correlation between the brain-likeness on the representational level of a network and robustness, without explicitly training on neural data, as the model architecture was hand-crafted to emulate V1. In this work, we show similar

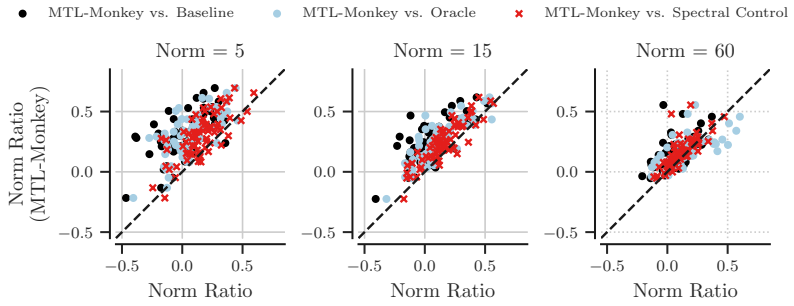


Figure 7: Each datapoint corresponds to one ImageNet image, for which we compute the normalized norm ratio (equation 1) of the reconstructed image from the MTL-Monkey model (y-axis) compared to the Oracle, Baseline or spectral control model in blue, black or red respectively (x-axis). The panels show varying norm constraints as the basis of the reconstructions. For all panels, if the datapoint is above the diagonal, it means that the salient regions in the corresponding image are more emphasized by our MTL-Monkey model than the model associated with the particular marker style.

results on the representational level by directly co-training on neural data while using an established image classification architecture (see sections 2 and 3). Therefore, we believe that our approach of learning a neurally informed representation from data is more flexible and readily generalizable in comparison to Dapello et al. [33]. For example, extending our MTL approach to higher brain areas is straightforward to do, whereas that is not immediately clear with the approach from Dapello et al. [33]. Finally, we think that Dapello et al. [33] supports our results about transferring properties of V1 into our network, especially that the similar behavior on the ImageNet-C test set, with "weather" corruptions also having a weaker performance than the rest, supports the validity of our results.

6 Discussion and conclusion

In this work, we show a successful transfer of robustness properties via multi-task learning on neural data and object classification. Our findings are generally consistent with prior works on inductive bias transfer from the brain, and constitutes a first test of the neural co-training hypothesis [5] for improving the robustness of neural networks. Furthermore, we are the first, to the best of our knowledge, to go a step further by introducing a novel attribution method to understand what makes our neurally co-trained model robust. Through that analysis, we find that our MTL monkey model is more sensitive to salient regions of an image compared to other models. Since V1 has been implicated in bottom-up saliency before [25], it could be that our finding might be connected to this computational property of V1 neurons. In that respect, our results are consistent with the V1 saliency hypothesis, which has been already supported by several studies in the literature [38–41].

In this work, our aim was to learn robustness from a system that is known to be robust against most corruptions – the mammalian visual system [9]. Data augmentation with image corruptions during training –as in our Oracle model– is a simple and effective baseline for robust networks [7, 42]. However, the effectiveness is limited when it comes to unseen corruptions [9], although some noise types might generalize when calibrated carefully [42]. Thus, to get a truly robust network, one would need to anticipate every possible corruption to include them in training, which is obviously intractable. Notably, our MTL approach achieves better generalization without modifying the network's input and without the additional overhead of a noise generator, and we hope that further improvements can eventually replace data corruption. Although our models are still far behind the human visual system in terms of generalization, our work is a conceptual step towards bridging the gap between artificial and biological intelligence. This could be a deciding factor in helping the reliability and generalization capabilities of computer vision. In addition, our findings might help to get a better understanding of the computational role of V1, such as its role in bottom-up saliency. A promising future direction is to include higher brain areas for neural co-training, inspired by Kietzmann et al. [43]. They trained a network for object classification with RDL on neural dynamics of multiple visual areas –including higher ones– and showed that recurrent architectures achieve better test classification performance than feedforward architectures with additional self-connections (ramping feedforward architectures). Thus, using higher areas for neural co-training while potentially relying on recurrence will presumably yield stronger robustness against more complex distortions, and when combined with our analyses, it could improve our understanding of the functional role of these areas as well.[3]

Our work represents fundamental research into the link between biological and artificial vision. Since this direction is at an early stage, the risk of misuse or unethical use of our results is present but not larger than in other fundamental investigations of the principles of vision. Our data is collected through animal experiments which are currently the only method to get high numbers of single unit responses across brain regions. All data coming from monkeys complied with the approved protocol of local authorities (see appendix). A key advantage of the type of data we used is that it is suited for a wide range of analyses beyond this paper. Therefore, our paper simply improves the scientific yield of animal recordings. Furthermore, our approach highlights that it is not strictly necessary to record dozens of new datasets for new tasks - the model that we trained on the original neural data was powerful enough to predict responses to unseen images, and these responses, in turn, can be successfully used for multi-task learning. We do believe that a network trained on a larger amount of already existing neuronal data (possibly also from other areas) of the primate visual system can be used for many more conceivable tasks. Thus, we think that our work contributes to reducing the need for invasive animal experiments and rather encourages the use of surrogate models.

[3] The data and code for this work are made public here: https://github.com/sinzlab/neural_cotraining.

Acknowledgments

We thank Konstantin Lurz, Mohammad Bashiri, Christoph Blessing, Pawel Pierzchlewicz, and Matthias Kümmerer for helpful comments on the manuscript. The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting Arne Nix and Konstantin Willeke.

Funding Transparency Statement

Shahd Safarani was funded by the Claussen-Simon Foundation. This work was partially supported by the Cyber Valley Research Fund (CyVy-RF-2019-01). FHS is supported by the Carl-Zeiss-Stiftung and acknowledges the support of the DFG Cluster of Excellence "Machine Learning – New Perspectives for Science", EXC 2064/1, project number 390727645. This work was supported by an AWS Machine Learning research award to FHS. Supported by the Intelligence Advanced Research Projects Activity via Department of Interior/Interior Business Center contract number D16PC00003, DP1 EY023176 Pioneer Grant (to A.S.T.) and grants from the US Department of Health & Human Services, National Institutes of Health, National Eye Institute (nos. R01 EY026927 to A.S.T. and T32 EY00252037 and T32 EY07001).

References

[1] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015. doi: 10.1038/nature14539. URL `https://doi.org/10.1038/nature14539`.

[2] Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal processing magazine*, 29(6):82–97, 2012.

[3] Waseem Rawat and Zenghui Wang. Deep convolutional neural networks for image classification: A comprehensive review. *Neural computation*, 29(9):2352–2449, 2017.

[4] 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.

[5] Fabian H Sinz, Xaq Pitkow, Jacob Reimer, Matthias Bethge, and Andreas S Tolias. Engineering a less artificial intelligence. *Neuron*, 103(6):967–979, 2019.

[6] Thomas Serre. Deep learning: the good, the bad, and the ugly. *Annual review of vision science*, 5:399–426, 2019.

[7] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*, 2019.

[8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[9] 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 31*, 2018.

[10] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. 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 (ICLR)*, May 2019. URL `https://openreview.net/forum?id=Bygh9j09KX`.

[11] W. Brendel and M. Bethge. Approximating CNNs with bag-of-local-features models works surprisingly well on ImageNet. In *International Conference on Learning Representations (ICLR)*, May 2019.

[12] R. Geirhos, K. Meding, and F. A. Wichmann. Beyond accuracy: quantifying trial-by-trial behaviour of CNNs and humans by measuring error consistency. In *Advances in Neural Information Processing Systems 33*, 2020.

[13] Demis Hassabis, Dharshan Kumaran, Christopher Summerfield, and Matthew Botvinick. Neuroscience-inspired artificial intelligence. *Neuron*, 95(2):245–258, 2017.

[14] K Fukushima. Neocognitron: a self organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics*, 36(4):193, 1980.

[15] David Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. Henry Holt and Co., Inc., New York, NY, USA, 1982. ISBN 0716715678.

[16] Daniel LK Yamins and James J DiCarlo. Using goal-driven deep learning models to understand sensory cortex. *Nature neuroscience*, 19(3):356–365, 2016.

[17] S. A. Cadena, G. H. Denfield, E. Y. Walker, L. A. Gatys, A. S. Tolias, M. Bethge, and A. S. Ecker. Deep convolutional models improve predictions of macaque V1 responses to natural images. *PLoS Computational Biology*, 2019. doi: 10.1101/201764.

[18] Tiago Marques, Martin Schrimpf, and James J. DiCarlo. Multi-scale hierarchical neural network models that bridge from single neurons in the primate primary visual cortex to object recognition behavior. March 2021. doi: 10.1101/2021.03.01.433495. URL `https://doi.org/10.1101/2021.03.01.433495`.

[19] Konstantin-Klemens Lurz, Mohammad Bashiri, Konstantin Willeke, Akshay Jagadish, Eric Wang, Edgar Y. Walker, Santiago A Cadena, Taliah Muhammad, Erick Cobos, Andreas S. Tolias, Alexander S Ecker, and Fabian H. Sinz. Generalization in data-driven models of primary visual cortex. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=Tp7kI90Htd`.

[20] E Y Walker, F H Sinz, E Cobos, T Muhammad, E Froudarakis, P G Fahey, A S Ecker, J Reimer, X Pitkow, and A S Tolias. Inception loops discover what excites neurons most using deep predictive models. *Nat. Neurosci.*, 2019.

[21] Ruth C. Fong, Walter J. Scheirer, and David D. Cox. Using human brain activity to guide machine learning. *Scientific Reports*, 8(1):5397, Mar 2018. doi: 10.1038/s41598-018-23618-6. URL `https://doi.org/10.1038/s41598-018-23618-6`.

[22] Zhe Li, Wieland Brendel, Edgar Walker, Erick Cobos, Taliah Muhammad, Jacob Reimer, Matthias Bethge, Fabian Sinz, Zachary Pitkow, and Andreas Tolias. Learning from brains how to regularize machines. In *Advances in Neural Information Processing Systems*, pages 9529–9539, 2019.

[23] Callie Federer, Haoyan Xu, Alona Fyshe, and Joel Zylberberg. Improved object recognition using neural networks trained to mimic the brain's statistical properties. *Neural Networks*, 131: 103–114, 2020.

[24] Richard A. Caruana. Multitask Learning: A Knowledge-Based Source of Inductive Bias. In *Machine Learning Proceedings 1993*. 1993. doi: 10.1016/b978-1-55860-307-3.50012-5.

[25] Li Zhaoping and Zhaoping Li. *Understanding vision: theory, models, and data*. Oxford University Press, USA, 2014.

[26] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[27] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, page 448–456. JMLR.org, 2015.

[28] John S Bridle. Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. In *Neurocomputing*, pages 227–236. Springer, 1990.

[29] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[30] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491, 2018.

[31] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[32] C. Michaelis, B. Mitzkus, R. Geirhos, E. Rusak, O. Bringmann, A. S. Ecker, M. Bethge, and W. Brendel. Benchmarking robustness in object detection: Autonomous driving when winter is coming. In *Machine Learning for Autonomous Driving Workshop, NeurIPS 2019*, volume 190707484, Jul 2019.

[33] Joel Dapello, Tiago Marques, Martin Schrimpf, Franziska Geiger, David Cox, and James J DiCarlo. Simulating a primary visual cortex at the front of CNNs improves robustness to image perturbations. In *Advances in Neural Information Processing Systems (NeurIPS 2020)*, volume 33, pages 13073–13087. Curran Associates, Inc., 2020.

[34] Matthias Kummerer, Thomas S. A. Wallis, Leon A. Gatys, and Matthias Bethge. Understanding low- and high-level contributions to fixation prediction. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[35] Kuniharu Arai, Edward L Keller, and Jay A Edelman. Two-dimensional neural network model of the primate saccadic system. *Neural networks*, 7(6-7):1115–1135, 1994.

[36] Omid Rezai, Pinar Boyraz Jentsch, and Bryan Tripp. A video-driven model of response statistics in the primate middle temporal area. *Neural Networks*, 108:424–444, 2018.

[37] Patrick McClure and Nikolaus Kriegeskorte. Representational distance learning for deep neural networks. *Frontiers in computational neuroscience*, 10:131, 2016.

[38] Zhaoping Li. A saliency map in primary visual cortex. *Trends in cognitive sciences*, 6(1):9–16, 2002.

[39] Li Zhaoping and Li Zhe. Primary visual cortex as a saliency map: a parameter-free prediction and its test by behavioral data. *PLoS Comput Biol*, 11(10):e1004375, 2015.

[40] Xilin Zhang, Li Zhaoping, Tiangang Zhou, and Fang Fang. Neural activities in v1 create a bottom-up saliency map. *Neuron*, 73(1):183–192, 2012.

[41] Nobuhiko Wagatsuma, Akinori Hidaka, and Hiroshi Tamura. Correspondence between monkey visual cortices and layers of a saliency map model based on a deep convolutional neural network for representations of natural images. *Eneuro*, 8(1), 2021.

[42] Evgenia Rusak, Lukas Schott, Roland S Zimmermann, Julian Bitterwolf, Oliver Bringmann, Matthias Bethge, and Wieland Brendel. A simple way to make neural networks robust against diverse image corruptions. In *European Conference on Computer Vision*, pages 53–69. Springer, 2020.

[43] Tim C. Kietzmann, Courtney J. Spoerer, Lynn K. A. Sörensen, Radoslaw M. Cichy, Olaf Hauk, and Nikolaus 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. ISSN 0027-8424. doi: 10.1073/pnas.1905544116. URL `https://www.pnas.org/content/116/43/21854`.

Energy Guided Diffusion for Generating Neurally Exciting Images

Paweł A. Pierzchlewicz[*,1-2], Konstantin F. Willeke[1-2], Arne F. Nix[1-2], Pavithra Elumalai[2], Kelli Restivo[3-4], Tori Shinn[3-4], Cate Nealley[3-4], Gabrielle Rodriguez[3-4], Saumil Patel[3-4], Katrin Franke[3-4], Andreas S. Tolias[3-5], Fabian H. Sinz[1-4]

[1]Institute for Bioinformatics and Medical Informatics, Tübingen University, Tübingen, Germany
[2]Institute of Computer Science and Campus Institute Data Science, University of Göttingen, Germany
[3]Department of Neuroscience, Baylor College of Medicine, Houston, TX, USA
[4]Center for Neuroscience and Artificial Intelligence, Baylor College of Medicine, Houston, TX, USA
[5]Department of Electrical and Computer Engineering, Rice University, Houston, TX, USA
[*]ppierzc@cs.uni-goettingen.de

Abstract

In recent years, most exciting inputs (MEIs) synthesized from encoding models of neuronal activity have become an established method for studying tuning properties of biological and artificial visual systems. However, as we move up the visual hierarchy, the complexity of neuronal computations increases. Consequently, it becomes more challenging to model neuronal activity, requiring more complex models. In this study, we introduce a novel readout architecture inspired by the mechanism of visual attention. This new architecture, which we call attention readout, together with a data-driven convolutional core outperforms previous task-driven models in predicting the activity of neurons in macaque area V4. However, as our predictive network becomes deeper and more complex, synthesizing MEIs via straightforward gradient ascent (GA) can struggle to produce qualitatively good results and overfit to idiosyncrasies of a more complex model, potentially decreasing the MEI's model-to-brain transferability. To solve this problem, we propose a diffusion-based method for generating MEIs via Energy Guidance (EGG). We show that for models of macaque V4, EGG generates single neuron MEIs that generalize better across varying model architectures than the state-of-the-art GA, while at the same time reducing computational costs by a factor of 4.7x, facilitating experimentally challenging closed-loop experiments. Furthermore, EGG diffusion can be used to generate other neurally exciting images, like most exciting naturalistic images that are on par with a selection of highly activating natural images, or image reconstructions that generalize better across architectures. Finally, EGG is simple to implement, requires no retraining of the diffusion model, and can easily be generalized to provide other characterizations of the visual system, such as invariances. Thus, EGG provides a general and flexible framework to study the coding properties of the visual system in the context of natural images.[1]

1 Introduction

From the early works of Hubel and Wiesel [1], visual neuroscience has used the preferred stimuli of visual neurons to gain insight into the information processing in the brain. In recent years, deep learning has made big strides in predicting neuronal responses [2–16] enabling *in silico* stimulus

[1]The code is available at `https://github.com/sinzlab/energy-guided-diffusion`

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

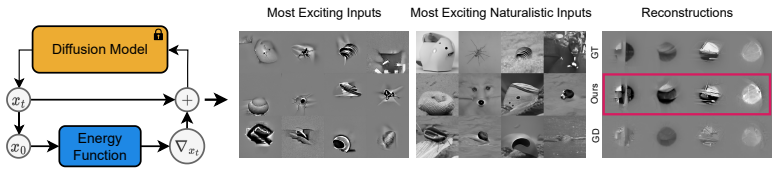


Figure 1: **Schematic** of the EGG diffusion method with a pre-trained diffusion model. Examples of applications: **Left**: Most Exciting Inputs for different neurons, **Middle:** Most Exciting Naturalistic Inputs matched unit-wise to the MEIs. **Right**: Reconstructions in comparison to the ground truth (top) and gradient descent optimized (bottom).

synthesis of non-parametric most exciting inputs (MEIs) [17–19]. MEIs are images that strongly drive a selected neuron and can thus provide insights into its tuning properties. Up until now, they have been successfully used to find novel properties of neurons in various brain areas in mice and macaques [17–24].

However, as we move up the visual hierarchy, such as monkey visual area V4 and IT, the increasing non-linearity of neuronal responses with respect to the visual stimulus makes it more challenging to ❶ obtain models with high predictive performance for single neurons, and ❷ optimize perceptually plausible MEIs, that is, those not corrupted by adversarial high-frequency noise for example. Particularly, area V4 is known to be influenced by attention effects [25], and shifts in attention before the onset of saccades can change the location of its neurons' receptive fields [26, 27]. When models become more complex or units are taken from deeper layers of a network, existing MEI optimization methods based on gradient ascent (GA) can sometimes have difficulties producing qualitatively good results [28] and can overfit to the idiosyncrasies of more complex models, potentially decreasing the MEI's model-to-brain transferability. Typically, these challenges are addressed by biasing MEIs towards the statistic of natural images, for instance by gradient pre-conditioning [28], by including a total variation loss to reduce high-frequency noise [29] or by image synthesis via GANs [19]. However, as discussed by Engstrom et al. [30] and Feather et al. [31] including additional priors into the generation process can result in obfuscated model biases.

Here, we make two contributions towards the above points: ❶ We introduce a new model architecture, called the attention readout, for predicting the activity of neurons in macaque area V4 , which together with a data-driven convolutional core outperforms previous task-driven models [24, 32]. ❷ To improve the quality of MEI synthesis we introduce a novel method for optimizing MEIs via Energy Guided Diffusion (EGG). EGG diffusion guides a pre-trained diffusion model with a learned neuronal encoding model to generate MEIs with a bias towards natural image statistics. Our proposed EGG method is simple to implement and, in contrast to similar approaches [33–35], requires no retraining of the diffusion model (Fig. 1). We show that EGG diffusion not only yields MEIs that generalize better across architectures and are thus expected to drive real neurons equally well or better than GA-based MEIs but also provides a significant (4.7x) speed up over the standard GA method enhancing its utility for close-loop experiments such as inception loops [17, 18, 20, 24]. Since optimizing MEIs for thousands of neurons can take weeks [24], such a speed-up directly decreases the energy footprint of this technique. Moreover, the rapid verification of synthesized images *in vivo* is particularly important for close-loop experiments given that maintaining the stability of single unit recordings is challenging, and there's also the issue of representational drift [36], where tuning functions can change over time. We also demonstrate that EGG diffusion straightforwardly generalizes to provide other characterizations of the visual system that can be phrased as an inverse problem, such as image reconstructions based on neuronal responses. The flexibility and generality of EGG thus make it a powerful tool for investigating the neural mechanisms underlying visual processing.

2 Attention readout for macaque area V4

Background Deep network-based encoding models have set new standards in predicting neuronal responses to natural images [2–15]. Virtually all architectures of these encoding models consist of at least two parts: a *core* and a *readout*. The core is usually implemented via a convolutional

network that extracts non-linear features $\Phi(\boldsymbol{x})$ from the visual input and is shared across all neurons to be predicted. It is usually trained through one of two paradigms: i) *task-driven*, where the core is pre-trained on a different task like object recognition [3, 4, 37–39] and then only the readout is trained to predict the neurons' responses or ii) *data-driven* where the model is trained end-to-end to predict the neurons' responses. The *readout* is a collection of predictors that map the core's features to responses of individual neurons. With a few exceptions [40], the readout components and its parameters are neuron-specific and are therefore kept simple. Typically, the readout is implemented by a linear layer with a rectifying non-linearity. Different readouts differ by the constraints they put on the linear layer to reduce the number of parameters [3, 4, 37, 40–42]. One key assumption all current readout designs make is that the readout mechanism does not change with the stimulus. In particular, this means that the location of the receptive field is fixed. While this assumption is reasonable for early visual areas like V1, it is not necessarily true for higher or mid-level areas such as macaque V4, which are known to be affected by attention effects and can even shift the location of the receptive fields [26]. This motivated us to create a more flexible readout mechanism for V4.

State-of-the-art model: Robust ResNet core with Gaussian readout In this study, we compare our data-driven model to a task-driven model [24], which is also composed of a *core* and *readout*. The core is a pre-trained robust ResNet50 ($L_2, \varepsilon = 0.1$) [43, 44]. We use the layers up to layer 3 in the ResNet, which has 1,024 channels, thus providing a 1,024 dimension feature space. Then batch normalization is applied [45], followed by a ReLU non-linearity. The *Gaussian readout* [40] learns the position of each neuron and extracts a feature vector at this position. During training, the positions are sampled from a 2D Gaussian distribution with means μ_n and Σ_n, during inference the μ_n positions are used. Then the extracted features are used in a linear non-linear model to predict neuronal responses. We will refer to this model as the **Gaussian model**.

Proposed model: Data-driven core with attention readout The predictive model is trained from scratch to predict the neuronal responses in an end-to-end fashion. Following Lurz et al. [40], the architecture is comprised of two main components. First, the *core*, a four-layer CNN with 64 channels per layer with an architecture identical to Lurz et al. [40]. Secondly, the attention *readout*, which builds upon the attention mechanism [46, 47] as it is used in the popular transformer architecture [48]. After adding a fixed positional embedding to $\Phi(\boldsymbol{x})$ and normalization through LayerNorm [49] to get $\tilde{\Phi}(\boldsymbol{x})$, key and value embeddings are extracted from the core representation. This is done by position-wise linear projections $V \in \mathbb{R}^{c \times d_k}$ and $U \in \mathbb{R}^{c \times d_v}$ both of which have parameters shared across all neurons. Then, for each neuron a learned query vector $\mathbf{q}_n \in \mathbb{R}^{d_k}$ is compared with each position's key embedding using scaled dot-product attention [48].

$$\alpha_n = \mathrm{softmax}\left(\sum_{c,d_k} \frac{\tilde{\Phi}(\boldsymbol{x})_c W_{c,d_k} q_{n,d_k}}{\sqrt{d_k}}\right) \tag{1}$$

The result is a spatially normalized attention map $\alpha_n \in \mathbb{R}^{h \times w \times 1}$ that indicates the most important feature locations for a neuron n given an input image. Using this attention map to compute a weighted sum of the value embeddings gives us a single feature vector for each neuron. Finally, a neuron-specific affine projection with ELU non-linearity [50] gives rise to the predicted spike rate $\hat{r}_n$ (Fig. 2A). The model training is performed by minimizing the Poisson loss using the same setup as described in Willeke et al. [24]. We will refer to this model as the **Attention model**.

Training data We use data from 1,244 Macaque V4 neurons from Willeke et al. [24] and briefly summarize their data acquisition in the supplementary materials section A.1.

Results Our Attention model significantly outperforms the Gaussian model in predicting neuronal responses of macaque V4 cells on unseen natural and model-derived images. We evaluate the model performance by the correlation between the model's prediction and the averages of actual neuron responses across multiple presentations of a set of test images, as described by Willeke et al. [24]. We compared this predictive performance to the Gaussian model [44] on 1,244 individual neurons (Fig. 2B). The Attention model significantly outperforms the Gaussian model by 12% (Wilcoxon signed-rank test, p-value $= 6.79 \cdot 10^{-82}$). In addition, we evaluated the new readout on how well it predicts the real neuronal responses to 48 MEIs generated from the Gaussian model [see 24] and 7 control natural images. Our Attention model is better at predicting real neuronal responses,

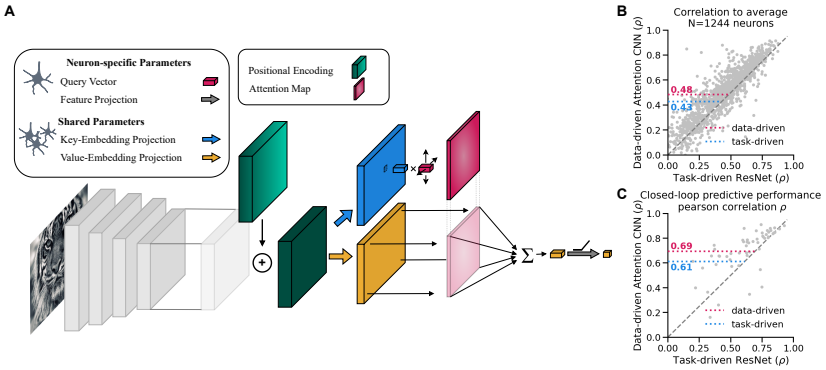


Figure 2: **a**) Schematic of the Attention Readout. **b**) Correlation to average scores for 1,244 neurons. The Attention model (pink) shows a significant (as per the Wilcoxon signed rank test, p-value $= 6.79 \cdot 10^{-82}$) increase in the mean correlation to average in comparison to the Gaussian model (blue). **c**) Predictive performance comparison of the two models in a closed-loop MEI evaluation setting. Showing that the data-driven with attention readout model better predicts the in-vivo responses of the MEIs.

even for MEIs of another architecture (Fig. 2C). Please note that Willeke et al. [24] experimentally verified MEIs in only a subset of neurons and only used the neurons with high functional consistency across different experimental sessions. For that reason, we too can only compare the performance of model-derived MEIs on this subset of neurons. We additionally show that the Attention model and Gaussian model show representational similarity (see Table S1) and that the Attention model uses its ability to shift its receptive field (Fig. S1).

Readout \ Core	Task-Driven	Data-Driven
Factorized	-	0.153
Gaussian	0.262	0.229
Attention	0.276	**0.294**

Table 1: Ablation study test correlation comparison for combinations of different cores and readouts. Bold indicates the best-performing model.

Ablation Study We perform an ablation study comparing the effects of the choice of core and readout on the performance in terms of test correlation (Table 1). We identify that the data-driven core + Attention readout model outperforms all previous setups. Furthermore, the ablation study shows that the Attention readout generally improves performance across cores.

3 Energy guided diffusion (EGG)

3.1 Algorithm and methods

In this section, we describe our approach to extract tuning properties of neuronal encoding models using a natural image prior as described by a diffusion model. In brief, we use previously established links between diffusion and score-based models and the fact that many tuning properties can be described as inverse problems (most exciting image, image reconstruction from neuronal activity, etc.) to combine an energy landscape defined by the neuronal encoding model with the energy landscape defined by the diffusion model and synthesize images via energy minimization. We show that this method leads to better generalization of MEIs and image reconstructions across architectures, faster generation, and allows for generating natural-looking stimuli.

Background: diffusion models Recently, Denoising Diffusion Probabilistic Models (DDPMs) have proved to be successful at generating high-quality images [33, 51–56]. These models can be formalized as a variational autoencoder with a fixed encoder $\boldsymbol{x}_0 \mapsto \boldsymbol{x}_T$ that turns a clean sample $\boldsymbol{x}_0$ into a noisy one $\boldsymbol{x}_T$ by repeated addition of Gaussian noise, and a learned decoder $\boldsymbol{x}_T \mapsto \boldsymbol{x}_0$ [33],

which is often described as inverting a diffusion process [51]. After training, the sampling process is initialized with a standard Normal sample $\boldsymbol{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ which is iteratively "denoised" for T steps until $\boldsymbol{x}_0$ is reached. In the encoding, each step t corresponds to a particular noise level such that

$$\boldsymbol{x}_t = \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\varepsilon}_0 \tag{2}$$

where $\bar{\alpha}_t$ controls the signal strength at time t and $\varepsilon_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is independent Gaussian noise. In the decoding step, the diffusion model predicts the noise component $\varepsilon_\theta(\boldsymbol{x}_t, t)$ at each step t of the diffusion process [33]. Then the sampling is performed according to

$$\boldsymbol{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_\theta\left(\boldsymbol{x}_t, t\right)\right) + \sigma_t \boldsymbol{z} \tag{3}$$

where $\boldsymbol{z} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$.

Several previous works have established a link between diffusion models and energy-based models [57–59]. In particular, the diffusion model $\varepsilon_\theta\left(\boldsymbol{x}_t, t\right)$ can be interpreted as a *score function*, i.e. the gradient of a log-density or energy w.r.t. the data $\nabla_{\boldsymbol{x}} \log p(\boldsymbol{x})$ [60]. This link is particularly useful since combining two density models via a product is equivalent to adding their score functions.

EnerGy Guided Diffusion (EGG) To optimize neurally exciting images, we require a method that can guide diffusion models via neural encoding models. The parameterization of diffusion models introduced by Ho et al. [33] only allows for the unconditioned generation of samples. Dhariwal and Nichol [53] introduced a method for sampling from a conditional distribution $p_t(\boldsymbol{x} \mid \boldsymbol{y})$, with diffusion models using a classifier $p_t(\boldsymbol{y} \mid \boldsymbol{x})$ known as classifier guidance. However, this method requires i) the classifier to be trained on the noisy images, and ii) is limited to conditions for which classification makes sense. Essentially, this method relies on computing the score of the posterior distribution.

$$\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t \mid \boldsymbol{y}) = \nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t) + \nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{y} \mid \boldsymbol{x}_t) \tag{4}$$

For classifier-guidance, the gradient of a model $\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{y} \mid \boldsymbol{x}_t)$ with respect to the noisy input $\boldsymbol{x}_t$ is combined with the diffusion model $\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t)$, resulting in samples $\boldsymbol{x}_0$ conditioned on the class $\boldsymbol{y}$. Note that this requires a model $\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{y} \mid \boldsymbol{x}_t)$ that has been trained on noisy samples of the diffusion before. Here we extend this approach to i) use neuronal encoding models, such as the ones described above, to guide the diffusion process and ii) use a model trained on *clean* samples only. We achieve i) by defining conditioning as a sum of energies. Specifically, we redefine equation (4) in terms of the output of the diffusion model $\varepsilon_\theta(\boldsymbol{x}_t, t)$ and an arbitrary energy function $E(\boldsymbol{x}_t, t)$:

$$\bar{\varepsilon}(\boldsymbol{x}_t, t) = \varepsilon_\theta(\boldsymbol{x}_t, t) + \lambda_t \nabla_{\boldsymbol{x}_t} E(\boldsymbol{x}_t, t) \tag{5}$$

where λ_t is the energy scale. This takes advantage of the fact that sampling in DDPMs is functionally equivalent to Langevin dynamics [51]. Langevin dynamics generally define the movement of particles in an energy field and in the special case when $E(x) = -\log p(x)$, Langevin dynamics generates samples from $p(x)$. For this study, we use a constant value of λ and normalize the gradient of the energy function to a magnitude of 1.

To achieve ii) we use an approximate clean sample $\bar{\boldsymbol{x}}_0$, i.e. the original image, that can be estimated at each time step t. This is achieved by a simple trick introduced in Li et al. [61]. By inverting the forward diffusion process, with the assumption that the predicted $\varepsilon_\theta(\boldsymbol{x}_t, t)$ is the true noise:

$$\bar{\boldsymbol{x}}_0(\boldsymbol{x}_t, t) = \frac{1}{\sqrt{\bar{\alpha}_t}}(\boldsymbol{x}_t - \sqrt{1-\bar{\alpha}_t}\varepsilon_\theta(\boldsymbol{x}_t, t)). \tag{6}$$

As a result, the energy function receives inputs that are in the domain of $\boldsymbol{x}_0$ at much earlier time steps t, and hence makes it feasible to use energy functions only defined on $\boldsymbol{x}_0$ and not $\boldsymbol{x}_t$, dropping the requirement to provide an energy $E(\boldsymbol{x}_t, t)$ that can take noisy images. Thus, the new score can be defined as

$$\bar{\varepsilon}(\boldsymbol{x}_t, t) = \varepsilon_\theta(\boldsymbol{x}_t, t) + \lambda_t \nabla_{\boldsymbol{x}_t} E(\bar{\boldsymbol{x}}_0(\boldsymbol{x}_t, t)) \tag{7}$$

This is particularly relevant in the domain of neural system identification, as encoding models are trained on neuronal responses to natural "clean" images [2–15, 17, 21, 24, 40]. To get an energy that can understand noisy images would require showing the noisy images to the animals in experiments, which would make the use of this method prohibitively more difficult. Therefore, a guidance method that does not require training an additional model on noisy images allows researchers to apply EGG diffusion directly to existing models trained on neuronal responses and extract tuning properties from them.

Related work Many other methods have been proposed to condition the samples of diffusion processes on additional information. Ho and Salimans [55] provided a method that addressed the second requirement of classifier-guidance by incorporating the condition $\boldsymbol{y}$ into the denoiser $\varepsilon_\theta(\boldsymbol{x}_t, t, \boldsymbol{y})$. However, to introduce a conditioning domain $\boldsymbol{y}$ in this classifier-free guidance, the whole diffusion model needs to be retrained. Furthermore, this link between diffusion models and energy-based models allowed several previous works to compose diffusion models to generate outputs that contain multiple desired aspects of a generated image [57–59]. However, these studies focus solely on generalizing the classifier-free guidance to allow guiding diffusion models with other diffusion models. Nichol and Dhariwal [52] have used a similar gradient conditioning to guide the diffusion process using the gradient of the dot product of the CLIP image and text vectors. It has been shown that CLIP models that have not been trained on noisy images can be used for guiding diffusion models [62, 63]. Kadkhodaie and Simoncelli [64] introduced a stochastic coarse-to-fine gradient ascent procedure for generating samples from the implicit prior embedded within a CNN. While we were working on this project, Feng et al. [65] published a preprint where they used the score-based definition of diffusion models to introduce an image-based prior for inverse problems where the posterior score function is available. This work is most closely related to our approach. However, they focus on how to obtain samples and likelihoods from the true posterior. For that reason, they need guiding models to be proper score functions. We do not need that constraint and focus on guiding inverse problems defined by a more general energy function and focus particularly on the application to neuronal encoding models.

Image preprocessing for neural models The neural models used in this study expect 100×100 images in grayscale. However, the output of the ImageNet pre-trained Ablated Diffusion Model (ADM) [53] is a 256×256 RGB image. We, therefore, use an additional compatibility step that performs i) downsampling from $256 \times 256 \rightarrow 100 \times 100$ with bilinear interpolation and ii) takes the mean across color channels providing the grayscale image. Each of these preprocessing steps is differentiable and is thus used end-to-end when generating the image.

3.2 Experiments

Most exciting images We apply EGG diffusion to characterize the properties of neurons in macaque area V4. For each of these experiments, we use the pre-trained ADM diffusion model trained on 256×256 ImageNet images from Dhariwal and Nichol [53]. In each of our experiments, we consider two paradigms: 1) **within** architecture, where we use two independently pre-trained ensembles containing 5 models of the same architecture (Gaussian model or Attention model). We generate images on one and evaluate them on the other. 2) **cross** architecture, two independently pre-trained ensembles containing 5 models of different architectures (Gaussian model and Attention model). We demonstrate EGG on three tasks ❶ Most Exciting Input (MEI) generation, where the generation method needs to generate an image that maximally excites an individual neuron, ❷ naturalistic image generation, where a natural-looking image is generated that maximizes individual neuron responses, and ❸ reconstruction of the input image from predicted neuronal responses. Running the experiments required a total of 7 GPU days. All computations were performed on a single consumer-grade GPU: NVIDIA GeForce RTX 3090 or NVIDIA GeForce RTX 2080 Ti depending on the availability.

MEIs have served as a powerful tool for visualizing features of a network, providing insights and testable predictions [17–21, 23, 66]. For the generation of MEIs, we selected 90 units at random from a subset of all 1,244 for which both the Gaussian model and the Attention model achieve at least a correlation of 0.5 to the average responses across repeated presentations. We compare our method to a vanilla gradient ascent (GA) method [24] which optimizes the pixels of an input image $\boldsymbol{x}$ to obtain the maximal response of the selected neuron. For the GA method, we use Gaussian blur preconditioning of the gradient. The stochastic gradient descent (SGD) optimizer was used with a learning rate of 10 and the image was optimized for 1,000 steps. We also evaluated other setups for the GA method without finding major differences (see Fig. S2). We define EGG diffusion with the energy function $E(\bar{\boldsymbol{x}}_0) = f_i(\bar{\boldsymbol{x}}_0)$, where f_i is the i-th neuron model and $\bar{\boldsymbol{x}}_0$ is the estimated clean sample. We optimize MEIs for both the Gaussian model and the Attention model. We set the energy scale to $\lambda = 10$ for the Gaussian model and $\lambda = 5$ for the Attention model. λ was chosen via a grid search, for more details refer to Fig. 5B. The diffusion process was run for 100 respaced time steps for the Gaussian model and 50 respaced time steps for the Attention model. For both EGG and GA, we set the norm of the 100×100 image to a fixed value of 25. For each of the methods, we chose

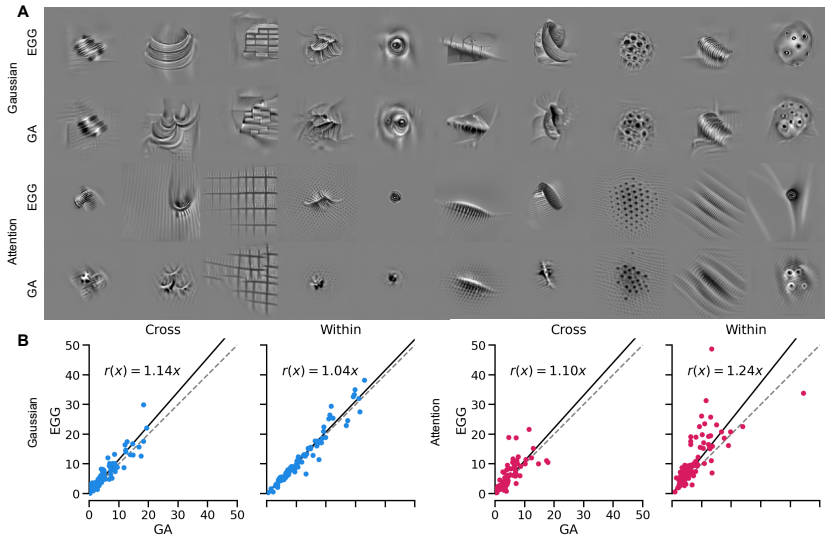


Figure 3: **a**) Examples of MEIs optimized using EGG diffusion and GA for macaque V4 Gaussian and Attention models. **b**) Comparison of activations for different neurons between EGG diffusion and GA on the Within and Cross Architecture validation paradigms. Line fits obtained via Huber regression with $\varepsilon = 1.1$. Curated image selection to show various properties of the neurons like fur, eyes, curves and edges.

the best of 3 MEIs optimized from different seeds. We show the influence of the initial seed on the generated MEI in figure S3. Furthermore, the images that are generated by the ADM model are RGB. We show examples of the color outputs in figure S4.

We show some examples of MEIs generated with EGG diffusion and GA for the two architectures in figure 3A. For more examples, refer to the supplementary materials figure S5. We find that the EGG-generated MEIs are significantly better (Attention) or similarly (Gaussian) activating within architectures and are significantly better at generalizing across architectures (Fig. 3B). This can also be observed by a significant increase in the mean activation across all units (Table 2). Perceptually, EGG-generated MEIs of the Attention model looked more complex and natural than the GA-generated MEIs, and more similar to MEIs of the Gaussian model pre-trained on natural image classification.

Comparing EGG-based MEIs to the ones found by Willeke et al. [24] using GA, we find that the preferred image feature is usually preserved, but MEIs generated for the Attention model are in most cases smaller in visual angle than their Gaussian model counterparts (Fig. S5). To quantify that the MEIs from the Attention model are smaller we compute an isotropic Gaussian envelope for the MEIs. We find that the Attention model generates MEIs for which their Gaussian envelope on average is smaller than for the Gaussian MEIs (σ_{At} = 49.62 vs σ_{Ga} = 55.36, Wilcoxon signed rank test p-value: 0.0078).

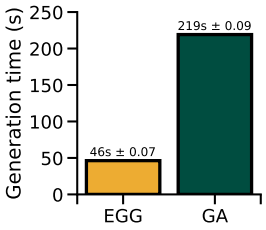


Figure 4: Mean comparison of the generation times between the EGG and GA (error bars denote standard error).

Finally, EGG diffusion is almost 4.7-fold faster than GA, requiring only on average 46s per MEI in comparison to the required 219s for the GA method (Fig. 4) on a single NVIDIA GeForce RTX 3090 across 10 repetitions. This

Paradigm	Within (Gaussian)	Cross (Gaussian)	Within (Attention)	Cross (Attention)
Gradient Ascent	11.43	5.51	7.59	4.42
EGG Diffusion	**11.76**	**6.53**†	**10.56**†	**5.50**†

Table 2: Comparison of the average unit activations in response to MEIs in two paradigms 1) within architectures and 2) cross architectures, for two architectures Gaussian and Attention. Bold marks the method which has higher mean activation, and the † marks the increases which are statistically significant (Wilcoxon signed-rank test, respective p-values: 0.08, $2.87{\cdot}10^{-6}$, $2.84{\cdot}10^{-10}$, $4.39{\cdot}10^{-5}$). The architecture in the bracket indicates the generator architecture.

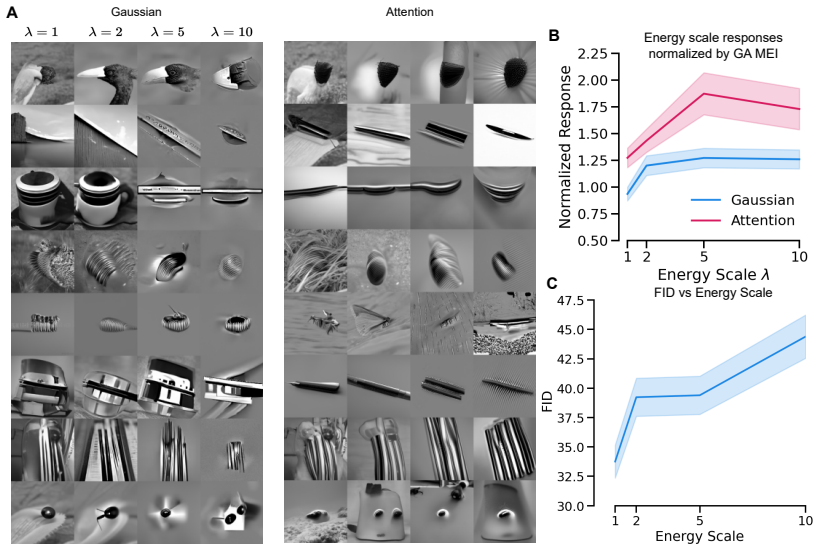


Figure 5: **a**) Examples of MENIs optimized using EGG diffusion in the macaque V4 for different neurons and different energy scales $\lambda \in \{1, 2, 5, 10\}$. **b**) Mean and standard error of the activations of neurons across different energy scales normalized by the activation of the GA MEI. **c**) FID between MEIs across energy scales and top-5 activating ImageNet images.

is a substantial gain, as Willeke et al. [24] required approximately 1.25 GPU years to optimize the MEIs presented in their study. With EGG, only approximately 0.25 GPU years would be needed to produce the results of the study, while providing higher quality and higher resolution MEIs. Thus, EGG can provide major savings in time and energy, *and* improve the quality of MEIs.

Controlling the "naturalness" to generate most exciting natural images Unlike GA, EGG can also be used to synthesize more natural-looking stimuli by controlling the energy scale hyperparameter λ. Changing the value of λ trades off the importance of the maximization property of the image and its "naturalness". To demonstrate this, we generated images for 150 neurons with the highest correlations to the average for the Gaussian model. We used energy scales $\lambda \in \{1, 2, 5, 10\}$, fixed the 100×100 image norm to 50, and used 50 steps re-spaced from 1000. Each image was generated using 3 different seeds and the best-performing image on the generator model was selected.

We show examples of the generated images across different energy scales in figure 5A for both the Gaussian model and the Attention model. For more examples, refer to the supplementary material (Fig. S6, Fig. S7). We subsequently quantified the predicted responses across different values of λ. We find that increasing λ increases the predicted responses (Fig. 5B), however, at higher λ values

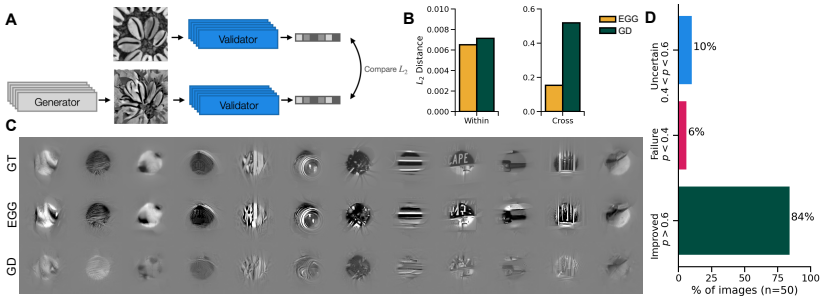


Figure 6: **a**) Schematic of the reconstruction paradigm. The generated image is compared to the ground truth image via L_2 distance in the unit activations space. Reconstructions from 1,244 units. **b**) L_2 distances in the unit activations space for the Within and Cross architecture domains comparing the EGG and GD generation methods. Shows that the EGG method generalizes better than GD across architectures. **c**) examples of reconstructions generated by EGG and GD in comparison to the ground truth (GT). **d**) Survey results on 45 voluntary human participants. Indicates that in 84% of images, the participants preferred the EGG generated reconstructions with a rate ≥ 0.6.

the responses begin to plateau, or even decrease. Therefore, for generating MEIs, we use $\lambda = 10$ for the Gaussian model and $\lambda = 5$ for the Attention model. It can be further observed that decreasing λ increases the naturalness of the generated image while preserving the features of the image that the neuron is tuned towards. To quantify the increase in the naturalness of the MEIs across λs, we measured the FID score between the generated images at different λ values and the top-5 ImageNet images (Fig. 5C). Our results show that by changing λ we approach the natural images manifold (lower FID). We also find that EGG generates MEIs ($\lambda = 1$) similarly activating to the top-1 Imagenet images (Fig. S8).

Image reconstruction from unit responses Another application of EGG diffusion is image reconstruction from neuronal responses. A similar task has been attempted with success using diffusion models from human fMRI data [34, 35]. Given that only a small fraction of neurons were recorded, the image is encoded in an under-complete, significantly lower-dimensional space. Therefore, it is to be expected that the reconstructed image $\boldsymbol{x}$ will not necessarily be equal to the ground truth image $\boldsymbol{x}_{gt}$. However, a better reconstruction $\boldsymbol{x}^*$ is one that generalizes across models. Therefore, regardless of the model f used, we should get $||f(\boldsymbol{x}^*) - f(\boldsymbol{x}_{gt})||_2 = 0$. This is trivially true for $\bar{x}_0 = x_{gt}$ but, given the complexity of the model, there are likely other solutions. We therefore consider a masked version of the reconstructions for visualization. We mask the reconstructions to the joint receptive field of all 1,244 neurons. The mask is obtained by computing the average absolute gradients $\text{mask} = \mathbb{E}_x[|\nabla_x f(x)|]$ across the responses to the test images. The masks were normalized to be between 0 and 1 and the values below 0.25 are clamped to 0.

We can reconstruct images in the EGG framework by defining the energy function as an L_2 distance between the predicted responses to the ground truth image $f(\boldsymbol{x}_{gt})$ and the predicted responses to a generated image (Fig. 6A) $E(\boldsymbol{x}) = ||f(\boldsymbol{x}) - f(\boldsymbol{x}_{gt})||_2$. Note that, instead of $f(\boldsymbol{x}_{gt})$, we could also use recorded neuronal responses. The images are generated from the Gaussian model with $\lambda = 2$ and 1000 timesteps, with the norm of the 100×100 image fixed to 60. We compare EGG to a gradient descent (GD) method that simply minimizes the L2 distance. The GD uses an AdamW optimizer with a learning rate of 0.05. In GD, at each optimization step the image $\boldsymbol{x}_t$ is Gaussian blurred and the norm is set to 60 before passing it to the neural encoding model. We optimize the GD reconstruction up to the point where the train L_2 distance is matched between the GD and the EGG for a fair comparison of the generalization capabilities. We verified that the GD images do not improve qualitatively with more optimization steps (Fig. S9) We find that when generating the reconstruction using EGG diffusion we obtain 1) comparable within-architecture generalization and 2) much better cross-architecture generalization (Fig. 6B). The EGG-generated images produce lower within architecture distances for 84% of the images and for 98% in the cross-architecture case.

We show examples of EGG diffusion and GD reconstructions in Fig. 6C. Qualitatively, the images optimized by EGG resemble the ground truth image much more faithfully than the GD images. More examples are available in the supplementary materials (Fig. S10) and for reconstructions from the attention model see figure S11. We furthermore reconstruct images from real neurons by minimizing the distance between the model responses and recorded average neural responses to the image (Fig. S12).

Human perceptual evaluation As shown in Cobos et al. [67], metrics like SSIM are not necessarily a good predictor of how well neuronal responses are reproduced in vivo. Therefore, we conducted a voluntary anonymous survey with 45 voluntary participants on 50 test images (Fig. S13). The participants were instructed to choose which image (GD optimized or EGG generated) was more similar to the ground truth image. Results show an 82.22% average preference for EGG-generated images (95% confidence interval [80.59%, 83.75%]; Wilson score interval). In 84% of the images, the EGG method was preferred more than 60% of the time (Fig. 6D).

Limitations While EGG diffusion on average performs better than GA, it does come with limitations. Firstly, It is important to note that the results shown in this study have not been verified in vivo. Moreover, while the energy scale provides additional flexibility, it is important to keep it mind that it is an additional hyperparameter that needs to be selected to obtain the desired results, for which careful controls are necessary before the method can be used for in-vivo verification. For example, if the energy scale is too high, our method can be unreliable, which we identified in 3 out of 90 cases where EGG diffusion failed to provide a satisfactory result with the Gaussian model (Fig. 7). Furthermore, the parameter value also does not necessarily represent the same value across energy functions. Another limitation is that the maximal number of steps to generate the sample is constricted by the pre-trained diffusion model, i.e. at most 1000 steps can be used. In addition, the encoding model needs to generalize to the manifold of the diffusion model. Finally, since neurons are strongly driven by contrast, encoding models often tend to push generated images to very high contrast values. To avoid this effect and make the model focus on the image content, we evaluate the energy guiding encoding model at a normalized image to eliminate the contrast direction from the guidance. This can be seen as an additional prior.

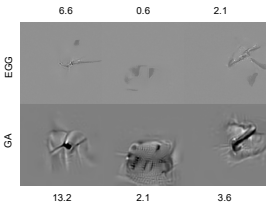


Figure 7: Examples of failure cases in comparison to the gradient ascent method. The text shows the predicted response rate by the within-architecture validation.

4 Discussion

In this study, we introduced a new model architecture, called the attention readout, for predicting the activity of neurons in macaque area V4 , which together with a fully data-driven convolutional core outperforms previous task-driven models. Furthermore, we propose a novel method for synthesizing images based on guiding diffusion models via energy functions (EGG). Our results indicate that EGG diffusion produces most exciting inputs (MEIs) which generalize better across architectures than the previous standard gradient ascent (GA) method. In addition, EGG diffusion significantly reduces compute time enabling larger-scale synthesis of visual stimuli. EGG diffusion is not limited to the generation of MEIs and, within the same framework, allows, among other characterizations, to 1) generate most exciting naturalistic images which approach the manifold of most activating images in the ImageNet database, and 2) reconstruct images from unit responses, which generalize better across architectures and qualitatively resemble more the original image than images obtained via regular gradient descent optimization. While the dataset we use for this study was recorded from the macaque visual cortex, it is in principle possible to use EGG for MEI generation and reconstructions with calcium imaging similar to the GA method on two-photon data in Walker et al. [17]. In fact, EGG can be applied to any modality that yields an encoding model and where a suitable diffusion model is available. More generally, EGG can be used whenever the "constraint" on a particular image can be phrased in terms of an energy function. In summary, EGG diffusion provides a flexible and powerful framework for studying coding properties of the visual system.

Acknowledgments

The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting Konstantin Willeke and Arne Nix. The authors also thank Mohammad Bashiri and Suhas Shirinvasan for their technical support and helpful discussions. The research was supported by the Cyber Valley Research Fund (AN, FHS). FHS is further supported by the German Federal Ministry of Education and Research (BMBF) via the Collaborative Research in Computational Neuroscience (CRCNS) (FKZ 01GQ2107), as well as the Collaborative Research Center (SFB 1233, Robust Vision). PP is supported by the German Federal Ministry for Economic Affairs and Climate Action (FKZ ZF4076506AW9). We also acknowledge support from the National Institute of Mental Health and National Institute of Neurological Disorders And Stroke under Award Number U19MH114830 and National Eye Institute award numbers R01 EY026927 and Core Grant for Vision Research T32-EY-002520-37 as well as the National Science Foundation Collaborative Research in Computational Neuroscience, USA with grant number IIS-2113173, Germany with FKZ: 01GQ2107. We thank our lab members, family, and friends for participating in the anonymous survey.

References

[1] D H Hubel and T N Wiesel. Receptive fields of single neurones in the cat's striate cortex. *J. Physiol.*, 148(3):574–591, October 1959.

[2] Charles F Cadieu, Ha Hong, Daniel L K Yamins, Nicolas Pinto, Diego Ardila, Ethan A Solomon, Najib J Majaj, and James J DiCarlo. Deep neural networks rival the representation of primate IT cortex for core visual object recognition. *PLoS Comput. Biol.*, 10(12):e1003963, December 2014.

[3] Daniel L K 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. *Proc. Natl. Acad. Sci. U. S. A.*, 111(23):8619–8624, June 2014.

[4] Santiago A Cadena, George H Denfield, Edgar Y Walker, Leon A Gatys, Andreas S Tolias, Matthias Bethge, and Alexander S Ecker. Deep convolutional models improve predictions of macaque V1 responses to natural images. *PLoS Comput. Biol.*, 15(4):e1006897, April 2019.

[5] Fabian H Sinz, Alexander S Ecker, Paul G Fahey, Edgar Y Walker, Erick Cobos, Emmanouil Froudarakis, Dimitri Yatsenko, Xaq Pitkow, Jacob Reimer, and Andreas S Tolias. Stimulus domain transfer in recurrent models for large scale cortical population prediction on video. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, pages 7199–7210, Red Hook, NY, USA, December 2018. Curran Associates Inc.

[6] Yimeng Zhang, Tai Sing Lee, Ming Li, Fang Liu, and Shiming Tang. Convolutional neural network models of V1 responses to complex patterns. *J. Comput. Neurosci.*, 46(1):33–54, February 2019.

[7] Lane T McIntosh, Niru Maheswaranathan, Aran Nayebi, Surya Ganguli, and Stephen A Baccus. Deep learning models of the retinal response to natural scenes. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pages 1369–1377, Red Hook, NY, USA, December 2016. Curran Associates Inc.

[8] David A Klindt, Alexander S Ecker, Thomas Euler, and Matthias Bethge. Neural system identification for large populations separating "what" and "where". November 2017.

[9] William F Kindel, Elijah D Christensen, and Joel Zylberberg. Using deep learning to probe the neural code for images in primary visual cortex. *J. Vis.*, 19(4):29, April 2019.

[10] Alexander S Ecker, Fabian H Sinz, Emmanouil Froudarakis, Paul G Fahey, Santiago A Cadena, Edgar Y Walker, Erick Cobos, Jacob Reimer, Andreas S Tolias, and Matthias Bethge. A rotation-equivariant convolutional neural network model of primary visual cortex. September 2018.

[11] Benjamin R Cowley and Jonathan W Pillow. High-contrast "gaudy" images improve the training of deep neural network models of visual cortex. June 2020.

[12] Max F Burg, Santiago A Cadena, George H Denfield, Edgar Y Walker, Andreas S Tolias, Matthias Bethge, and Alexander S Ecker. Learning divisive normalization in primary visual cortex. *PLoS Comput. Biol.*, 17(6):e1009028, June 2021.

[13] Eleanor Batty, Josh Merel, Nora Brackbill, Alexander Heitman, Alexander Sher, Alan Litke, E J Chichilnisky, and Liam Paninski. Multilayer recurrent network models of primate retinal ganglion cell responses. July 2022.

[14] Mohammad Bashiri, Edgar Walker, Konstantin-Klemens Lurz, Akshay Jagadish, Taliah Muhammad, Zhiwei Ding, Zhuokun Ding, Andreas Tolias, and Fabian Sinz. A flow-based latent state generative model of neural population responses to natural images. *Adv. Neural Inf. Process. Syst.*, 34:15801–15815, December 2021.

[15] Ján Antolík, Sonja B Hofer, James A Bednar, and Thomas D Mrsic-Flogel. Model constrained by visual hierarchy improves prediction of neural responses to natural scenes. *PLoS Comput. Biol.*, 12(6):e1004927, June 2016.

[16] Eric Y Wang, Paul G Fahey, Kayla Ponder, Zhuokun Ding, Andersen Chang, Taliah Muhammad, Saumil Patel, Zhiwei Ding, Dat Tran, Jiakun Fu, et al. Towards a foundation model of the mouse visual cortex. *bioRxiv*, 2023.

[17] Edgar Y Walker, Fabian H Sinz, Erick Cobos, Taliah Muhammad, Emmanouil Froudarakis, Paul G Fahey, Alexander S Ecker, Jacob Reimer, Xaq Pitkow, and Andreas S Tolias. Inception loops discover what excites neurons most using deep predictive models. *Nat. Neurosci.*, 22(12): 2060–2065, December 2019.

[18] Pouya Bashivan, Kohitij Kar, and James J DiCarlo. Neural population control via deep image synthesis. *Science*, 364(6439), 2019.

[19] Carlos R Ponce, Will Xiao, Peter F Schade, Till S Hartmann, Gabriel Kreiman, and Margaret S Livingstone. Evolving images for visual neurons using a deep generative network reveals coding principles and neuronal preferences. *Cell*, 177(4):999–1009.e10, 2019.

[20] Katrin Franke, Konstantin F Willeke, Kayla Ponder, Mario Galdamez, Na Zhou, Taliah Muhammad, Saumil Patel, Emmanouil Froudarakis, Jacob Reimer, Fabian H Sinz, and Andreas S Tolias. State-dependent pupil dilation rapidly shifts visual feature selectivity. *Nature*, 610(7930): 128–134, October 2022.

[21] Larissa Höfling, Klaudia P Szatko, Christian Behrens, Yongrong Qiu, David A Klindt, Zachary Jessen, Gregory W Schwartz, Matthias Bethge, Philipp Berens, Katrin Franke, Alexander S Ecker, and Thomas Euler. A chromatic feature detector in the retina signals visual context changes. December 2022.

[22] Jiakun Fu, Suhas Shrinivasan, Kayla Ponder, Taliah Muhammad, Zhuokun Ding, Eric Wang, Zhiwei Ding, Dat T Tran, Paul G Fahey, Stelios Papadopoulos, Saumil Patel, Jacob Reimer, Alexander S Ecker, Xaq Pitkow, Ralf M Haefner, Fabian H Sinz, Katrin Franke, and Andreas S Tolias. Pattern completion and disruption characterize contextual modulation in mouse visual cortex. March 2023.

[23] Zhiwei Ding, Dat T Tran, Kayla Ponder, Erick Cobos, Zhuokun Ding, Paul G Fahey, Eric Wang, Taliah Muhammad, Jiakun Fu, Santiago A Cadena, Stelios Papadopoulos, Saumil Patel, Katrin Franke, Jacob Reimer, Fabian H Sinz, Alexander S Ecker, Xaq Pitkow, and Andreas S Tolias. Bipartite invariance in mouse primary visual cortex. March 2023.

[24] Konstantin F Willeke, Kelli Restivo, Katrin Franke, Arne F Nix, Santiago A Cadena, Tori Shinn, Cate Nealley, Gabby Rodriguez, Saumil Patel, Alexander S Ecker, Fabian H Sinz, and Andreas S Tolias. Deep learning-driven characterization of single cell tuning in primate visual area V4 unveils topological organization. May 2023.

[25] Tirin Moore, Katherine M Armstrong, and Mazyar Fallah. Visuomotor origins of covert spatial attention. *Neuron*, 40(4):671–683, November 2003.

[26] A S Tolias, T Moore, S M Smirnakis, E J Tehovnik, A G Siapas, and P H Schiller. Eye movements modulate visual receptive fields of V4 neurons. *Neuron*, 29(3):757–767, March 2001.

[27] Till S Hartmann, Marc Zirnsak, Michael Marquis, Fred H Hamker, and Tirin Moore. Two types of receptive field dynamics in area v4 at the time of eye movements? *Frontiers in systems neuroscience*, 11:13, 2017.

[28] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2(11): e7, November 2017.

[29] Pouya Bashivan, Kohitij Kar, and James J DiCarlo. Neural population control via deep image synthesis. *Science*, 364(6439), May 2019.

[30] Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Brandon Tran, and Aleksander Madry. Adversarial robustness as a prior for learned representations. June 2019.

[31] Jenelle Feather, Guillaume Leclerc, Aleksander Mądry, and Josh H McDermott. Model metamers illuminate divergences between biological and artificial neural networks. May 2022.

[32] Yifei Ren and Pouya Bashivan. How well do models of visual cortex generalize to out of distribution samples? May 2023.

[33] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Adv. Neural Inf. Process. Syst.*, 33:6840–6851, 2020.

[34] Yu Takagi and Shinji Nishimoto. High-resolution image reconstruction with latent diffusion models from human brain activity. March 2023.

[35] Yizhuo Lu, Changde Du, Dianpeng Wang, and Huiguang He. MindDiffuser: Controlled image reconstruction from human brain activity with semantic and structural diffusion. March 2023.

[36] Laura N Driscoll, Lea Duncker, and Christopher D Harvey. Representational drift: Emerging theories for continual learning and experimental future directions. *Current Opinion in Neurobiology*, 76:102609, 2022.

[37] U. Guclu and M. A. J. van Gerven. Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. *Journal of Neuroscience*, 35(27):10005–10014, July 2015. doi: 10.1523/jneurosci.5023-14.2015. URL `https://doi.org/10.1523/jneurosci.5023-14.2015`.

[38] Daniel L K Yamins and James J DiCarlo. Using goal-driven deep learning models to understand sensory cortex. *Nat. Neurosci.*, 19(3):356–365, March 2016.

[39] Santiago A Cadena, Konstantin F Willeke, Kelli Restivo, George Denfield, Fabian H Sinz, Matthias Bethge, Andreas S Tolias, and Alexander S Ecker. Diverse task-driven modeling of macaque V4 reveals functional specialization towards semantic tasks. May 2022.

[40] Konstantin-Klemens Lurz, Mohammad Bashiri, Konstantin Willeke, Akshay K Jagadish, Eric Wang, Edgar Y Walker, Santiago A Cadena, Taliah Muhammad, Erick Cobos, Andreas S Tolias, Alexander S Ecker, and Fabian H Sinz. Generalization in data-driven models of primary visual cortex. April 2021.

[41] D A Klindt, A S Ecker, T Euler, and M Bethge. Neural system identification for large populations separating "what" and "where". In *Advances in Neural Information Processing Systems*, pages 4–6, 2017.

[42] F Sinz, A S Ecker, P Fahey, E Walker, E Cobos, E Froudarakis, D Yatsenko, X Pitkow, J Reimer, and A Tolias. Stimulus domain transfer in recurrent models for large scale cortical population prediction on video. In *Advances in Neural Information Processing Systems 31*. 2018.

[43] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. December 2015.

[44] Hadi Salman, Andrew Ilyas, Logan Engstrom, Ashish Kapoor, and Aleksander Madry. Do adversarially robust ImageNet models transfer better? July 2020.

[45] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. February 2015.

[46] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[47] Alex Graves. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*, 2013.

[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[49] Jingjing Xu, Xu Sun, Zhiyuan Zhang, Guangxiang Zhao, and Junyang Lin. Understanding and improving layer normalization. *Advances in Neural Information Processing Systems*, 32, 2019.

[50] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 2015.

[51] Jascha Sohl-Dickstein, Eric A Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. March 2015.

[52] Alex Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. February 2021.

[53] Prafulla Dhariwal and Alex Nichol. Diffusion models beat GANs on image synthesis. May 2021.

[54] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-Resolution image synthesis with latent diffusion models. December 2021.

[55] Jonathan Ho and Tim Salimans. Classifier-Free diffusion guidance. July 2022.

[56] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic Text-to-Image diffusion models with deep language understanding. May 2022.

[57] Nan Liu, Shuang Li, Yilun Du, Joshua B Tenenbaum, and Antonio Torralba. Learning to compose visual relations. November 2021.

[58] Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B Tenenbaum. Compositional visual generation with composable diffusion models. June 2022.

[59] Yilun Du, Conor Durkan, Robin Strudel, Joshua B Tenenbaum, Sander Dieleman, Rob Fergus, Jascha Sohl-Dickstein, Arnaud Doucet, and Will Grathwohl. Reduce, reuse, recycle: Compositional generation with Energy-Based diffusion models and MCMC. February 2023.

[60] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-Based generative modeling through stochastic differential equations. November 2020.

[61] Wei Li, Xue Xu, Xinyan Xiao, Jiachen Liu, Hu Yang, Guohao Li, Zhanpeng Wang, Zhifan Feng, Qiaoqiao She, Yajuan Lyu, and Hua Wu. UPainting: Unified Text-to-Image diffusion generation with cross-modal guidance. October 2022.

[62] Katherine Crowson. v-diffusion, 2021. URL `https://github.com/crowsonkb/v-diffusion-pytorch`.

[63] Katherine Crowson. Clip guided diffusion hq 256x256, 2021. URL `https://colab.research.google.com/drive/12a_Wrfi2_gwwAuN3VvMTwVMz9TfqctNj`.

[64] Zahra Kadkhodaie and Eero P Simoncelli. Solving linear inverse problems using the prior implicit in a denoiser. July 2020.

[65] Berthy T Feng, Jamie Smith, Michael Rubinstein, Huiwen Chang, Katherine L Bouman, and William T Freeman. Score-Based diffusion models as principled priors for inverse imaging. April 2023.

[66] Jiakun Fu, Konstantin F Willeke, Pawel A Pierzchlewicz, Taliah Muhammad, George H Denfield, Fabian Hubert Sinz, and Andreas S Tolias. Heterogeneous orientation tuning across Sub-Regions of receptive fields of V1 neurons in mice. February 2022.

[67] Erick Cobos, Taliah Muhammad, Paul G Fahey, Zhiwei Ding, Zhuokun Ding, Jacob Reimer, Fabian H Sinz, and Andreas S Tolias. It takes neurons to understand neurons: Digital twins of visual cortex synthesize neural metamers. December 2022.

[68] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, June 2009.

A Supplementary Material

A.1 Training Data

Electrophysiological data were acquired as broadband signal (0.5Hz-16kHz), from a pair of male rhesus macaque monkeys (*Macaca mulatta*), using 32 channel linear silicon probes (NeuroNexus V1x32-Edge-10mm-60-177). The data was spike-sorted, and single units were isolated based on unit stability, refractory periods, and channel principal component pairs. Visual stimuli were presented to the animals on a 16:9 widescreen HD LCD monitor at 100c̆m viewing distance. The animals were rewarded with juice if they maintained their gaze around a red fixation target throughout each trial. At the beginning of each recording session, the receptive fields (RFs) of the neurons were mapped in relation to a fixation target using sparse random dot stimuli, and the population RF was pulled towards the center of the screen by adjusting the fixation target. A collection of 24,075 images from ImageNet [68] was transformed into gray-scale and cropped to the central 420^2 px and had 8 bit intensity resolution. These images were presented as visual stimuli during standalone generation recordings of 1244 units and during closed-loop recordings of 82 units. For details on the closed loop paradigm, please refer to Willeke et al. [24].

A.2 Supplementary Experiments

A.2.1 Attention Model

Attention readout uses its ability to shift receptive fields Receptive fields of neurons in area V4 can shift before the onset of saccades, believed to be associated with attentional shifts [26]. We investigate whether the Attention model actually uses its ability to shift the receptive field depending on image content. We inspect the attention mask of the attention model. We compute the center of mass of the upper 5% percentile of each attention mask. We then compute the average distance between the center of masses across different images for each neuron. We plot the average distance against the test correlation of each neuron observing that the attention readout does perform shifts (Fig. S1a). We also show qualitative examples of the masks and the means in Fig. S1b.

Centered Kernel Alignment We computed CKA of the neural encodings across architectures between the Attention model and Gaussian model and within architecture between different seeds (e.g. Attention 1 and Attention 2 are models with the same architecture, but trained with different seeds). The CKA is computed between the predicted neuronal responses. We observe that the within-architecture similarity is very high (> 0.99) for both architectures and the cross architecture similarity is slightly lower, but also high (> 0.9) (Table 2). We expect such an outcome, since both architectures were trained to model the same neural representation.

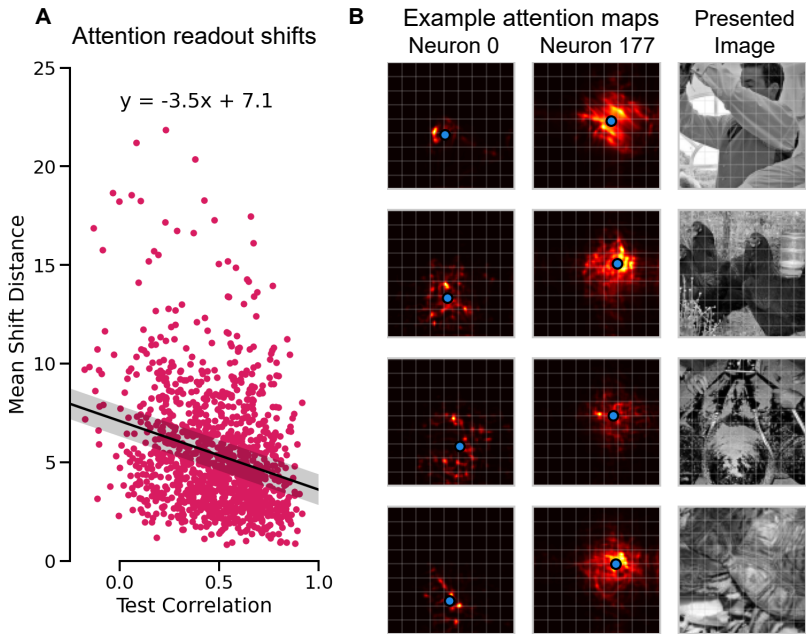
Model	Attention 1	Attention 2	Gaussian 1	Gaussian 2
Attention 1	1	0.9949	0.9133	0.9116
Attention 2	0.9949	1	0.9145	0.9129
Gaussian 1	0.9133	0.9145	1	0.9994
Gaussian 2	0.9116	0.9129	0.9994	1

Supplementary Table S1: Centered Kernel Alignment for the two architectures comparing across and within architectures.

A.2.2 MEI generation

Comparison of experimental setups For the GA optimization, we use the established method for generating MEIs that has been tested in vivo Walker et al. [17]. However, we perform a comparison study to show that the parameters chosen are selected to maximize the performance of the GA method. We rerun the MEI optimizations using the AdamW optimizer and find a significant decrease in performance in comparison to the SGD optimizer (r = 0.69). We also run the MEIs for 100 steps instead of 1000 and also find a performance decrease (r = 0.95) (Fig. S2).

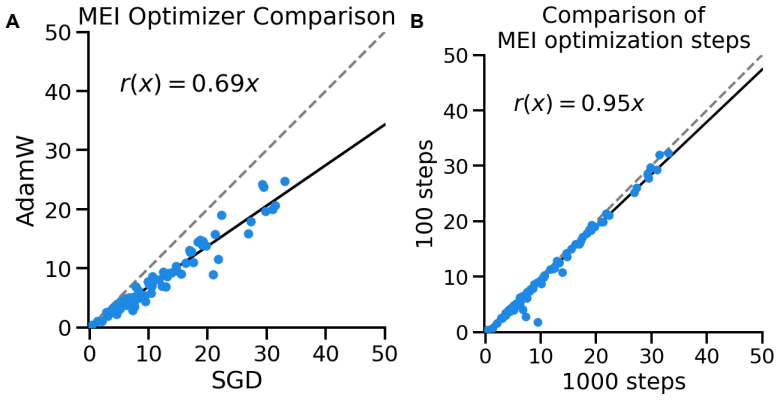
Generating MEIs in color The diffusion model generates color images, so in principle, it can generate color MEIs. We attach some examples (Fig. S4). Since the encoding models are trained



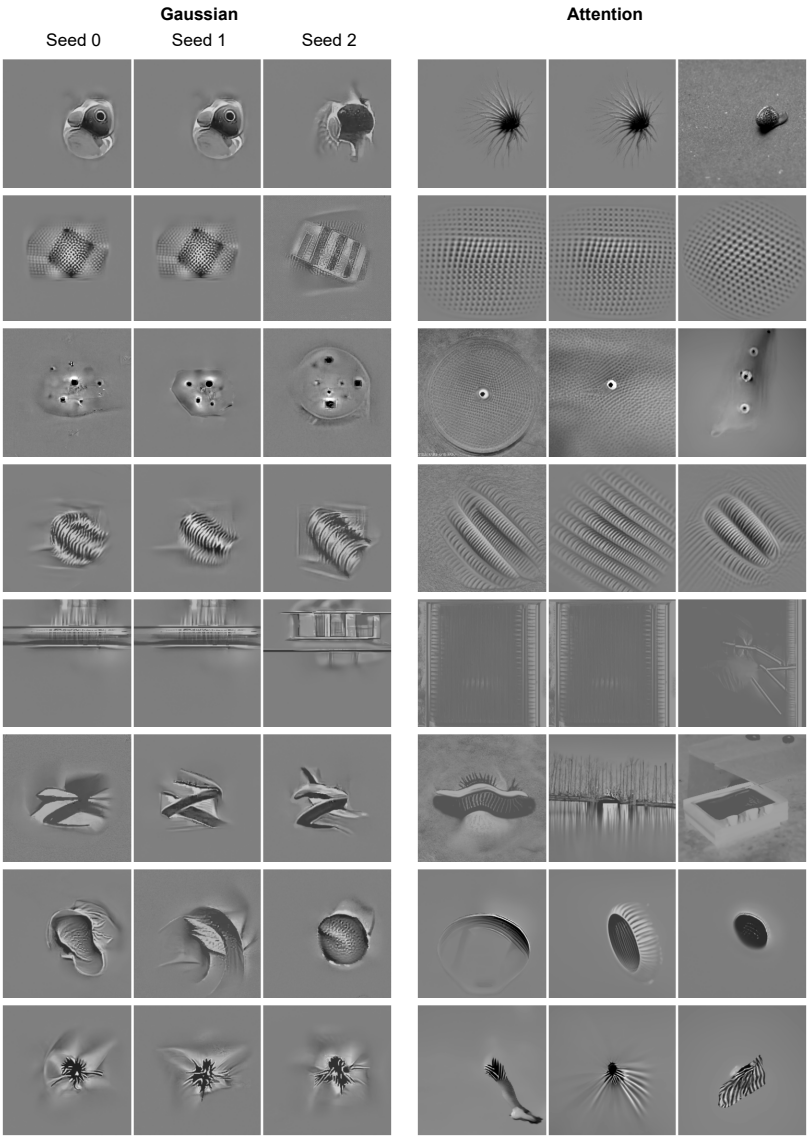
Supplementary Figure S1: **a**) Comparison of the mean shift distance between the center of mass for the attention masks against the test correlation of the neurons. **b**) Example attention maps of neurons responding to different neurons. The blue dot shows the center of mass. The examples show that the neuron shifts its center of mass.

on grayscale images because the animals only saw grayscale images the colors in these may not be meaningful. However, if one were to use color stimuli it would be possible to generate MEIs that are colored and potentially meaningful.

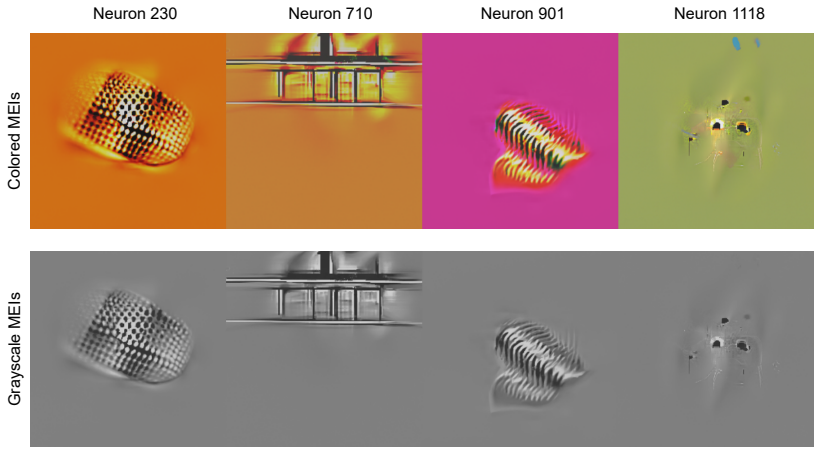
MENIs vs ImageNet search We compare the generated MEIs ($\lambda = 1$) to a standard approach for finding natural images for individual neurons. To that end, we perform a search across 100k images from the ImageNet dataset [68] to find the top-1 most activating image for a particular unit. We then compare the predicted activations of the top-1 ImageNet image and the generated MEIs ($\lambda = 1$) in the cross-architectures paradigm (Fig. S8). We find that the generated MENIs drive comparable activation to the top-1 ImageNet images. Like in the MEI generation paradigm, EGG can thus significantly speed up the search for activating natural images, as it does not need to search through millions of images.



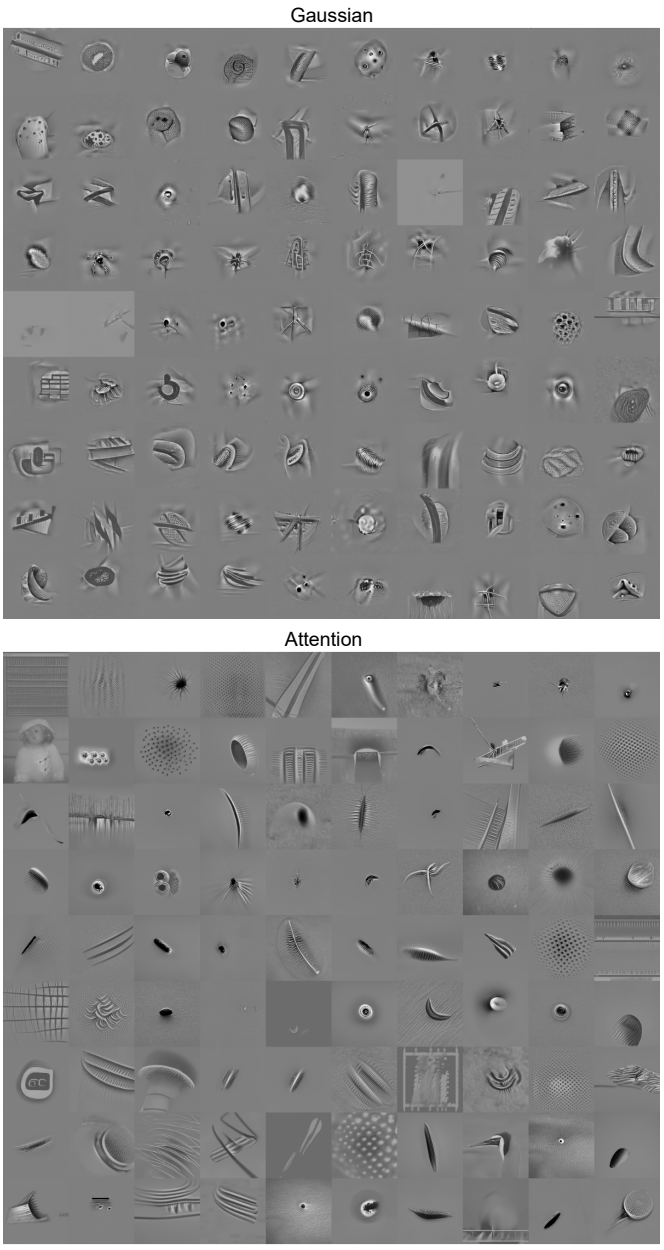
Supplementary Figure S2: Comparison of different experimental setups of MEI optimization using the GA method. **a**) Use of SGD vs AdamW optimizer. The SGD optimizer outperforms the AdamW optimizer on within architecture evaluation. **b**) Increasing the number of steps slightly decreases the within architecture performance.



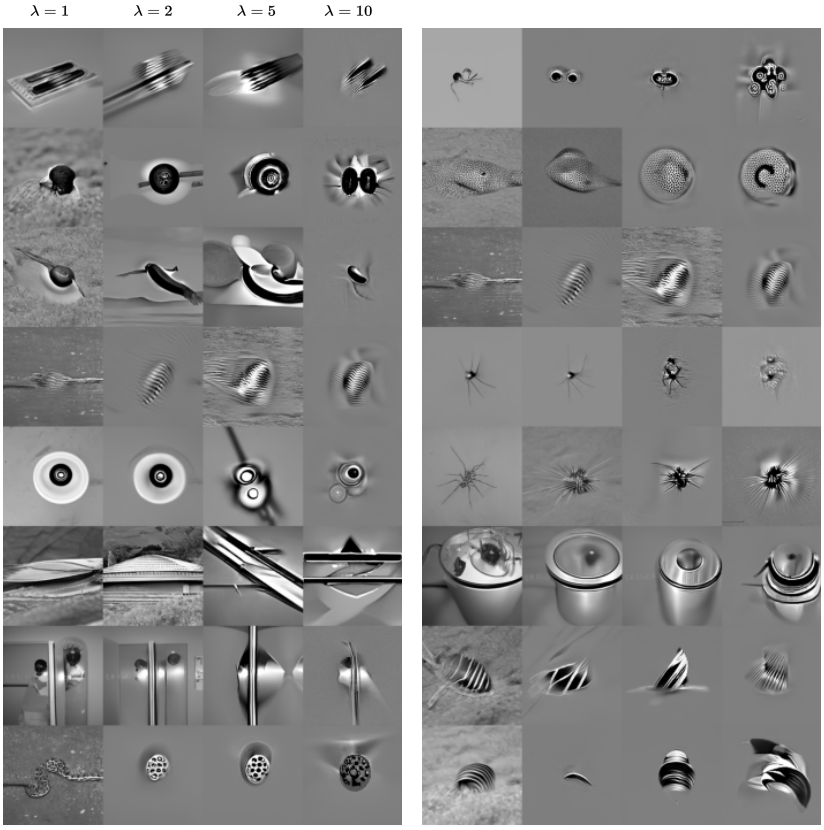
Supplementary Figure S3: Variability dependent on the seed used for generating MEIs in the Gaussian model and the Attention model. Each column represents a different seed and each row a different neuron. Results shown for the Gaussian and Attention models.



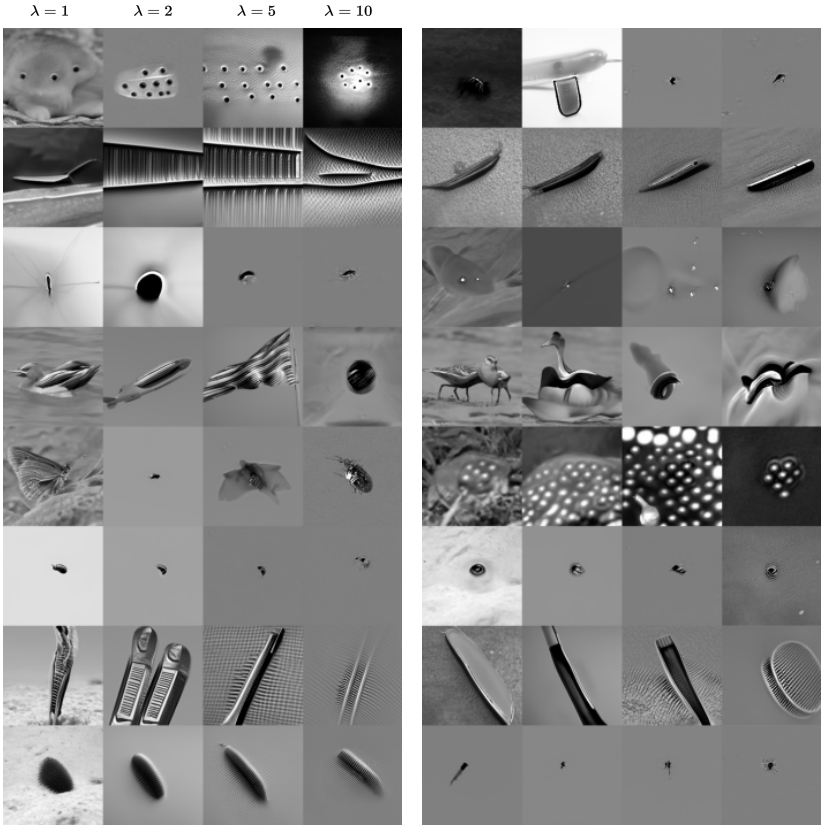
Supplementary Figure S4: Examples of MEIs in their original color version (before converting to grayscale). Top row is the direct output RGB images from the diffusion model, the bottom shows the grayscaled version. Each column corresponds to a different neuron.



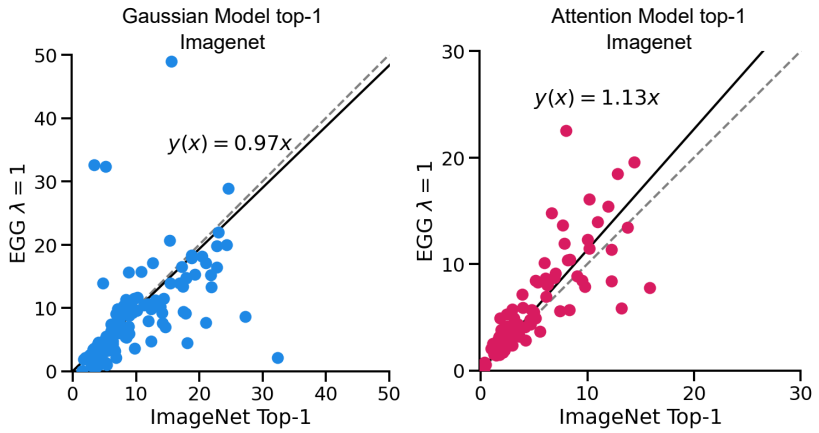
Supplementary Figure S5: Examples of MEIs generated using EGG for the Attention and Gaussian models.



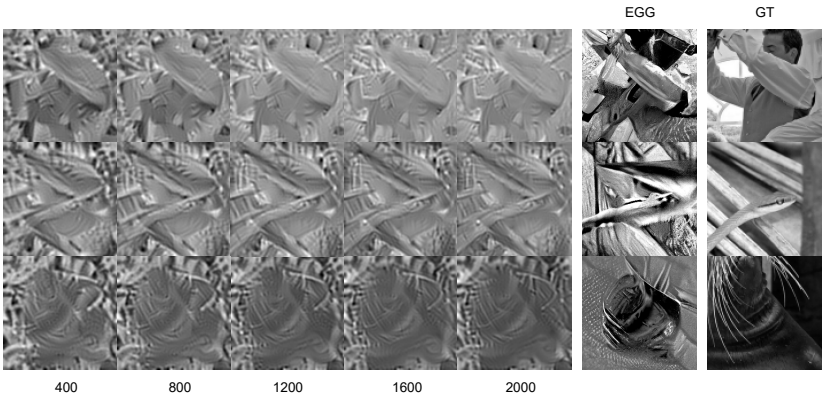
Supplementary Figure S6: Examples of images generated using EGG diffusion in the Monkey V4 with different energy scales $\lambda \in \{1, 2, 5, 10\}$. Generated for the Gaussian model. Units not matched with the images shown for the Attention model.



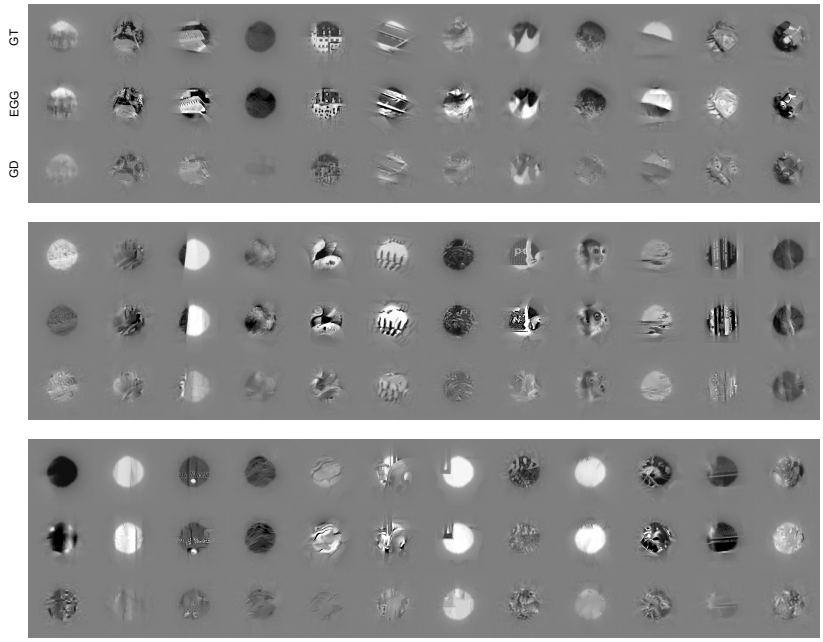
Supplementary Figure S7: Examples of images generated using EGG diffusion in the Monkey V4 with different energy scales $\lambda \in \{1, 2, 5, 10\}$. Generated for the Attention model. Units not matched with the images shown for the Gaussian model.



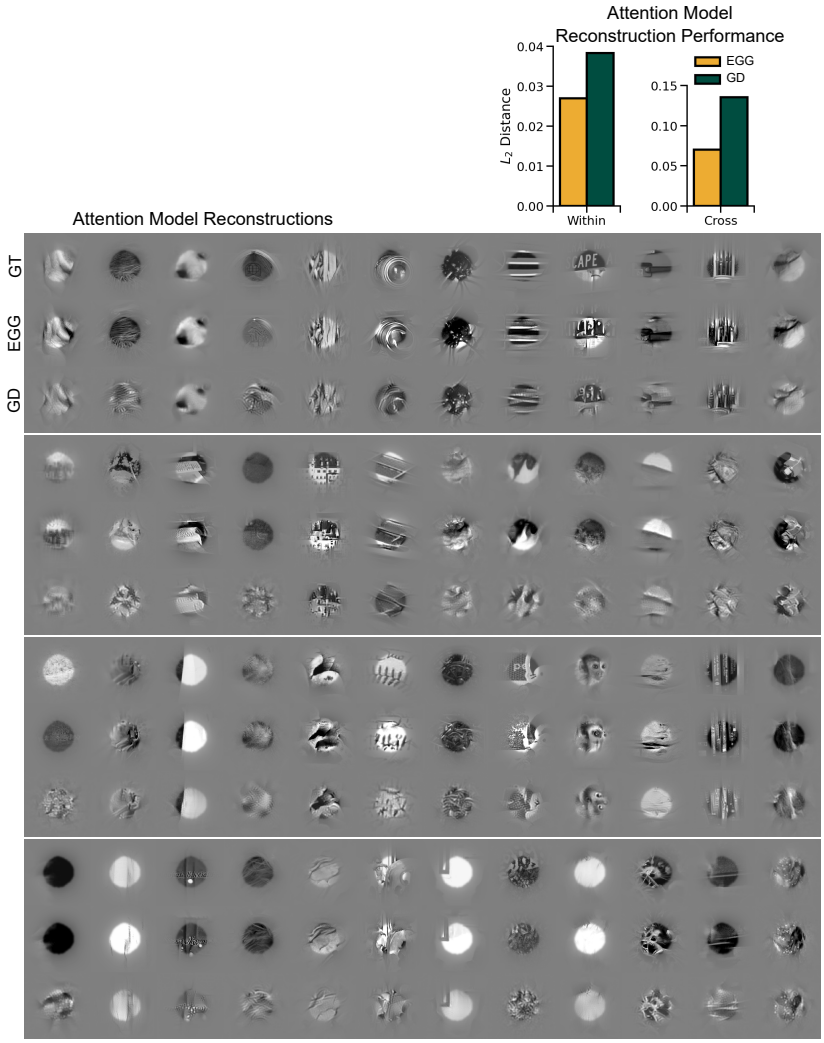
Supplementary Figure S8: Comparison of the MEIs $\lambda = 1$ activations to the top-1 most activating ImageNet images per neuron in the cross-architecture domain. Line fit obtained via Huber regression with $\varepsilon = 1.1$. In the left panel, three points at (11, 65), (9, 70), and (16, 120) are not shown for visualization purposes.



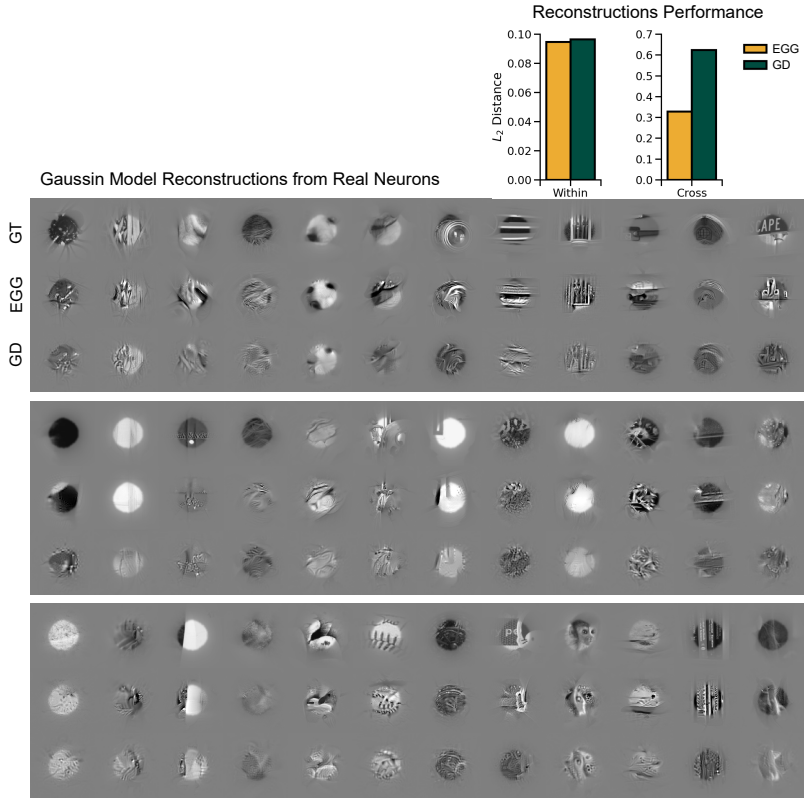
Supplementary Figure S9: Examples of reconstructions using GD across various training lengths. Increasing the training does not bring the generated image closer visually to the GT nor EGG.



Supplementary Figure S10: Reconstruction examples from the Gaussian model. Generated using EGG diffusion and gradient descent.



Supplementary Figure S11: Reconstructions from the Attention model. Top row in each panel is the ground truth image, middle is our EGG generated reconstruction and last row is the GD optimized reconstruction. Bar plot shows the performance of both EGG and GD in the within and cross architecture paradigms in terms of L_2 distance.



Supplementary Figure S12: Reconstructions from real neurons using the Gaussian model. Top row in each panel is the ground truth image, middle is our EGG generated reconstruction and last row is the GD optimized reconstruction. Bar plot shows the performance of both EGG and GD in the within and cross architecture paradigms in terms of L_2 distance.

Supplementary Figure S13: Setup for the human perceptual evaluation. The voluntary participants were presented with 50 images with the GT image always in the middle and the GD or EGG reconstructions were placed randomly on each side of the GT image. The participants were provided with the question "Which of the images looks more like the image in the center?". They were provided with context text: "In our study, we are reconstructing images from the brain activity. We have two methods to do so and we want to find out which one looks better to the human eye. Your participation is entirely voluntary, and you have the right to withdraw at any time without providing a reason. Please note that all responses will be kept confidential, and the data collected will be used solely for research purposes. Your identity will remain anonymous, and your personal information will not be disclosed to anyone."

Can Functional Transfer Methods Capture Simple Inductive Biases?

Arne F. Nix[1,2,†] **Suhas Shrinivasan**[2,3] **Edgar Y. Walker**[1,4,5] **Fabian H. Sinz**[1,2,‡]
[1] Institute for Bioinformatics and Medical Informatics, University of Tübingen
[2] Department of Machine Learning, Institute of Computer Science, University of Göttingen
[3] Graduate Center for Neurosciences, Biophysics, and Molecular Biosciences, University of Göttingen
[4] Department of Physiology and Biophysics, University of Washington
[5] UW Computational Neuroscience Center, University of Washington
[†] arne-fabian.nix@uni-tuebingen.de [‡] sinz@cs.uni-goettingen.de

Abstract

Transferring knowledge embedded in trained neural networks is a core problem in areas like model compression and continual learning. Among knowledge transfer approaches, functional transfer methods such as knowledge distillation and representational distance learning are particularly promising, since they allow for transferring knowledge across different architectures and tasks. Considering various characteristics of networks that are desirable to transfer, equivariance is a notable property that enables a network to capture valuable relationships in the data. We assess existing functional transfer methods on their ability to transfer equivariance and empirically show that they fail to even transfer shift equivariance, one of the simplest equivariances. Further theoretical analysis demonstrates that representational similarity methods, in fact, cannot guarantee the transfer of the intended equivariance. Motivated by these findings, we develop a novel transfer method that learns an equivariance model from a given teacher network and encourages the student network to acquire the same equivariance, via regularization. Experiments show that our method successfully transfers equivariance even in cases where highly restrictive methods, such as directly matching student and teacher representations, fail.[1]

[1]Code: https://github.com/sinzlab/orbit_transfer

Proceedings of the 25[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2022, Valencia, Spain. PMLR: Volume 151.

1 INTRODUCTION

With the rise of large and high-capacity models being trained on unprecedented amounts of data (Riquelme et al., 2021; Kolesnikov et al., 2020), the paradigm of transfer learning has grown in prominence (Zhuang et al., 2021). In transfer learning, the goal is to transfer useful functional properties learned by a *teacher*-model to a *student*-model. This is often realized by copying the teacher's parameters, or a subset thereof, to the student (weights-based transfer methods). However, copying the teacher's parameters is not feasible when the model architectures of student and teacher are different, for instance when the student has substantially fewer parameters (for the sake of efficiency) than the teacher (Hinton and Dean, 2015). Furthermore, not all of the teacher's parameters may be useful for the student. In such cases, *functional transfer* methods present an alternative to weights-based methods. Functional transfer methods rely on comparing and transferring functional properties, i.e., layer activation or outputs, for a given input. Transfer methods of this nature have applications in numerous areas like model compression (Bucilă et al., 2006; Hinton and Dean, 2015), continual learning (Pan et al., 2020; Titsias et al., 2019; Benjamin et al., 2019) or even neuroscience (Li et al., 2019).

Functional transfer methods exist in many variations. Some rely entirely on the network's final output, while others use activity within the network. Some methods require that the tasks shared by the teacher and the student be identical, while others only require that the network's hierarchical structure be somewhat relatable (McClure and Kriegeskorte, 2016). Regardless of differences, the goal remains the same—to transfer useful knowledge from teacher to student. This prompts us to question: *How effectively do existing functional transfer methods transfer useful knowledge from a teacher to a student?* Despite the significance of this question,

thus far there hardly exists any definite answer regarding functional transfer methods. For instance Abnar et al. (2020) investigate whether during knowledge distillation (a functional transfer method) the teacher's inductive biases are also reflected in its logit outputs, but not specifically whether the inductive biases are transferred to the student. One reason for the lack of answers may be that the kind of knowledge that is useful to transfer is hard to characterize. It could be knowledge that the teacher learned from its training data, such as feature extraction capabilities resultant from large-scale pre-training (Beyer et al., 2021). It could also be knowledge that is inherent to the teacher's network architecture, such as the inductive bias of shift equivariance in convolutional neural networks, consequently used to improve a student network that does not have this knowledge built-in (Touvron et al., 2020).

As this could entail a very wide range of properties that are potentially useful to transfer, we choose to focus our study on one specific family of properties that are mathematically characterized and have a strong impact on the generalization of a model – equivariance. We believe transferring equivariance to be the minimal ability any useful functional transfer method ought to possess. We will thus investigate the following question: *Can functional transfer methods effectively transfer equivariance properties between student and teacher?* The answer to this question turns out to be that it is surprisingly difficult to transfer equivariance properties, and that existing functional transfer methods fail to do so. Abnar et al. (2020) investigate transfer of equivariances empirically and found that knowledge distillation improved shift and scale invariance of the student, but as we show, that is only guaranteed under strong assumptions (as we discuss in 3), and other methods do not focus on transferring equivariance itself. Creager et al. (2020) develop a framework for domain invariant learning, i.e., encouraging a model to be invariant to environment and background changes. Zhou et al. (2020) show that via meta-learning, one can recover convolutional architectures from the data itself, which they achieve by learning parameter sharing patterns, as opposed to functional transfer methods.

To approach our question, we first discuss output-level functional transfer methods (Section 3) such as knowledge distillation, as they represent one of the most popular transfer methods. Although they are theoretically capable of transferring equivariance for the entire network, we show empirically that they do not deliver on that promise in practice, if, for instance, the student is very flexible. Based on this, we hypothesize that additional within-network restriction is necessary for the transfer to be successful. This leads us to investigate representation-level transfer methods (Section 4), i.e. functional transfer between different layers inside the network. Here we show empirically that except for one-to-one matching of hidden representations, none of the methods are capable of even transferring shift equivariance. A theoretical exploration of this observation reveals that many representational similarity methods, a subclass of representational transfer methods, are not restrictive enough to guarantee a transfer of the same equivariance that was present in the teacher.

For each method, we first introduce the method followed by empirical and theoretical analysis of equivariance transfer, before moving on to the next method. Finally, we introduce a novel method of functional transfer to enable equivariance transfer (Section 5). Our method successfully captures equivariance properties of the teacher and transfers it to the student.

2 PRELIMINARIES AND SETUP

The no-free-lunch theorem (Shalev-Shwartz and Ben-David, 2013) shows that models need to be constrained in some way to have a chance of good generalization. A constraint like that is called an *inductive bias*. Our goal is to develop a method of functional transfer that guarantees transfer of knowledge—specifically transfer of useful inductive biases—between two neural networks. Here, we focus on equivariance properties, which, as alluded to earlier, constitute fundamentally important and useful knowledge to transfer.

2.1 Equivariance

Many tasks contain useful *symmetries* in their data that can be exploited by models to improve generalization. One famous example that successfully leverages symmetries are convolutional neural networks (CNNs, Fukushima, 1980). CNNs encode the shift symmetries of natural images in their architecture. They do this by being *equivariant* to shifts, which means that a shift in the input to a CNN(-layer) will result in a related shift in its output. Other examples of equivariance in models can be found in many areas from natural language processing (Gordon et al., 2020) to molecular biology (Thiede et al., 2020).

To formalize equivariances, we extend the notation of Cohen (2021) to our setting of transfer learning. Thus we will be regarding equivariance in the context of symmetry groups. A simple example would be the set $\mathbb{Z}^2$ of integer shifts in a 2D pixel grid underlying an image. E.g. a group element $(m, n) \in \mathbb{Z}^2$ corresponds to a shift of all pixels by m positions along the vertical and n positions along the horizontal axis. With addition $(m, n) + (p, q) = (m + p, n + q)$ as the group operation, negative shifts $(m, n) + (-m, -n) = (0, 0)$ as the inverse for each group element and $(0, 0)$ as the identity element, we can show that $\mathbb{Z}^2$ is indeed a group.

Arne F. Nix[1,2,†], Suhas Shrinivasan[2,3], Edgar Y. Walker[1,4,5], Fabian H. Sinz[1,2,‡]

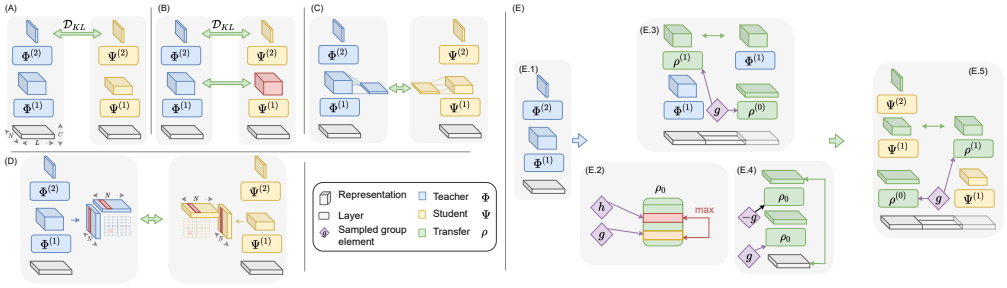


Figure 1: Overview of the transfer methods discussed in this paper: Knowledge distillation (A, Section 3), Direct matching (B, Section 4), Attention transfer (C, Section 4.1), RDL (D, Section 4) and Orbit transfer (E, Section 5).

We define a neural network as a composition of functions $\Phi = \Phi^{(L)} \circ ... \circ \Phi^{(1)}$ with layers $\Phi^{(l)} : \mathcal{X}^{(l-1)} \to \mathcal{X}^{(l)}$. Each individual layer $\Phi^{(l)}$ acts on an input or feature space $\mathcal{X}^{(l-1)}$ and projects to a feature space $\mathcal{X}^{(l)}$ that serves as input for the next layer. We further focus on symmetries generated by a group G. A group element $g \in G$ acts on an input element x via a *linear representation* ρ_g. Wherever it is clear from the context, we will leave the dependency on g implicit to simplify notation. If a function—or layer $\Phi^{(l)}$—is G-equivariant then its output will have a corresponding action $\rho^{(l)}$, that leads to the same results as transforming the input:

$$\rho_g^{(l)} \Phi^{(l)}(x) = \Phi^{(l)}(\rho_g^{(l-1)} x) \tag{1}$$

for all $x \in \mathcal{X}_t^{(l-1)}$ and any group element $g \in G$.

Known equivariance properties are useful as they allow parameter sharing in the network (Cohen, 2021). However, it is generally hard to discover these symmetries from data alone, since it may require a lot of data. Hence it is beneficial if we could transfer them from a extensively trained teacher network to a student. To formalize the transfer, we refer to the teacher- and student-network as Φ and Ψ, respectively. The corresponding layers are then defined by $\Phi^{(l)} : \mathcal{X}_t^{(l-1)} \to \mathcal{X}_t^{(l)}$ and $\Psi^{(l)} : \mathcal{X}_s^{(l-1)} \to \mathcal{X}_l^{(s)}$. We assume that the student network differs in architecture from its teacher[2] and is not necessarily G-equivariant by default, but that it is expressive enough to learn equivariance with the right guidance. This means generally that the student has more capacity, or is less constrained. In other words the student has less inductive bias than the teacher model. This is important since the aim of the experiments is to see if functional transfer would be strong enough to constrain, or transfer inductive bias, to an over-parameterized model that inherently lacks it. In practice, we achieve this by combining the standard cross-entropy loss $\mathcal{L}_{\text{CE}}$ with a transfer term Ω. Therefore the final loss will be: $\mathcal{L} = (1-\gamma)\mathcal{L}_{\text{CE}} + \gamma\Omega$. The two components are combined by an interpolation weight $\gamma \in [0, 1]$ which is treated as a hyperparameter.

[2]We assume the number of layers is identical to simplify the derivations. Since layers are not necessarily expected to be linear, we can always get to the same number of layers by adding identity layers or by summarizing multiple layers in one.

2.2 Analysis Setup

An important aspect of both the empirical and analytical settings we address is the extent to which transformed group elements are present in the training data. Formally, the set of data points $\{\rho_g x : g \in G\}$ that can be reached from a single data point x is called *orbit*. As we will see later, the extent to which entire orbits are in the training data will affect the generalization performance of the student. For our theoretical analysis we generally consider the case of unlimited training data—in other words data that contains all orbit elements of the transformation we aim to transfer.

To empirically investigate the transfer abilities of the various methods we analyze in this work, we construct a simple experimental setup on the MNIST-1D (Greydanus, 2020) task. This task generates 4000 training and 1000 test inputs, which are 40-dimensional vectors that are procedurally generated based on templates inspired by the original MNIST digits. The synthetic generation process allows fine-grained control over various properties of the resulting samples. For our purposes importantly, we can control the range of orbits range present in the training data. This is done through a "shift limit" s that partitions the set of shifts for which we generate an MNIST-1D variant with random shifts into two sets: *seen shifts* $[0, s]$ and *unseen shifts* $(s, 39]$. Our training data consists of *seen shifts*. We can then test the resulting model on three different ranges of shift values applied to the test data: *seen shifts* $[0, s]$, *unseen shifts* $[s, 39]$, and *all shifts* $[0, 39]$. This setup allows us to verify whether a network has learned shift equivariance and generalizes well to unseen shifts. Since we aim to create a very simple evaluation task, we chose teacher and student networks such that their

hidden layer outputs have the same size. Both have two hidden layers and a spatial max-pooling operation after the last layer to obtain the class predictions. The two networks only differ in the nature of the first two layers, where the teacher network uses convolutions, the student network instead uses fully connected layers (more flexible and less inductive bias). More details on the architecture and training setup can be found in the supplementary material. To fully explore the abilities of each transfer method, we first perform a hyper-parameter search for each on the setting with $s = 30$. Since we are mainly interested in generalization to unseen shifts, not images, we decided to select the best performing set of hyper-parameters on the image test set with seen shifts. The selected model is then trained and evaluated with $s = 0, 10, 20, 30, 40$ to explore the transfer abilities under different data conditions. As we introduce more shifts in the training data, the *seen shifts* test set will also contain those shifts. This makes it increasingly difficult to generalize for models without shift equivariance. As expected, the student network trained without any transfer knowledge degrades on seen shifts as training progresses (see Figure 2). Analogously, the performance on unseen shifts stays consistently bad until $s = 40$ where all shifts are seen.

3 OUTPUT-LEVEL TRANSFER

The most well-known methods for functional transfer fall in the category of output-level transfer methods. These methods aim to transfer the entire network function Φ by matching the function values directly at the final layer, i.e. $\Phi(x) = \Psi(x) \, \forall x$ in the case of infinite data. For classification, this is known as *knowledge distillation (KD)* (Hinton and Dean, 2015) which transfers Φ by minimizing the Kullback–Leibler divergence between the output distributions given by the softmax output of teacher and student network respectively (see Figure 1.A). Thus the transfer regularization would be $\Omega = p_t(x) \log \frac{p_t(x)}{p_s(x)}$ with softmax distributions $p_t(x)_i = \frac{\exp(\Phi(x)_i/\tau)}{\sum_j \exp(\Phi(x)_j/\tau)}$ and $p_s(x)_i = \frac{\exp(\Psi(x)_i/\tau)}{\sum_j \exp(\Psi(x)_j/\tau)}$. The corresponding transfer method for regression tasks would be *functional distance regularization* (Benjamin et al., 2019), which minimizes the euclidean distance between function values of student and teacher networks.

Can KD transfer equivariance? Abnar et al. (2020) investigated this question empirically and found that KD improved shift and scale invariance of fully-connected student model when they trained it with KD from the outputs of a scale and shift invariant teacher. At first glance, these results are expected: a student that is trained with KD will obtain the teacher's equivariance properties if the training is successful, i.e. if the function is matched perfectly. However, this is only guaranteed if the optimization leads to zero loss and the training happens in the limit of infinite data. It is unclear, however what happens if this is not the case. One scenario could be that the optimization yields zero loss, i.e. the teacher function is matched perfectly, but on the limited training data which might not contain all orbits of the symmetry group. Such a solution will likely not generalize well.

Our experimental results on MNIST1D verify this intuition. KD generally will not improve over the baseline student model when evaluated on unseen shifts (see Figure 2). We hypothesize that KD, by only regularizing on the final outputs, is simply not restrictive enough for this task.

4 REPRESENTATION-LEVEL TRANSFER

As we have just seen, it is hard to transfer equivariance on the level of final outputs and we hypothesized that this due to the lack of constraints on the representations within the network. One approach for this problem would be to directly transfer on the level of hidden representations, which we study in this section.

Can direct matching of internal representations transfer equivariance? *Direct matching* of the representations of the teacher and the student networks, e.g. by minimizing the mean-squared error between the teacher and student representations (see Figure 1.B), will likely be an effective way to transfer at this level.

To confirm this experimentally, we minimize the mean-squared distance on the internal layers' outputs and add a KD objective on the final layer. This approach works as suspected. Figure 2 shows a nearly perfect performance on unseen shifts as long as a few orbit elements are contained in the training data. However, making the problem slightly harder by replacing the student's final pooling operation with a linear projection while keeping the teacher the same reveals issues similar to KD as we no longer see any improvement over the student baseline in this scenario (see Figure 3).

Although we have just shown that direct matching is successful in transferring equivariance, a direct comparison of teacher and student representations requires the hidden representations to be of the same shapes. Enforcing this on all levels of the network would mean restricting the student network architecture to conform to that of the teacher, which is contrary to the goals of functional transfer methods. We thus explore alternative solutions to representation-level transfer that can be applied without requiring the same hidden layer shapes between student and teacher.

Arne F. Nix[1,2,†], **Suhas Shrinivasan**[2,3], **Edgar Y. Walker**[1,4,5], **Fabian H. Sinz**[1,2,‡]

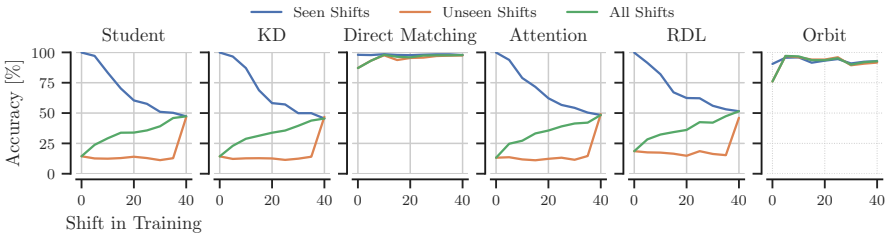


Figure 2: Test performance after transfer for a student with max pooling as final layer.

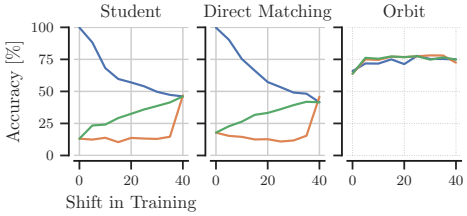


Figure 3: Test performance after transfer for a student with a linear final layer.

4.1 Attention Transfer

CNNs and related architectures generally maintain a three-dimensional structure throughout the outputs of most of their layers. This gives hidden activations the semantic interpretation of width, height and channel dimensions. *Attention transfer* (Zagoruyko and Komodakis, 2017) leverages this and the fact that layers from comparable processing steps usually have comparable spatial dimensions and only differ in the channel dimension. Slightly differing spatial dimensions can be aligned by up- or down-sampling, and the channel dimension is pooled by summation or maximum selection to extract an "attention map" that can be matched between teacher and student (see Figure 1.C).

$$\Omega^{(l)} = \| \frac{A_t^{(l)}(x)}{\|A_t^{(l)}(x)\|_2} - \frac{A_t^{(l)}(x)}{\|A_t^{(l)}(x)\|_2} \|_2^2$$

whith $A_t^{(l)}(x) = \sum_{c=1}^{C_s^{(l)}} |\Phi^{(l)}(x)_c|$ and $A_s^{(l)}$ defined analogously, channel size $C_s^{(l)}$ and $C_t^{(l)}$ for student and teacher layers respectively.

Attention transfer aligns the spatial dimensions between student and teacher. This should theoretically be beneficial when transferring equivariances to spatial effects like shift. However, our results show that that in practice this is not the case (Figure 2). Attention transfer fails to capture the shift equivariance in our experiment. The attention method has a potential drawback that could both be responsible for this outcome. Collapsing the channel dimension is certainly not a lossless operation and may likely hide information important for equivariance.

4.2 Representational Similarity Transfer

The problem of comparing representations of two networks with distinct architectures is of wider interest outside of transfer learning. Unsurprisingly, there exists a broad range of methods designed to compare neural network representations (Kornblith et al., 2019). For these methods, the general idea is to not match the two networks on the individual representations but to consider the representations for an entire batch, i.e. consider the representation as matrices $\mathbf{\Phi}^{(l)} = [\Phi^{(:l)}(x_1), \dots, \Phi^{(:l)}(x_N)]^\top$ and $\mathbf{\Psi}^{(l)} = [\Psi^{(:l)}(x_1), \dots, \Psi^{(:l)}(x_N)]^\top$ for a batch of inputs $x_1, \dots, x_N \in \mathcal{X}^{(0)}$. Note that here we treat $\Phi^{(:l)}$ and $\Psi^{(:l)}$ as functions on the input features, which means that we consider the composition of all layers up to l as one function. Then a comparison can be done along the number of samples N, providing a similarity for general network representations even if $\mathcal{X}_t^{(l+1)} \neq \mathcal{X}_s^{(l+1)}$. Maximizing this similarity (or equivalently minimizing the distance between two representations) can thus be used as an objective for functional transfer.

Representational Distance Learning A method following this principle is *representational distance learning* (RDL, McClure and Kriegeskorte, 2016). Here the representational distance is measured through comparison of *representational dissimilarity matrices* (RDM). These matrices are essentially Gram matrices of the batch representations $\mathbf{\Phi}^{(l)}$ (with additional normalization). Thus each entry (n, m) in the RDM shows the dissimilarity of samples x_n and x_m w.r.t. representation $\Phi^{(:l)}$. Such RDMs are computed for student and teacher respectively and used to compute representation distance by minimizing the Frobenius norm between the two.

$$\Omega^{(l)} = \frac{1}{N^2} \| \mathbf{\Phi}^{(l)} \mathbf{\Phi}^{(l)\top} - \mathbf{\Psi}^{(l)} \mathbf{\Psi}^{(l)\top} \|_F^2$$

This approach could be used to enforce representations similar to a teacher on any layer of a neural network without restricting either network in any way. To

ease notation, we will drop the layer index l for the remainder of this section.

However, this method does not match the individual responses directly, which raises questions as to how powerful and accurate the transfer will be. For RDL, an optimal solution has to fulfill $\mathbf{\Psi}\mathbf{\Psi}^\top = \mathbf{\Phi}\mathbf{\Phi}^\top$. However, $\mathbf{\Psi}\mathbf{\Psi}^\top = \mathbf{\Phi}\mathbf{\Phi}^\top$ if and only if $\mathbf{\Psi} = Q\mathbf{\Phi}$ with $Q \in SO(N)$ (Co, 2013, theorem 7.3.11). This means that RDL matches the teacher representation up to orthogonal transformations. In the following, we will show that this is not enough to guarantee the transfer of equivariance properties.

Alternative methods that follow the same principle as RDL to quantify representational similarity are summarized and compared in Kornblith et al. (2019). In principle all of them can be utilized to formulate a functional transfer objective similar to RDL. Nevertheless, most representational similarity methods have been shown to be invariant to orthogonal transformations (Kornblith et al., 2019). This means that a global optimum found by these methods will also have the property $\mathbf{\Psi} = Q\mathbf{\Phi}$ for $Q \in SO(N)$.

Can representational similarity methods transfer equivariance? Following the same methodology as before, we evaluate the transfer abilities of RDL. Similar to previous methods it fails in transferring shift equivariance (see Figure 2). In the following, we theoretically investigate why that is the case.

As we have shown above, functional transfer methods that rely on representational similarity are invariant to orthogonal transformations. That means that an optimal solution found by these methods can only restrict the student representations to match the teacher representations up to orthogonal transformation, i.e. $\mathbf{\Psi} = Q\mathbf{\Phi}$ for $Q \in SO(N)$. In the limit of infinite data, this also implies that the functions are equal up to the same orthogonal transformation, which we will denote (slight abuse of notation) as $\Psi = Q\Phi$. In this case the following theorem holds:

Lemma 1. *Given two representations Φ and Ψ with relationship $\Psi = Q\Phi$ for some orthogonal Q, the following holds: Φ is equivariant w.r.t. group representation $(\rho^{(0)}, \rho^{(1)})$ if and only if Ψ is equivariant w.r.t. group representation $(\rho^{(0)}, Q\rho^{(1)}Q^\top)$.*

Proof of Lemma 1. We first prove the forward direction, i.e. we show that if Φ is G-equivariant w.r.t. group representations $(\rho^{(0)}, \rho^{(1)})$, then Ψ is G-equivariant w.r.t. $(\rho^{(0)}, Q\rho^{(1)}Q^\top)$.

$$\begin{aligned}\Psi(\rho^{(0)}x) &= Q\Phi(\rho^{(0)}x) = Q\rho^{(1)}\Phi(x)\\ &= Q\rho^{(1)}(Q^\top Q)\Phi(x)\\ &= (Q\rho^{(1)}Q^\top)\Psi(x)\end{aligned}$$

Here we first use the fact that $\Psi = Q\Phi$, then we exploit the orthogonality of Q and the equivariance property of Φ w.r.t. $\rho^{(1)}$. Finally, we use the definition of Ψ a second time to get back from $Q\Phi$ to Ψ.

The backward direction the symmetric nature of Lemma 1 which lets us exploit the fact that $\Psi = Q\Phi$ entails $\Phi = \tilde{Q}\Psi$ where $Q \in SO(N)$ and $\tilde{Q} = Q^\top$. Then from the forward direction of Lemma 1 it follows that, Ψ is equivariant w.r.t. $(\rho^{(0)}, Q\rho^{(1)}Q^T)$ yields that Φ is equivariant w.r.t $(\rho^{(0)}, \tilde{Q}Q\rho^{(1)}Q^\top\tilde{Q}^\top) = (\rho^{(0)}, \rho^{(1)})$. □

The consequence of Lemma 1 is that representational similarity methods and other methods that can not ensure an exact matching of the teacher representations will not guarantee a transfer of equivariance properties w.r.t. the same group representation on the output. As we showed, the student representation will be equivariant w.r.t. the transformed group representation $Q\rho^{(1)}Q^\top$. Thus, RDL does transfer equivariance in the limit of infinite data, but might end up with a different "global" linear representation of the group at the final layer which might not aid generalization in the same way as the teacher representation itself. In particular the possible orthogonal transform Q will destroy the shift invariance of the last max-pooling layer since the supremum norm is not invariant under rotation. In this sense, the learned linear representation of the group does not fit to the expected input representation of the max-pooling layer which destroys its invariance property.

5 ORBIT MODEL TRANSFER

As we have seen in Section 3, output-level functional transfer from an equivariant teacher, such as performing KD, does not necessarily transfer equivariance to the student. However, we have also seen (Section 4) that trying to enforce similarity within the network, i.e. on the level of representations, is also insufficient. Such representational similarity methods are theoretically capable of enforcing an equivariance property on the student, but they cannot restrict the exact nature of that equivariance enough to guarantee a successful transfer.

These findings have revealed that matching the function of the entire network or even that of individual layers is too broad a task to reliably transfer specific equivariances. Hence we hypothesize that decoupling the equivariance property from the function is the issue.

We took this problem as inspiration and leveraged the fact that the problem definition for equivariance transfer is well-defined. For a transfer to be successful, the student has to fulfill the equation $\Psi(\rho_0 x) = \rho_1(\Psi(x))$ after training. Therefore, we propose a new approach

Arne F. Nix[1,2,†], Suhas Shrinivasan[2,3], Edgar Y. Walker[1,4,5], Fabian H. Sinz[1,2,‡]

where we directly learn the group representation ρ that the teacher is equivariant to and encourage the same equivariance in the student network.

5.1 Method

Our approach separates the transfer process into two steps. First, we learn a model of the equivariance throughout the teacher network, and then we use this model to regularize the student network.

Learning the equivariance from the teacher The idea for capturing the teacher's equivariance is to learn a model ρ that fulfills $\Phi(\rho_g^{(0)}x) = \rho_g^{(1)}\Phi(x)$ for any given g and x, i.e. a model of the group representation that Φ is equivariant to. To find such a model, we freeze the teacher network Φ and minimize the following objective for a given g and x:

$$\mathcal{L}_{\text{equiv}} = \frac{1}{H}\|\Phi(\rho_g^{(0)}x) - \rho_g^{(1)}\Phi(x)\|_2^2 \tag{2}$$

for hidden size H. This minimizes the distance between the layer representation with the group operation applied on the input or the layer's output.

One can see that the "true" equivariance representation that Φ is equivariant to would minimize this objective. However, at the same time, a trivial solution where ρ simply learns an identity operation on $\mathcal{X}$ for every g would also achieve the same result. To prevent this, we additionally minimize the absolute cosine dissimilarity between all representations, i.e. the kernels, of distinct group elements:

$$\mathcal{L}_{\text{group}} = \frac{1}{|G|\cdot(|G|-1)}\sum_{g\in G}\sum_{h\in G\setminus\{g\}}|\cos(\rho_g^{(0)},\rho_h^{(0)})| \tag{3}$$

Finally, we want to encourage a group structure in ρ and thus we add the following objective to encourage ρ to be invertible:

$$\mathcal{L}_{\text{inv}} = \|\rho_{-g}^{(0)}\rho_g^{(0)}x - x\|_2^2 \tag{4}$$

The parameters of the orbit model ρ are optimized to minimize a sum of all three objectives and the standard cross-entropy loss $\mathcal{L}_{\text{CE}}$ on the transformed inputs:

$$\min_{\rho}\gamma_{\text{CE}}\mathcal{L}_{\text{CE}} + \gamma_{\text{equiv}}\mathcal{L}_{\text{eqiv}} + \gamma_{\text{group}}\mathcal{L}_{\text{group}} + \gamma_{\text{inv}}\mathcal{L}_{\text{inv}} \tag{5}$$

with weights $\gamma_{\text{CE}}, \gamma_{\text{equiv}}, \gamma_{\text{group}}, \gamma_{\text{inv}} \in \mathbb{R}^+$.

Modeling group representations through convolutions So far, we have described an objective function to extract the equivariance from a given teacher network. This method is generally agnostic to the choice of group representation model ρ, nevertheless this decision is crucial for the effectiveness of the transfer.

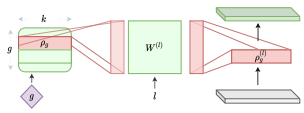


Figure 4: The group representation model first selects a filter of size k based on the group element g, then it applies the linear transformation $W^{(l)}$ to finally get the group representation that can be applied on the input x to get $\rho_g^{(l)}x$.

The group representation model needs to be powerful enough to capture the equivariance as well as the change of group representation throughout the depth of the network. At the same time, we need a model that is flexible enough to be applied on all layers of both teacher and student. To allow this flexibility, we decided to use convolution to model the group operation on every layer. Thus, for any given layer l and group element g, our model needs to provide a convolution filter that can be applied to the input (after padding it to preserve the size). A naive implementation would learn a separate filter for each group element and layer, which would not only require a lot of parameters, but also ignores the connectedness that group representations of the same g can have throughout an equivariant network. We decided to leverage this quality by factorizing the parameterization. Our model therefore learns one filter of size k per group element as well as L linear projections of size $k \times k$. To obtain the group operation $\rho_g^{(l)}$ the lth linear projections $W^{(l)}$ is applied to the filter corresponding to group element g. The group operation can then be applied to the input to get its group representation $\rho_g^{(l)}x$. The entire process is illustrated in Figure 4.

Modeling group representations as affine transformation on the coordinate space To demonstrate the flexibility of our transfer learning framework, we propose an alternative, more general, way of modeling group representations. For this, we parameterize a 3D transformation directly by learning an affine transformation matrix for each g and l. This $\rho_g^{(l)} \in \mathbb{R}^{3\times4}$ determines the transformation of a 3D input feature map, akin to spatial transformer networks (Jaderberg et al., 2015). With a group representation modeled in this way, Orbit can, in principle, transfer equivariance to any affine transformation across both spatial dimensions and the channel dimension.

Training the student to have the same equivariance Once the group representation model is learned, applying it in the student training is straight-forward. We simply use the same objective that we used for training the group representation model on the teacher, but here we freeze the group representation model instead.

	CNN → MLP		ResNet18 → ViT	
Method	Centered	Translated	Centered	Translated
Teacher	99.0	93.4	99.6	90.3
Student	98.5	35.7	98.6	37.3
+ Augment	54.3	97.0	56.5	97.4
KD	**98.8**	41.1	98.8	41.2
Attention	98.4	31.9	—	—
RDL	98.6	31.9	**99.3**	59.6
Orbit	**98.8**	**95.2**	98.4	**84.0**

Table 1: MNIST (column 1 and 3) and MNIST-C (column 2 and 4) test results for four different transfer methods. Left two columns show the transfer results from a small CNN teacher to an MLP student. The right columns show analogous experiments between a ResNet18 teacher and a small ViT student. The best performing transfer is shown in bold for each column.

This means can ignore the objectives from Equation 3 and 4, which leaves us with:

$$\Omega^{(l)} = \|\Psi(\rho_g^{(0)}x) - \rho_g^{(1)}(\Psi(x))\|_2^2 \tag{6}$$

Additionally, we can also use our model to sample data $\rho_g^{(0)}$ which we can use as data augmentation when computing the standard cross-entropy loss. One potential caveat to our method is the fixed filter-size that limits the range of operations we can learn. For instance, a $5\times$ 5 filter can only learn shifts of length two in all directions. This can be solved by iteratively applying the same filter for a random number of repeats when computing the objectives in Equation 2 and 6.

5.2 Experiments

First, we evaluate the generalization abilities to unseen shifts for our student model after transfer. In contrast to most methods presented above, our approach manages to outperform the student baseline after transfer (see Figure 2). This is the case not only for unseen shifts, but also for seen shifts, as the transferred equivariance helps with the increasingly more difficult test set as we increase the shifts in training. Additionally we also see that orbit transfer helps with a student architecture that replaces the pooling with a linear layer before the network's output (see Figure 3). This is especially noteworthy since direct matching failed to perform well in this scenario.

After exploring the different transfer methods in a controlled environment, we finally verify our results on a slightly more realistic task, namely MNIST (Deng, 2012). We follow the example of Abnar et al. (2020) and use a CNN trained on standard MNIST as the teacher for a student network without a built-in inductive bias for shift equivariance. In the first, focused, setting, we use simple three layer networks for both student and teacher, whereas the teacher consists of convolutional layers with maximum pooling and the student is a purely fully-connected network. The second, more realistic setting transfers from a ResNet-18 (He et al., 2015) to a small six layer vision transformer (ViT; Dosovitskiy et al., 2020). More details on the architecture and the training procedure can be found in the supplementary material. After training, we evaluate the trained model on both the standard – *centered* – MNIST test set, as well as the *translated* version provided by MNIST-C (Mu and Gilmer, 2019). In Table 1 we report the results for hyper-parameters that were selected on the translated test set in the *CNN → MLP* setting.

The results confirm the findings from the MNIST-1D experiments we report above. We again see that transferring shift equivariance to a fully connected network is a hard task for conventional functional transfer methods. Attention transfer (-3.8%) and RDL (-3.8%) both underperform compared to the student's baseline performance in the *CNN → MLP* setting. Even KD, which was reported by Abnar et al. (2020) to improve performance on the very same evaluation set, only achieves marginal improvements (+5.4%) in our setting, that focuses on shift equivariance exclusively. Our approach of Orbit transfer shows a strong improvement over the baseline performance of the student network (+59.5%). We observe similar results in the *ResNet18 → ViT* setting. KD only slightly improves shift performance (+3.9%) and Orbit shows major gains on the translated test set (+46.7%). Interestingly, RDL performs significantly better in this setting, both on the translated (+22.3%) as well as the centered test set (+0.7%).

5.3 Analysis

A control that trains the student with data augmentation identical to the translated test set reveals that our method almost reaches the same performance (-1.8%), even though it has never seen the translated set before test time. Following the same procedure of Mu and Gilmer (2019), we trained the student network with shift augmentations that do not include the center position. Therefore this student network shows overfitting behavior on the test shifts, which leads to a large drop in centered performance (-44.2%). The same is not true for our approach, since technically there are no "seen" or "unseen" shifts in our approach.

One big advantage of our equivariance transfer method is its interpretability. In Figure 5 we inspect the kernels that are learned for each group element in our group representation model. For a perfect model of shift equivariance, we would expect each filter to have a single non-zero position and all filters to be distinct. The learned filters resemble this expectation to a some

Arne F. Nix[1,2,†], Suhas Shrinivasan[2,3], Edgar Y. Walker[1,4,5], Fabian H. Sinz[1,2,‡]

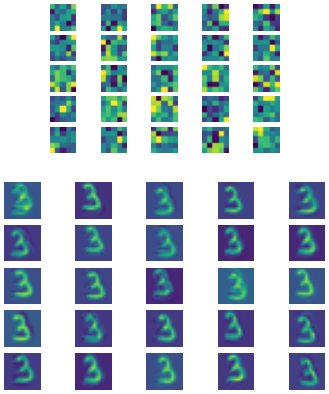


Figure 5: Rows 1-5: Kernels for all 25 linear representation of the group elements learned in the equivariance model. Rows 6-10: Kernels (in identical order) applied to an example input.

Method	Upright	Rotated
Teacher (G-CNN)	99.1	87.7
Student (MLP)	98.3	39.5
KD	98.7	46.3
RDL	**99.0**	45.6
Attention	98.3	41.1
Orbit	97.5	**76.5**

Table 2: Results on the MNIST test set with images in upright (column 1) or randomly rotated (column 2) orientation for four different transfer method for a G-CNN teacher and an MLP student.

extend. In cases where the filters are somewhat ambiguous, or where we do not know how the filters are expected to look like, we can even look at an example input after transforming it with out learned transformation network. In our case it becomes clear that the model did learn shifts, as we clearly see shifted versions of the input when applying the filters in Figure 5.

5.4 Rotation Experiments

In order to verify that our Orbit method can generalize to inductive biases other than shift equivariance, we apply it to the task of transferring equivariance to random rotations by multiples of ninety degree. Here the teacher model is a group-convolutional neural network (G-CNN; Cohen and Welling, 2016) and the goal is to transfer its inductive bias to a simple MLP. For Orbit, we model $\rho_g^{(l)}$ as an affine transformation (see above). We observe an effect similar to the shift equivariance setting (cf. Table 2), where established transfer methods show minimal effectiveness (up to +6.8% for KD) and Orbit performs remarkably well (+37.0%). However, we have to note that the corresponding optimization problem suffers from local minima, making it sensitive to initialization and – so far – preventing us from jointly transferring shift and rotation equivariance with the same model. More details on the experimental setup can be found in the supplementary material.

6 CONCLUSION

We investigated the transfer abilities of functional transfer methods and empirically showed in a simple controlled example that they are incapable of transferring even simple equivariances such as shift. We then showed for methods based on representational similarity that they cannot guarantee that the student network has the same linear representation of the equivariance as the teacher after training. Based on our insights, we developed Orbit, a novel transfer method that learns the equivariance properties of a given network and transfers them to a student network. Finally, we demonstrated that our method surpasses other methods by a large margin when transferring shift equivariance from a CNN to a fully-connected network. For future work, we are expanding these experiments to larger models and datasets, especially to more challenging symmetries such as rotations. Most importantly, our work shows promise and hopes to inspire approaching the transfer learning problem from the view of transferring useful and interpretable inductive biases.

Acknowledgements

We thank all reviewers for their constructive and thoughtful feedback. Furthermore, we thank Mohammad Bashiri for helpful comments and discussions. The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting Arne Nix.

This work was supported by the Cyber Valley Research Fund (CyVy-RF-2019-01). FHS is supported by the Carl-Zeiss-Stiftung and acknowledges the support of the DFG Cluster of Excellence "Machine Learning – New Perspectives for Science", EXC 2064/1, project number 390727645. This work was supported by an AWS Machine Learning research award to FHS. This work was supported by the German Research Foundation (DFG): SFB 1233, Robust Vision: Inference Principles and Neural Mechanisms, TP 06, project number: 276693517.

References

Samira Abnar, Mostafa Dehghani, and Willem Zuidema. Transferring Inductive Biases through Knowledge Distillation. 2020. URL https://github.com/samiraabnar/Reflect.http://arxiv.org/abs/2006.00555.

Abien Fred Agarap. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018.

Ari S Benjamin, David Rolnick, and Konrad P Kording. Measuring and regularizing networks in function space. In *7th International Conference on Learning Representations, ICLR 2019*, 2019.

Lucas Beyer, Xiaohua Zhai, Amélie Royer, Larisa Markeeva, Rohan Anil, and Alexander Kolesnikov. Knowledge distillation: A good teacher is patient and consistent. jun 2021. URL https://arxiv.org/abs/2106.05237v1http://arxiv.org/abs/2106.05237.

Cristian Bucilă, Rich Caruana, and Alexandra Niculescu-Mizil. Model compression. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, volume 2006, pages 535–541, 2006. ISBN 1595933395. doi: 10.1145/1150402.1150464.

Tomas B. Co. Matrix Analysis. In *Methods of Applied Mathematics for Engineers and Scientists*, pages 99–146. Cambridge University Press, oct 2013. ISBN 9781139020411. doi: 10.1017/cbo9781139021821.005.

Taco S. Cohen. *Equivariant convolutional networks*. PhD thesis, University of Amsterdam, 2021. URL https://dare.uva.nl.

Taco S. Cohen and Max Welling. Group Equivariant Convolutional Networks. *33rd International Conference on Machine Learning, ICML 2016*, 6: 4375–4386, feb 2016. URL https://arxiv.org/abs/1602.07576v3.

Elliot Creager, Jörn-Henrik Jacobsen, and Richard Zemel. Environment Inference for Invariant Learning. 2020. URL http://arxiv.org/abs/2010.07249.

Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. 2020. URL https://github.com/http://arxiv.org/abs/2010.11929.

Kunihiko Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4):193–202, 1980. ISSN 03401200. doi: 10.1007/BF00344251.

Jonathan Gordon, David Lopez-Paz, Marco Baroni, and Diane Bouchacourt. Permutation Equivariant Models for Compositional Generalization in Language. *Iclr*, (2019):1–12, sep 2020. URL https://github.com/facebookresearch/Permutation-Equivariant-Seq2Seqhttps://github.com/facebookresearch/Permutation-Equivariant-Seq2Seq%0Ahttps://openreview.net/forum?id=SylVNerFvr#.

Sam Greydanus. Scaling down Deep Learning. nov 2020. URL https://arxiv.org/abs/2011.14439v3http://arxiv.org/abs/2011.14439.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-Decem:770–778, dec 2015. ISSN 10636919. doi: 10.1109/CVPR.2016.90. URL https://arxiv.org/abs/1512.03385v1.

Geoffrey Hinton and Jeff Dean. Distilling the Knowledge in a Neural Network. Technical report, 2015.

Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *Advances in Neural Information Processing Systems*, volume 2015-Janua, pages 2017–2025. Neural information processing systems foundation, jun 2015. URL https://arxiv.org/abs/1506.02025v3.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, dec 2014. URL https://arxiv.org/abs/1412.6980v9.

Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big Transfer (BiT): General Visual Representation Learning. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 12350 LNCS, pages 491–507. Springer Science and Business Media Deutschland GmbH, dec 2020. ISBN 9783030585570. doi: 10.1007/978-3-030-58558-7_29. URL https://arxiv.org/abs/1912.11370v3.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of Neural Network Representations Revisited. *36th International Conference on Machine Learning, ICML 2019*, 2019-June:6156–6175, may 2019. URL https://arxiv.org/abs/1905.00414v4.

Zhe Li, Wieland Brendel, Edgar Y Walker, Erick Cobos, Taliah Muhammad, Jacob Reimer, Matthias Bethge, Fabian H Sinz, Xaq Pitkow, and Andreas S Tolias. Learning from brains how to regularize machines, 2019. ISSN 23318422.

Patrick McClure and Nikolaus Kriegeskorte. Representational distance learning for deep neural networks. *Frontiers in Computational Neuroscience*, 10(DEC): 131, dec 2016. ISSN 16625188. doi: 10.3389/fncom.2016.00131. URL http://journal.frontiersin.org/article/10.3389/fncom.2016.00131/full.

Norman Mu and Justin Gilmer. MNIST-C: A Robustness Benchmark for Computer Vision. jun 2019. doi: 10.5281/zenodo.3237938. URL https://arxiv.org/abs/1906.02337v1http://arxiv.org/abs/1906.02337.

Pingbo Pan, Siddharth Swaroop, Alexander Immer, Runa Eschenhagen, Richard E Turner, and Mohammad Emtiyaz Khan. Continual deep learning by functional regularisation of memorable past. In *Advances in Neural Information Processing Systems*, volume 2020-Decem, 2020.

Carlos Riquelme, Joan Puigcerver, Basil Mustafa, Maxim Neumann, Rodolphe Jenatton, André Susano Pinto, Daniel Keysers, and Neil Houlsby. Scaling Vision with Sparse Mixture of Experts. jun 2021. URL https://arxiv.org/abs/2106.05974v1http://arxiv.org/abs/2106.05974.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*, volume 9781107057. 2013. ISBN 9781107298019. doi: 10.1017/CBO9781107298019. URL https://books.google.de/books?hl=en&lr=&id=Hf6QAwAAQBAJ&oi=fnd&pg=PR15&ots=2IqhMhjJL2&sig=jvTRcWythPVe0bK8EttwOi7g6no.

Erik Henning Thiede, Truong Son Hy, and Risi Kondor. The general theory of permutation equivarant neural networks and higher order graph variational encoders. 2020. URL http://arxiv.org/abs/2004.03990.

Michalis K Titsias, Jonathan Schwarz, Alexander G. de G. Matthews, Razvan Pascanu, and Yee Whye Teh. Functional Regularisation for Continual Learning with Gaussian Processes. 2019. URL http://arxiv.org/abs/1901.11356.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers distillation through attention. 2020. URL http://arxiv.org/abs/2012.12877.

Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*, 2017. URL https://github.com/szagoruyko/attention-transfer.

Allan Zhou, Tom Knowles, and Chelsea Finn. Meta-Learning Symmetries by Reparameterization. 2020. ISSN 2331-8422. URL http://arxiv.org/abs/2007.02933.

Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A Comprehensive Survey on Transfer Learning, nov 2021. ISSN 15582256. URL https://arxiv.org/abs/1911.02685v3.

Supplementary Material: Can Functional Transfer Methods Capture Simple Inductive Biases?

A EXPERIMENTAL SETUP

A.1 MNIST-1D Experiments

Our goal was to design a simple teacher model that would generalize well to unseen shifts of the MNIST1D task in our evaluation setup. We ended up with a two-layer fully convolutional architecture with ReLU activations (Agarap, 2018), a stride of one and kernels of size five for all convolutions, fifteen channels in the first and ten in the second layer. A spatial max-pooling operation is performed at the final layer to obtain predictions for ten classes.

The student network is designed to replicate the teacher network as closely as possible, while using fully-connected layers instead of convolutions. Thus, to match the output-size of each layer, we use two hidden layers of size 600 and 400 with a ReLU activation in-between. To replicate the final max-pooling layer, we experiment with two variants of the student architecture. One architecture that replaces the max-pooling with a fully-connected layer, and then an alternative architecture that reshapes the output of the previous fully-connected layer into spatial and channel dimensions and on this representation performs a spatial max-pooling operation. The latter architecture simplifies the transfer problem, as only the equivariance of the lower layers has to be transferred instead of invariance of the entire network. More details on the architectures can be found in Table 3.

The training is always performed using Adam optimizer (Kingma and Ba, 2014) for 40,000 iterations with a batch size of 1000 examples. After every 100 steps the validation performance is evaluated and the learning rate is reduced by a factor of 0.8 if the accuracy has not improved for 20 evaluations. The training is interrupted if the learning rate has been reduced five times or if the maximum of 40,000 iterations is reached.

Teacher	Student	
1×15 Conv	40×600 FC	
ReLU	ReLU	
15×10 Conv	600×400 FC	
max-pooling	max-pooling	ReLU
		400×10 FC

Table 3: Comparison of different architectures we use for the MNIST1D experiments. (Channel-size for Conv and hidden-size for FC.) Note that both student architectures are identical for the first two layers.

Orbit Model

The model for the orbit transfer is selected in a way that allows it to capture the shift equivariance that we hope to transfer. Thus we chose a model that learns 40 filters of size 40 to model $\rho_g^{(0)}$ for $g = [0, 39]$. To additionally model $\rho_g^{(l)}$, we learn affine transformations (40×40 projections) for each layer as well as a down-projection for the output layer (40×10 projection).

Transfer Hyperparameters

As discussed in Section 2, we use a hyperparameter γ to interpolate between the standard cross-entropy loss $\mathcal{L}_{CE}$ and the transfer term Ω. In an extensive hyperparameter search, we selected the γ for each transfer method that performed best on the validation set. This left us with the following values: Attention 0.9, KD 0.6, Direct matching 0.4, RDL 0.9, Orbit 1.0. The same hyperparameter search gave us weights for the different components of the objective for learning the Orbit model ρ. The values selected for the final evaluation were $\gamma_{\text{equiv}} = 0.1, \gamma_{\text{group}} = 10.0, \gamma_{\text{inv}} = 10.0$, and γ_{CE} was fixed to 0.0.

A.2 MNIST-2D Shift Equivariance Experiments

A.2.1 CNN → MLP

In contrast to Abnar et al. (2020), we want to isolate the effect of shift equivariance and thus end up with a simpler architecture for both teacher and student. The teacher network has three convolutional layers with 128, 64 and 64 channels, a stride of one and a filter size of 3×3 each. Maximum-pooling is performed after layers two and three. A final linear layer is used to obtain the network's output. The student equals the teacher network in depth, but differs in layer size, with fully-connected layers of size 512,128 and 32. This discrepancy is expected in architectures as different as fully connected networks and CNNs, and it prevents the use of direct matching within the network for function transfer, as the layer outputs do not match in size. Both student and teacher use ReLU activation (Agarap, 2018), as well as dropout with $p = 0.1$ after each layer. More architectural details can be found in Table 4.

The training is performed using Adam optimizer (Kingma and Ba, 2014) with a learning-rate of 0.0003, a schedule of linear learning-rate warmup for 20 epochs, and a decay by factor 0.8 if there is no improvement in validation accuracy for 20 epochs. The training stops if the learning-rate is reduced five times or if it reaches 400 epochs. This training schedule is intentionally designed to be rather conservative, specifically to benefit functional distance methods as it was reported that knowledge distillation needs a lot of patience in training (Beyer et al., 2021).

Teacher	Student
1×128 Conv	784×512 FC
ReLU	ReLU
Dropout ($p = 0.1$)	Dropout ($p = 0.1$)
128×64 Conv	512×128 FC
ReLU	ReLU
max-pooling (2×2)	
Dropout ($p = 0.1$)	Dropout ($p = 0.1$)
64×64 Conv	128×32 FC
ReLU	ReLU
max-pooling (3×3)	
Dropout ($p = 0.1$)	Dropout ($p = 0.1$)
avg-pooling (global)	
64×10 FC	32×10 FC

Table 4: Comparison of student and teacher architecture for the MNIST experiments. (Channel-size for Conv and hidden-size for FC.)

A.2.2 ResNet18 → ViT

For the more realistic setting where we transfer a ResNet18 teacher to a ViT student, we use the same training and evaluation setup as above. The only change comes through the architectures we use. This is on the one hand a standard ResNet18, as it was described by He et al. (2015), and on the other hand a smaller variant of the vision transformer architecture (Dosovitskiy et al., 2020). The major changes compared to the original design are that we use six layers with eight attention heads and an embedding-size of 64. The patch-size applied on the input image is 7×7 pixels and we use a hidden-size of 128 for all position-wise feed-forward operations.

A.2.3 Orbit Model

The orbit model is trained with the same schedule as teacher and student. Here the validation loss is used to determine learning-rate decay and early stopping. We learn 25 kernels of size 5×5 to model $\rho_g^{(0)}$ for $g \in [0, 24]$. Additional affine projections are learned for each layer of the student, i.e. to transform $\rho_g^{(0)}$ into $\rho_g^{(l)}$. These projections are fully-connected, and thus we learn matrices of size 25×25 for each layer except for the output layer. There the filters are projected to size 10 instead. In order to stabilize the training, we cut off the gradient to the transformed input $\rho_g^{(0)} x$, which means that $\rho_g^{(0)}$ will only receive its feedback through $\rho_g^{(l)}$ with $l > 0$, which are affine projections of $\rho_g^{(0)}$.

A.2.4 Transfer Parameters

In prior experiments, we determined a reasonable value for γ for each transfer method. This left us with the following values: Attention 0.9, KD 1.0, RDL 0.8, Orbit 1.0. We also found that γ_{group} has a significant effect on the performance. Thus we searched for its value as part of our hyperparameter search and found γ_{group} to work best. The other components of the objective are all set to one.

A.3 MNIST-2D Rotation Equivariance Experiments

A.3.1 G-CNN $\rightarrow$ MLP

The student model in the rotation transfer experiments is the same MLP network that we used in the MNIST-2D experiments above. The training setup also remained unchanged, except that for the rotation transfer experiments we set a larger batch-size of 256 and train only for 200 epochs, as we noticed no change in performance after that point during prior experiments.

For the teacher model, we mainly followed the architecture of the G-CNN by Cohen and Welling (2016), but reduced the depth. Details can be seen in Table 5.

Teacher	Student
1×8 $p4$-Conv (kernel-size 5)	784×512 FC
max-pooling (2×2)	
ReLU	ReLU
8×32 $p4$-Conv (kernel-size 3)	512×128 FC
max-pooling (2×2)	
ReLU	ReLU
32×64 $p4$-Conv (kernel-size 3)	128×32 FC
max-pooling (2×2)	
ReLU	ReLU
64×10 $p4$-Conv (kernel-size 3)	32×10 FC
max-pooling over rotations	
Global spatial avg-pooling	

Table 5: Comparison of student and teacher architecture for the MNIST experiments. (Channel-size for Conv and hidden-size for FC.)

A.3.2 Orbit Model

As mentioned in Section 5, we decided to use a more general architecture for the orbit representation. Thus, we learn a separate affine transformation matrix $\rho_g^{(l)}$ for each group element $g = 1, \ldots, G$ and layer $l = 1, \ldots, L$. Each $\rho_g^{(l)}$ is initialized to a random affine transformation within a fair range (i.e. such that an image would still be recognizable). For this, we are considering shifts in range $[-0.1, 0.1]$ in x and y direction, rotations in range $[0, 360]$ degree along all axes, scaling by a factor in range $[0.8, 1.2]$ and shears by a factor in range $[0.0, 15.0]$. We trained models for $G = 4$, 8 and 25. All of our models outperformed existing transfer methods, but $G = 25$ performed the best. For this model, we report the average performance across three seeds in Table 2.

Arne F. Nix[1,2,†], Suhas Shrinivasan[2,3], Edgar Y. Walker[1,4,5], Fabian H. Sinz[1,2,‡]

B ABLATION STUDY ON LOSS COMPONENTS

We performed an ablation study to determine the interplay and importance of individual loss components. The results show that the best performance is reached when all loss components are used together (see Table 6). However, there are many combinations that show an acceptable performance on centered images while achieving above-baseline results on translated inputs. The most notable negative effect is observed when both the equivariance, as well as the invariance components are left out.

γ_{group}	γ_{equiv}	γ_{inv}	γ_{CE}	Centered	Translated
10.0	1.0	1.0	1.0	**99.02**	**96.48**
10.0	1.0	1.0	-	99.02	96.48
10.0	-	1.0	1.0	89.98	88.65
10.0	-	1.0	-	92.82	89.68
10.0	1.0	-	1.0	98.36	89.98
10.0	1.0	-	-	98.45	93.8
10.0	-	-	1.0	37.25	52.82
10.0	-	-	-	43.38	58.69
-	1.0	1.0	1.0	98.78	85.01
-	1.0	1.0	-	98.48	81.18
-	-	1.0	1.0	98.37	89.54
-	-	1.0	-	98.59	90.98
-	1.0	-	1.0	98.24	77.36
-	1.0	-	-	98.05	83.51

Table 6: Ablation study deactivating the different components of the objective: γ_{group}(Eq. 3), γ_{equiv} (Eq. 2), γ_{inv} (Eq. 4) and γ_{CE} (standard loss on transformed inputs)

HARD: Hard Augmentations for Robust Distillation

Arne F. Nix[1-2,*], Max F. Burg[2-3], Fabian H. Sinz[1-2,]**
[1] Institute for Bioinformatics and Medical Informatics, University of Tübingen
[2] Institute for Computer Science and Campus Institute Data Science, University of Göttingen
[3] Institute for Theoretical Physics, University of Tübingen

[*]arne.nix@uni-goettingen.de, [**]sinz@cs.uni-goettingen.de

Abstract

Knowledge distillation (KD) is a simple and successful method to transfer knowledge from a teacher to a student model solely based on functional activity. However, current KD has a few shortcomings: it has recently been shown that this method is unsuitable to transfer simple inductive biases like shift equivariance, struggles to transfer out of domain generalization, and optimization time is magnitudes longer compared to default non-KD model training. To improve these aspects of KD, we propose Hard Augmentations for Robust Distillation (HARD), a generally applicable data augmentation framework, that generates synthetic data points for which the teacher and the student disagree. We show in a simple toy example that our augmentation framework solves the problem of transferring simple equivariances with KD. We then apply our framework in real-world tasks for a variety of augmentation models, ranging from simple spatial transformations to unconstrained image manipulations with a pretrained variational autoencoder. We find that our learned augmentations significantly improve KD performance on in-domain and out-of-domain evaluation. Moreover, our method outperforms even state-of-the-art data augmentations and since the augmented training inputs can be visualized, they offer a qualitative insight into the properties that are transferred from the teacher to the student. Thus HARD represents a generally applicable, dynamically optimized data augmentation technique tailored to improve the generalization and convergence speed of models trained with KD.[1]

1 Introduction

Knowledge distillation (KD) methods [27, 37, 60] are powerful and flexible tools to transfer the knowledge of a given *teacher* model to the transfer target, the *student* model, without copying the weights. Instead, these methods match the student's functional activity (e.g. the softmax output) to that of the teacher for the presented inputs. Hence, those methods are independent of architectural details and allow knowledge distillation to be applied in scenarios like model compression [7, 27], continual learning [4, 42, 52], or even neuroscience [35], where traditional transfer learning would be impossible to use. KD methods also appear to be key to training new models that trade off inductive biases for more flexibility and more parameters [17, 53, 55] on smaller data [9, 40, 54]. However, Nix et al. [40] recently showed that current KD methods fail to transfer even simple equivariances between teacher and student. Additionally, previous work showed that KD leads to a larger gap between student and teacher on out-of-domain evaluation performance compared to within domain performance [6, 41], even in cases where the student almost perfectly matches the teacher [6] (see Table 5). This phenomenon is especially pronounced for particularly robust teachers [41]. Thus we expect that transferring robustness properties is a difficult problem for KD in general.

[1]Code available at https://github.com/sinzlab/HARD

Preprint. Under review.

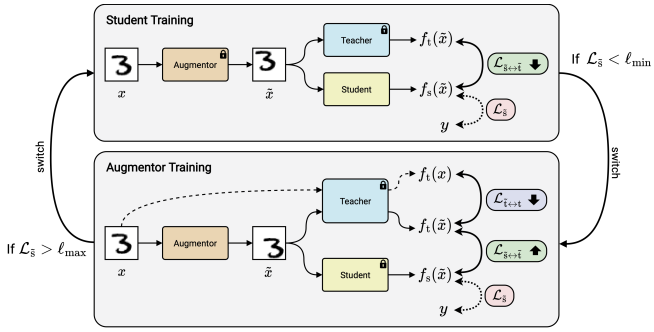


Figure 1: Our task-agnostic HARD framework switches between training the student to match the teacher and training the augmentor to generate new samples on which the student underperforms while maintaining high teacher performance. We optimize the augmentor and student in interchanging phases through a student-teacher loss $\mathcal{L}_{\tilde{s}\leftrightarrow\tilde{t}}$ and teacher-teacher loss $\mathcal{L}_{\tilde{t}\leftrightarrow t}$. We switch between the two phases by comparing the default loss $\mathcal{L}_{\tilde{s}}$ on augmented data to pre-defined thresholds.

We hypothesize that KD methods are in principle capable of transferring most knowledge from a teacher to a student if the training data is chosen adequately. We confirm this hypothesis on a small toy example (Section 3), showing the importance of input data for KD. Motivated by this demonstration, we propose our *Hard Augmentations for Robust Distillation (HARD)* method, a general framework (Section 4) to generate augmented training inputs which improve knowledge transfer by maximizing the distance between teacher and student while leaving the teacher's output unchanged. Consequently, our framework moves the input in directions that the teacher is invariant to but which are most challenging for the student. Our experiments (Section 5) show that our task-agnostic framework improves transfer effectiveness and thereby solves the problem of of KD not being able to transfer shift equivariance [40]. Additionally, as part of our framework, we propose several parameterized augmentations (Section 4.1) that can be integrated with most existing KD methods and are applicable to a variety of different computer vision tasks. Finally, we demonstrate across multiple different models on the tasks of CIFAR10 and ImageNet that our framework learns interpretable augmentations that improve KD to the same level and in many cases even beyond established data augmentation methods, even when evaluated in an out-of-domain setting.

2 Related Work

There is a long tradition in using data augmentations to artificially extend training data for deep learning models and particularly in computer vision, be it through adding Gaussian noise, random crops, shifts, flips, or rotations [18, 33]. In recent years, data augmentations became more complex [12, 24, 28, 39, 59, 61], employing a multitude of different heuristics with the aim to improve generalization and in some cases also out-of-domain performance [24]. A particularly popular augmentation method is *Mixup* [61], which randomly interpolates two input samples and their labels respectively. Similarly, *Cutmix* [59] combines two input images by pasting a random crop of one image on top of the other. Also, many studies use parameterized augmentations optimized to improve a given objective [11, 25, 48, 67], and some even optimize the augmentations to improve on an adversarial objective [2, 3, 20, 56, 63–65], however, without applying them for knowledge transfer.

In KD, applying data augmentations is a very effective tool to improve matching between student and teacher [6, 58] and optimizing on a meta level can be useful to aide the teaching [43]. Similar to our work, Haidar et al. [21], Rashid et al. [45], Zhang et al. [62] utilized adversarial objectives to optimize data augmentations for KD, however, they were solely focused on natural language processing tasks and do not optimize the augmentations towards invariance.

Inspired by this large body of work we formulate a task-agnostic framework containing only one building block that is specific to the data-domain – the instantiation of the augmentor model generating the augmented data samples – for which we offer a variety of reasonable model choices based on spatial transformer modules [29], Mixup [61], and variational autoencoders [10, 31, 34].

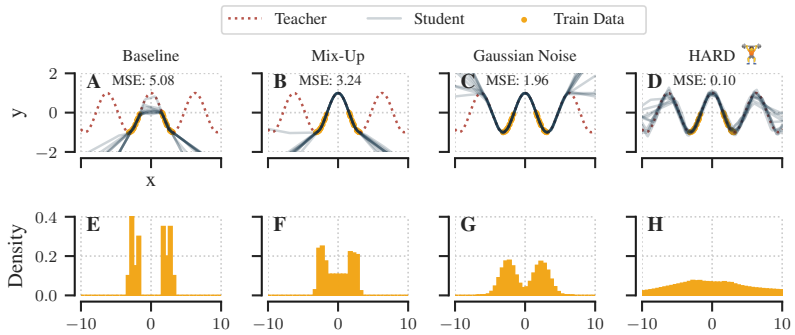


Figure 2: Fitting the student, a three-layer ReLU MLP, to the teacher function, $\cos(x)$, for $10,000$ iterations. We show results for 10 random seeds (**A**-**D**) and the distribution of (augmented) training inputs as a normalized histogram (**E**-**H**). We compare baseline (no augmentations) with Mixup, Gaussian noise and an HARD-optimized noise distribution. We report mean-squared-error (MSE) on 100 test inputs sampled from $\mathcal{U}_{[-10,10]}$.

3 Input Data Matters for Functional Transfer

We hypothesize that the choice of input data is crucial to successfully knowledge distillation and we illustrate the impact of training data by a simple toy example. To demonstrate this, consider a simple KD task in which we instantiate the teacher-model by the true function $f_{\mathrm{t}}(x) = \cos(x)$ and the student $f_{\mathrm{s}}(x)$ by a three layer Multilayer Perceptron (MLP) with ReLU activation [1]. We use input data x chosen such that it does not capture the teacher's $\cos(x)$ periodicity (orange points in Figure 2A). Simple KD does neither interpolate between the given training points nor extrapolates beyond them (Figure 2E). Hence the student neural network does not learn the teacher's periodicity and fails to interpolate and extrapolate beyond the training data (Figure 2A).

Augmenting the training data with more helpful inputs $\tilde{x}$ and teacher labels $f_{\mathrm{t}}(\tilde{x}) = \cos(\tilde{x})$ could mitigate this problem. One method successfully applied to KD [6] is to extend the input data through Mixup [61]. When applying this to our illustrative example, we create new training inputs $\tilde{x}$ through linear interpolation between pairs of input points $\tilde{x} = (1-\alpha)x_1 + \alpha x_2$ (Figure 2F), and recording the corresponding teacher responses $f_{\mathrm{t}}(\tilde{x}) = \cos(\tilde{x})$. Thus, the student learns to interpolate between training points, but mixup does not enhance extrapolation (Figure 2B).

To generate datapoints that would interpolate and extrapolate beyond already available training points, we could simply augment by adding Gaussian noise ϵ to the available data points, $\tilde{x} = x + \epsilon$, hence interpolating and extrapolating beyond the training data (Figure 2G). This strategy helps our student to match the teacher also outside the original training regime (Figure 2C). However, the student only improves within a fixed margin that is determined by the noise distribution's mean and variance.

We could obviously improve interpolation and extrapolation by increasing the noise distribution's variance or shifting it's mean, however, as we move to a high dimensional image input space ($x \in \mathbb{R} \rightarrow \vec{x} \in \mathbb{R}^N$) it becomes unclear how to heuristically select helpful new samples and at the same time random exploration strategies become computationally infeasible. Instead, we propose to optimize a parameterized augmentation to efficiently generate new, hard training samples on which the student lacks performance, as here the student could improve the most. In our toy example, we illustrate this by optimizing the Gaussian's parameters (mean and variance) according to our augmentation framework *HARD*, which we will present in the next section. This provides us with a noise distribution which we use to draw new helpful training examples $\tilde{x}$ that transfer inter- and extrapolation to the student network (Figure 2D,H). Overall, this toy example shows that learning hard augmentations to select new helpful data points is crucial to efficiently improve extrapolation beyond the training distribution.

4 Learning Hard Augmentations for Robust Distillation (HARD)

Our task-agnostic HARD framework learns augmenting training images to most efficiently help knowledge-transfer from a teacher to a student model. Our method requires three main components: a teacher model with frozen parameters, a student model that should learn knowledge from the teacher, and a parameterized augmentation model that learns to augment images such that most of the teacher's knowledge is transferred to the student.

In classical KD methods[27], the objective is to minimize a distance $\mathcal{D}\left[f_{\mathrm{s}}(x), f_{\mathrm{t}}(x)\right]$ between the student's activation $f_{\mathrm{s}}(x)$ and the teacher's activation $f_{\mathrm{t}}(x)$ on given inputs $x \in \mathbb{R}^n$. Usually, this would be the Kullback-Leibler divergence between the softmax distributions of teacher and student. Unfortunately, only considering training data could miss properties of the teacher (eg. shift invariance) that might be crucial for generalization (see Section 3 for an illustrative example). To resolve this issue, we learn a parametrized augmentation model g_{a} to generate new input data points $\tilde{x} = g_{\mathrm{a}}(x)$ transferring such invariance properties from the teacher to the student. Hence, we define a *teacher-student loss* considering the more general case of matching student and teacher on augmented inputs $\tilde{x} \in \mathbb{R}^n$:

$$\mathcal{L}_{\tilde{\mathrm{s}}\leftrightarrow\tilde{\mathrm{t}}} = \mathcal{D}\left[f_{\mathrm{s}}(\tilde{x}), f_{\mathrm{t}}(\tilde{x})\right] \ . \tag{1}$$

To specifically transfer the teacher's invariance properties to the student, we propose a *teacher-teacher loss* pushing the augmentor towards generating data points on which the teacher is invariant,

$$\mathcal{L}_{\tilde{\mathrm{t}}\leftrightarrow\mathrm{t}} = \mathcal{D}\left[f_{\mathrm{t}}(\tilde{x}), f_{\mathrm{t}}(x)\right] \ , \tag{2}$$

as these are often useful augmentations for generalization. Using both of these losses, we optimize the augmentor's parameters θ_{a} to generate augmented samples on which the teacher results in similar activations but the student differs from them (Figure 1 top) and simultaneously we optimize the student's parameters θ_{s} to perform well on those augmentations (Figure 1 bottom):

$$\max_{\theta_{\mathrm{a}}} \ \lambda_{\mathrm{s}}\mathcal{L}_{\tilde{\mathrm{s}}\leftrightarrow\tilde{\mathrm{t}}} - \lambda_{\mathrm{t}}\mathcal{L}_{\tilde{\mathrm{t}}\leftrightarrow\mathrm{t}} \qquad \text{and} \qquad \min_{\theta_{\mathrm{s}}} \ \mathcal{L}_{\tilde{\mathrm{s}}\leftrightarrow\tilde{\mathrm{t}}} \ . \tag{3}$$

Here, λ_{s} and λ_{t} trade off the loss terms and are treated as hyper-parameters. We train both components separately switching from training the augmentor to training the student when the student's performance on augmented data gets worse than a pre-defined threshold ($\mathcal{L}_{\tilde{\mathrm{s}}} > \ell_{\max}$) and we switch back from student to augmentor training when the student's performance on augmented data surpasses a pre-defined threshold ($\mathcal{L}_{\tilde{\mathrm{s}}} < \ell_{\min}$; Figure 1). To prevent catastrophic forgetting, we save augmentors at every switch and employ an augmentor randomly chosen out of the set of previously saved augmentors in each iteration when training the student.

4.1 The augmentor models

To generate new input data points it is important to choose an augmentor that suits the desired application and is powerful enough to generate useful augmentations. Usually, we do not know a priori what useful augmentations are and thus should try to allow as much flexibility as possible. Additionally, some variance over augmentations could benefit the transfer. Thus, all augmentors in our study introduce randomness in the model by adding Gaussian noise into the computation of the augmentation through the reparametrization trick [31]. While our framework is universally applicable across domains, choosing an effective augmentation model likely needs to be addressed for each task individually. In our experiments, we use the following augmentor models:

HARD-Affine In the simplest model, we limit the augmentations to affine transformations of the coordinate grid of pixel locations, i.e. shifts, rotations, scalings, and shears of images. Models implementing such transformations are known as *spatial transformers* [29]. We leverage this model for our augmentor by learning a distribution over the entries of an affine transformation matrix $\vartheta \in \mathbb{R}^{2\times 3}$ that defines the transformation of the sampling grid, i.e. a transformation that maps the pixel positions from the original image to the augmented image (Figure 3A).

HARD-Mix Additionally we consider a slightly more complex augmentor model, which is an adaptive variant of the commonly used Mixup [61] and Cutmix [59] augmentations. However, instead of randomly sampling the ratio and cutout position that are used to combine images, we learn how

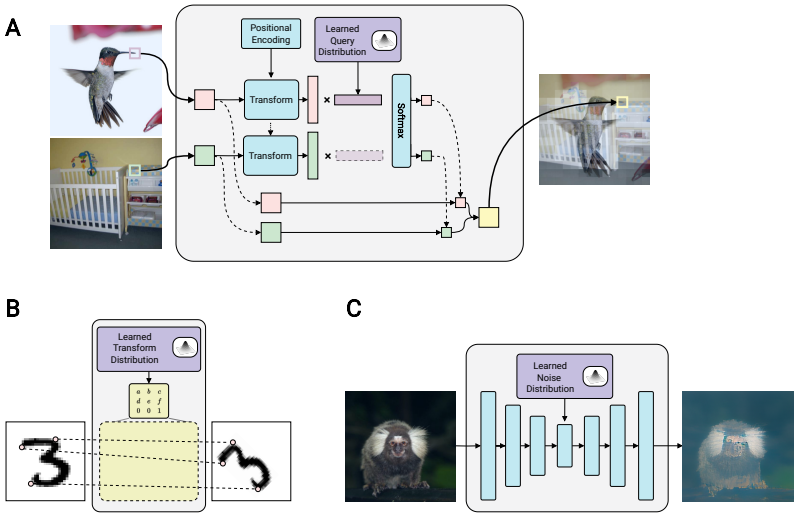


Figure 3: Illustration of the augmentor models used in our experiments. **(A)** HARD-Mix: Image-dependent patch-wise interpolation of multiple images. **(B)** HARD-Affine: Learned distribution of affine transformations in the pixel coordinates. **(C)** HARD-VAE: Finetuning (parts of) a pretrained VAE.

to combine the images dependent on the input images. We achieve this by performing a patch-wise projection of the input image, followed by comparing each patch with the same query vector sampled from a learned distribution (Figure 3B). We normalize similarities for each patch over each group of images and use the resulting weights to combine the original image patches, giving a combined image. This mechanism allows our augmentor to decide which features of which image are shown to the student, enabling it to explore the interpolated space between images systematically, instead of randomly. As it would not make sense for the teacher to be invariant to an interpolation as it is generated by HARD-Mix, we do not consider the teacher-teacher-loss $\mathcal{L}_{\tilde{t}\leftrightarrow t}$ in this case and optimize student and augmentor jointly instead.

HARD-VAE To lift constraints further, we wanted to use a more powerful augmentor that could generate a large variety of images across the entire image-space. As the augmentor has to generate new samples on-the-fly during the student training, the generation process needs to be very fast, limiting the choice of useful generative models. For this reason, we focus on variants of the variational autoencoder architecture [31], allowing for good image reconstructions which can be achieved reasonably fast in a single forward pass (Figure 3D). For CIFAR, we choose the *very deep VAE* [10] model, which we finetune by solely optimizing parameters of the posterior network from layer 10 onward in the decoder. For the experiments on ImageNet, we use a Residual-Quantized VAE (RQ-VAE) [34] pretrained on ImageNet, which we finetune in its entirety and add a noise vector on the latent state. Hence, as training progresses, the model changes from generating plain reconstructions of a given image to input conditioned generations that serve as our augmentations.

5 Experiments

5.1 Transferring equivariance

For our initial experiment, we reproduce the setup from Nix et al. [40] to test whether we can transfer the inductive bias from a shift equivariant teacher, CNN and ResNet18 [22], to a student that does not have this inductive bias built into its architecture: a Multi-Layer Perceptron (MLP) and a Vision Transformer (ViT) [17]. When training the students and teachers by themselves on standard MNIST [15] training data, we observe a small drop of generalization performance (-0.6% and -1.2%) between

Table 1: MNIST (columns "Centered") and MNIST-C (columns "Shifted") test accuracies (mean and standard error of the mean across 4 random seeds) comparing KD without augmentation and our HARD-Affine method to Orbit transfer [40], which also learns and transfers equivariances. The left two columns show the transfer results from a small CNN teacher to a MLP student. The right columns show analogous experiments between a ResNet18 teacher and a small ViT student. The best performing transfer is shown in bold for each column. Examples of our HARD-Affine learned data augmentations shown on the right. We include the controls *Random Affine* and *MNIST-C Shifts* (marked by italics).

	CNN → MLP		ResNet18 → ViT	
Method	Centered	Shifted	Centered	Shifted
Teacher only	99.0 ± 0.0	91.3 ± 0.5	99.5 ± 0.0	92.8 ± 0.5
Student only	98.4 ± 0.0	35.2 ± 0.7	98.3 ± 0.0	40.4 ± 0.8
+ *Random Affine*	*92.1 ± 0.6*	*81.0 ± 2.0*	*95.4 ± 0.3*	*90.4 ± 1.0*
+ *MNIST-C Shifts*	*98.1 ± 0.1*	*86.5 ± 0.3*	*98.5 ± 0.0*	*93.7 ± 0.2*
Orbit [40]	**98.8**	**95.2**	98.4	**84.0**
KD	98.6 ± 0.0	40.3 ± 0.6	98.6 ± 0.1	44.7 ± 1.9
+ HARD -Affine	98.6 ± 0.1	68.9 ± 2.5	**99.2** ± 0.0	**84.1** ± 2.3

teacher and student on the MNIST test set and a large gap (-56.1% and -52.4%) when we evaluate on a version of the test set in which digits were randomly shifted [38]. As another baseline, we applied plain KD to transfer shift equivariance from teacher to student. Consistent with the findings of Nix et al. [40], we only observe a small improvement on the centered (+0.2% and +0.3%) and the shifted (+5.1% and +4.3%) test sets, which likely result from the centered training data we use for transfer.

We then test if combining KD with our augmentations produced by HARD-Affine would outperform these baselines. The resulting student model improves significantly on shifted inputs (+28.6% and +39.4%) compared to plain KD and the generated images clearly show that the augmentor learns to shift the digits within the image. Compared to Nix et al. [40] our approach outperforms their results on the ViT task but, while improving the out-of-domain generalization by 28.6% over baseline, stays behind the Orbit performance on the MLP task. This demonstrates that our method while acting on fewer parts of the network compared to Orbit and while being a more general method, can improve or reach better performance when it comes to transferring invariances, and can be generalized to bigger datasets, as we show below.

We verify that the student's performance improvement is specifically due to our data generation framework in two control experiments. The first experiment (Random Affine) augments the training inputs of a stand-alone student model with a random affine transformation akin to our augmentor model, but using transformation parameters sampled uniformly from a pre-defined, reasonably constrained range (i.e. ensuring the digit is always fully visible). This student performs well on the shifted test set, however, performance significantly degrades on the centered test set. In comparison, our HARD-Affine model is unconstrained and learns more useful augmentations, leading to better performance on the centered test sets.

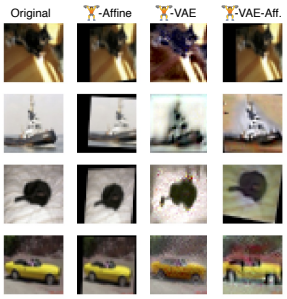
In our second control (Shifts) we asked how much data augmentation could improve the performance in the best case (without KD). For this, we augment the inputs by the same random shifts that were applied to obtain the shifted test data, leading to great improvements on the shifted test set. However, our learned augmentations achieve scores in a similar range on the shifted evaluation and outperform its results on the centered test set.

5.2 Transfer on natural images

After demonstrating that our method successfully captures the difference between teacher and student and bridges a gap in inductive bias, we now want to test whether this effect holds up in more realistic scenarios.

Table 2: Test accuracies on the CIFAR10 test set. Standard error of the mean is reported where available across three different seeds. Best transfer is highlighted in bold. The ResNet101* models were pretrained on ImageNet. Examples of augmented test images from ResNet18→ViT experiments with samples across different iterations are shown to the right.

	ResNet18 ↓ ViT	ResNet101* ↓ ViT	ResNet101* ↓ ResNet18
Teacher only	92.5 ± 0.0	95.5	95.5
Student only	68.5 ± 0.5	68.5 ± 0.5	78.5
+ Standard Aug.	78.3 ± 0.4	78.3 ± 0.4	92.6
+ Random Affine Aug.	58.9 ± 0.4	58.9 ± 0.4	79.3
KD	67.9 ± 0.1	68.5	84.4
+ Standard Aug.	80.9 ± 0.1	79.3	93.3
+ HARD-Affine	**87.8** ± **0.8**	84.4	93.5
+ HARD-VAE	81.9 ± 0.4	81.2	91.0
+ HARD-VAE-Affine	**87.6** ± **0.6**	**87.1**	**94.0**



CIFAR experiments We begin by applying our framework to CIFAR10 [32] on three different KD scenarios (see Table 2). Specifically, we test scenarios where the student lacks an inductive bias (ResNet18→ViT), where the teacher has more capacity and access to data than the student (ResNet101* →ResNet18), and to scenarios combining both properties (ResNet101* →ViT). For all experiments, we keep the experimental setup as close to our previous MNIST experiments as possible (see Appendix A for details).

We start by establishing baselines by training only the teacher and only the student models on the data and evaluating default KD. We observe that on this small data set a small ResNet18 performs better (78.5% accuracy) than a larger ViT (68.5%), likely because of the ResNet's superior inductive bias on this task and small data set. Next, we find that adding default data augmentations (random rotations, cropping, horizontal flips) to the student baselines significantly boosts performance to 92.6% and 78.3% for the ResNet18 and ViT, respectively. Adding these default augmentations to typical KD leads to a great performance boost, too (see Table 2).

Given that adding default data augmentation to KD already leads to a substantial performance boost, it is particularly noteworthy that the data augmentations learned by HARD-Affine outperform this baseline for the ViT. Qualitatively, the augmented images exhibit a large variety of spatial transformations, suggesting that a difference in these examples lead to the observed performance boost (Table 2, right).

We then investigated performance of our HARD-VAE augmentation strategy and found performance improvement over the KD + standard augmentations baseline for transfer to the ViT (+1.0% and +1.9%) student. However, inspecting the augmented images indicates that our augmentor lacks the expected shifts of object positions, but rather learns stylistic changes in the image (Table 2, right). This motivated us to combine HARD-Affine and HARD-VAE augmentation resulting in best performance (up to +7.8%) for all teacher-student pairings (HARD-VAE-Affine in Table 2) and the resulting images demonstrate variability in both style and spatial alignment (Table 2, right).

ImageNet experiments Having established our methods' performance for CIFAR10, we extend our results to classification on ImageNet [14]. Here we aim to distill a ResNet50 [22] teacher, trained with Deep-augment and AugMix data augmentations [25], into a smaller ResNet18 and ViT-S (small vision transformer variant) [17] that we want to be particularly robust to natural image corruptions,. The distillation into ResNet18 allows us to investigate the capability for model compression, because ResNet18 is a smaller network compared to ResNet50, but with a similar architecture. Distillation into a ViT-S architecture with a patch-size of 14 tests additionally if KD transfers the ResNet50's inductive bias of shift equivariance on a larger dataset.

We evaluate on common test sets for both in-domain (ID) [5, 46] and out-of-domain (OOD) [19, 23, 25, 26, 57] generalization performance (Tables 3 and 4, respectively). To properly investigate the extrapolation abilities of KD training, we trained a strong KD baseline by applying several

	ResNet50 → ResNet18			ResNet50 → ViT-S		
	Val	ReaL	V2	Val	ReaL	V2
Teacher	75.8	83.1	63.7	75.8	83.1	63.7
Student	70.7	78.1	57.4	73.2	79.4	60.3
KD	70.7	78.7	58.1	75.3	82.8	62.9
+ HARD🏋-Affine	**71.6**	**79.5**	58.6	74.9	82.3	62.4
+ HARD🏋-Mix	71.4	79.4	58.6	75.7	83.0	63.3
+ HARD🏋-VAE	71.0	78.9	**58.7**	**75.8**	**83.1**	**63.5**

Table 3: In-domain evaluation for ImageNet: reporting Top-1 accuracy in % on ImageNet-Validaton [14], ImageNet-ReaL [5] and ImageNet-V2 [46] with KD from a robust ResNet50 [25] teacher to ResNet18 (columns 2-4) and ViT-S (columns 5-7) students.

Table 4: In-domain evaluation for ImageNet: Reporting Top-1 accuracy in % on ImageNet-A [26], ImageNet-R [25], ImageNet-Sketch [57] and ImageNet-Style [19] and mean-corruption-error on ImageNet-C (lower is better) [23].

	ResNet50 → ResNet18					ResNet50 → ViT-S				
	Im-A	Im-R	Im-C ↓	Sketch	Style	Im-A	Im-R	Im-C ↓	Sketch	Style
Teacher	3.8	46.8	53.0	32.6	21.2	3.8	46.8	53.0	32.6	21.2
Student	1.6	30.0	88.1	18.4	4.4	**8.0**	26.3	78.1	13.8	6.6
KD	1.6	**40.2**	69.2	26.0	13.4	3.3	45.0	56.8	29.6	18.7
+ HARD🏋-Affine	1.5	38.2	73.1	24.9	10.4	3.4	40.8	62.2	26.2	14.5
+ HARD🏋-Mix	**1.8**	39.9	**68.8**	**26.1**	**13.7**	3.5	**45.4**	**56.2**	29.9	**19.2**
+ HARD🏋-VAE	1.7	39.5	72.5	25.8	12.1	3.4	**45.4**	57.4	**30.7**	18.1

data augmentations: we randomly switch between Cutmix [59] and Mixup [61], each drawing their interpolation weight from a β-distribution with $\alpha = 1$, as well as AugMix [24] augmentations. For the standalone student training, we additionally apply various lighter data augmentations (Cutmix with $\alpha = 1$, Mixup with $\alpha = 0.1$, and Trivialaugment [39]). Since we ask how KD can be improved in a setting of limited resources, we run our experiments an order of magnitude shorter than proposed for the state-of-the-art in KD [6] (200 epochs for all ResNet18 and 150 epochs for all ViT-S experiments). For student and KD models, we perform a small grid search over learning-rate and weight-decay hyperparameters. We then train the models with our HARD framework based on the hyperparameters of our best performing KD setting. The augmentor-specific settings are selected through a small grid-search in the ResNet18 setting (for details see Appendix A).

We first evaluate the ID performance of our methods (Table 3) beginning with the standalone teacher and student baselines, which reveal a larger performance gap between the ResNet18 student and the ResNet50 teacher compared to the ViT-S student (5.1% and 2.6% on the ImageNet validation set, respectively). Plain KD significantly reduces this gap for the ViT-S (+2.1% performance improvement compared to standalone). For the ResNet18 student KD achieves only small (0.7% V2) improvements or no improvements (0.0% Val), even though the initial gap between teacher and student is larger. Applying HARD-Affine, HARD-Mix and HARD-VAE augmentation on this task improves over plain KD across most augmentation models and test sets with student performance gains of up to 0.9% for ResNet18 (HARD-Affine) and 0.6% for ViT-S (HARD-VAE). For ViT-S, our best-performing HARD-VAE method even matches the teacher's performance on 2 out of 3 test sets.

For the OOD setting (Table 4), we observe that the initial gap between student and teacher is larger than on ID data across all data sets (up to 35.1% difference), except for Im-A in the ViT-S setting. The aggressive data augmentations we apply for the plain KD baseline favor OOD performance, hence it is expected that plain KD results in good performance improvement over the standalone baseline (up to 21.3% imporvement on Im-C). All three HARD approaches transfer some of the teacher's generalization abilities leading to improvements on a number of students and data sets, however, HARD-Affine fails to reach the KD performance in both settings and HARD-VAE underperforms for the ResNet18 student in these OOD scenarios. However, HARD-Mix and HARD-VAE (for ViT-S) outperform plain KD on several test sets and are roughly on par on all others, across the board. Given

that we chose a very strong baseline by applying aggressive state-of-the-art data augmentations we find these results especially encouraging.

6 Discussion

Interpretability HARD enables us to gain insight into the distillation mechanism as the augmented images illustrate the knowledge that is transferred (Figure 4). As expected, HARD-Affine learns to downscale images to shift and rotate the images such that the object in the image is shown in different places (row 2-4 in Figure 4) and scales such that the images is cropped (row 1). As HARD-Mix is a dynamically learnable extension of mixup, it either merges two objects into the same picture (row 1 and 4), especially if they are not in the same position, or uses one image to change the style (row 2) or background (row 3) of another. Finally, HARD-VAE mostly impacts the style of an image and additionally adds small distortions to specific image regions, which is noticable by the altered image brightness and the blurring of some high-frequency features.

Figure 4: Example augmentations applied to images of the ImageNet validation set obtained from augmentor models in the ViT-S setting at the end of training.

Limitations and broader impact State-of-the-art knowledge distillation typically deals with huge models (billions of parameters) and incredibly long training times (>9,000 epochs) [6, 13]. In comparison, our study is computationally lightweight in requiring approximately 400 A100GPU days across all our experiments. We believe exploring even more flexible augmentor models with a semantically meaningful latent space as for example diffusion models [44, 47, 49, 50] could improve our proposed methods even further. However, generating a single image with out-of-the-box diffusion models requires multiple seconds. This is prohibitively long, so leave exploring their usability in our proposed dynamic data augmentation technique for future work. In general, KD allows us to distill smaller models that perform similar to large foundation models. Improving the distillation process to be more efficient lowers the barrier of applying KD across labs with various compute budget and decreases environmental impact. At the same time, transferring generalization abilities effectively and consistently results in smaller distilled models that are appealing to use, thus we would expect such smaller models to be used abundantly hence lowering the general carbon footprint for model usage. In conclusion, our study proposes avenues to efficiently improve KD in terms of performance, efficiency, and hence environmental impact.

7 Conclusion

In this work we introduced a general, task-agnostic, and modular framework to extend knowledge distillation by learnable data augmentations. The augmentation models are optimized to generate inputs on which teacher and student disagree, keeping the teacher's predictions unchanged at the same time. We show that these augmentations can solve the issue of KD and transfer equivariance properties, even in cases where the teacher's inductive biases are distinct from the student's. We further demonstrate that our learned augmentations achieve performance competitive to established classical data augmentation techniques even when student and teacher share similar inductive biases. Overall our framework offers a powerful tool that enhances transfer performance and offers a unique insights into the transferred knowledge through its interpretable augmentations.

Acknowledgements

Furthermore, we thank Felix Schlüter for his helpful insights into evaluation problems as well as Mohammad Bashiri, Pawel Pierzchlewicz and Suhas Shrinivasan for helpful comments and discussions. The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting Arne Nix and Max F. Burg.

This work was supported by the Cyber Valley Research Fund (CyVy-RF-2019-01), by the German Federal Ministry of Education and Research (BMBF) through the Tübingen AI Center (FKZ: 01IS18039A), by the Deutsche Forschungsgemeinschaft (DFG) in the SFB 1233, Robust Vision: Inference Principles and Neural Mechanisms (TP12), project number: 276693517, and funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project-ID 432680300 – SFB 1456. FHS is supported by the Carl-Zeiss-Stiftung and acknowledges the support of the DFG Cluster of Excellence "Machine Learning – New Perspectives for Science", EXC 2064/1, project number 390727645.

References

[1] Abien Fred Agarap. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018.

[2] Anthreas Antoniou, Amos Storkey, and Harrison Edwards. Data Augmentation Generative Adversarial Networks. feb 2022.

[3] Sima Behpour, Kris M. Kitani, and Brian D. Ziebart. ADA: Adversarial data augmentation for object detection. *Proceedings - 2019 IEEE Winter Conference on Applications of Computer Vision, WACV 2019*, pages 1243–1252, mar 2019. doi: 10.1109/WACV.2019.00137.

[4] Ari S Benjamin, David Rolnick, and Konrad P Kording. Measuring and regularizing networks in function space. In *7th International Conference on Learning Representations, ICLR 2019*, 2019.

[5] Lucas Beyer, Olivier J Hénaff, Alexander Kolesnikov, Xiaohua Zhai, and Aäron van den Oord. Are we done with imagenet? *arXiv preprint arXiv:2006.07159*, 2020.

[6] Lucas Beyer, Xiaohua Zhai, Amélie Royer, Larisa Markeeva, Rohan Anil, and Alexander Kolesnikov. Knowledge distillation: A good teacher is patient and consistent. jun 2021. URL `https://arxiv.org/abs/2106.05237v1http://arxiv.org/abs/2106.05237`.

[7] Cristian Bucilă, Rich Caruana, and Alexandra Niculescu-Mizil. Model compression. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, volume 2006, pages 535–541, 2006. ISBN 1595933395. doi: 10.1145/1150402.1150464.

[8] Xiangning Chen, Chen Liang, Da Huang, Esteban Real, Kaiyuan Wang, Yao Liu, Hieu Pham, Xuanyi Dong, Thang Luong, Cho-Jui Hsieh, Yifeng Lu, and Quoc V. Le. Symbolic discovery of optimization algorithms, 2023.

[9] Xianing Chen, Qiong Cao, Yujie Zhong, Jing Zhang, Shenghua Gao, and Dacheng Tao. DearKD: Data-Efficient Early Knowledge Distillation for Vision Transformers. apr 2022. doi: 10.48550/arxiv.2204.12997. URL `https://arxiv.org/abs/2204.12997v2`.

[10] Rewon Child. Very Deep VAEs Generalize Autoregressive Models and Can Outperform Them on Images. nov 2020. doi: 10.48550/arxiv.2011.10650. URL `https://arxiv.org/abs/2011.10650v2http://arxiv.org/abs/2011.10650`.

[11] Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. AutoAugment: Learning Augmentation Policies from Data. *Cvpr 2019*, (Section 3):113–123, may 2018. doi: 10.48550/arxiv.1805.09501. URL `https://arxiv.org/abs/1805.09501v3`.

[12] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 702–703, 2020.

[13] Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, Rodolphe Jenatton, Lucas Beyer, Michael Tschannen, Anurag Arnab, Xiao Wang, Carlos Riquelme, Matthias Minderer, Joan Puigcerver, Utku Evci, Manoj Kumar, Sjoerd van Steenkiste, Gamaleldin F. Elsayed, Aravindh Mahendran, Fisher Yu, Avital Oliver, Fantine Huot, Jasmijn Bastings, Mark Patrick Collier, Alexey Gritsenko, Vighnesh Birodkar, Cristina Vasconcelos, Yi Tay, Thomas Mensink, Alexander Kolesnikov, Filip Pavetić, Dustin Tran, Thomas Kipf, Mario Lučić, Xiaohua Zhai, Daniel Keysers, Jeremiah Harmsen, and Neil Houlsby. Scaling vision transformers to 22 billion parameters, 2023.

[14] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[15] Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

[16] Tim Dettmers, Mike Lewis, Sam Shleifer, and Luke Zettlemoyer. 8-bit optimizers via block-wise quantization. *9th International Conference on Learning Representations, ICLR*, 2022.

[17] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. 2020. URL `https://github.com/http://arxiv.org/abs/2010.11929`.

[18] Logan Engstrom, Brandon Tran, Dimitris Tsipras, Ludwig Schmidt, and Aleksander Madry. A rotation and a translation suffice: Fooling cnns with simple transformations. 2017.

[19] Robert Geirhos, Claudio Michaelis, Felix A. Wichmann, Patricia Rubisch, Matthias Bethge, and Wieland Brendel. IMAGENET-TRAINED CNNS ARE BIASED TOWARDS TEXTURE; INCREASING SHAPE BIAS IMPROVES ACCURACY AND ROBUSTNESS, nov 2018. ISSN 23318422. URL `http://arxiv.org/abs/1811.12231`.

[20] Kehong Gong, Jianfeng Zhang, and Jiashi Feng. PoseAug: A Differentiable Pose Augmentation Framework for 3D Human Pose Estimation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 8571–8580, may 2021. ISSN 10636919. doi: 10.1109/CVPR46437.2021.00847. URL `https://arxiv.org/abs/2105.02465v1`.

[21] Md Akmal Haidar, Mehdi Rezagholizadeh, Abbas Ghaddar, Khalil Bibi, Philippe Langlais, and Pascal Poupart. CILDA: Contrastive Data Augmentation using Intermediate Layer Knowledge Distillation. apr 2022. doi: 10.48550/arxiv.2204.07674. URL `https://arxiv.org/abs/2204.07674v1http://arxiv.org/abs/2204.07674`.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-Decem:770–778, dec 2015. ISSN 10636919. doi: 10.1109/CVPR.2016.90. URL `https://arxiv.org/abs/1512.03385v1`.

[23] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations, mar 2019. ISSN 23318422. URL `http://arxiv.org/abs/1903.12261`.

[24] Dan Hendrycks, Norman Mu, Ekin D Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. *arXiv preprint arXiv:1912.02781*, 2019.

[25] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The Many Faces of Robustness: A Critical Analysis of Out-of-Distribution Generalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8320–8329, 2021. ISBN 9781665428125. doi: 10.1109/ICCV48922.2021.00823. URL `https://github.com/hendrycks/imagenet-r`.

[26] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. *CVPR*, 2021.

[27] Geoffrey Hinton and Jeff Dean. Distilling the Knowledge in a Neural Network. Technical report, 2015.

[28] Philip TG Jackson, Amir Atapour Abarghouei, Stephen Bonner, Toby P Breckon, and Boguslaw Obara. Style augmentation: data augmentation via style randomization. In *CVPR workshops*, volume 6, pages 10–11, 2019.

[29] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *Advances in Neural Information Processing Systems*, volume 2015-Janua, pages 2017–2025. Neural information processing systems foundation, jun 2015. URL https://arxiv.org/abs/1506.02025v3.

[30] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, dec 2014. URL https://arxiv.org/abs/1412.6980v9.

[31] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*. International Conference on Learning Representations, ICLR, dec 2014. doi: 10.48550/arxiv.1312.6114. URL https://arxiv.org/abs/1312.6114v10.

[32] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. CIFAR-10 (Canadian Institute for Advanced Research). URL http://www.cs.toronto.edu/$\sim$kriz/cifar.html.

[33] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

[34] Doyup Lee, Chiheon Kim, Saehoon Kim, Minsu Cho, and Wook-Shin Han. Autoregressive image generation using residual quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11523–11532, 2022.

[35] Zhu Li, Adrian Perez-Suay, Gustau Camps-Valls, and Dino Sejdinovic. Kernel Dependence Regularizers and Gaussian Processes with Applications to Algorithmic Fairness. Technical report, 2019.

[36] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts, 2017.

[37] Patrick McClure and Nikolaus Kriegeskorte. Representational distance learning for deep neural networks. *Frontiers in Computational Neuroscience*, 10(DEC):131, dec 2016. ISSN 16625188. doi: 10.3389/fncom.2016.00131. URL http://journal.frontiersin.org/article/10.3389/fncom.2016.00131/full.

[38] Norman Mu and Justin Gilmer. MNIST-C: A Robustness Benchmark for Computer Vision. jun 2019. doi: 10.5281/zenodo.3237938. URL https://arxiv.org/abs/1906.02337v1http://arxiv.org/abs/1906.02337.

[39] Samuel G Müller and Frank Hutter. Trivialaugment: Tuning-free yet state-of-the-art data augmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 774–782, 2021.

[40] Arne Nix, Suhas Shrinivasan, Edgar Y Walker, and Fabian Sinz. Can Functional Transfer Methods Capture Simple Inductive Biases? In Gustau Camps-Valls, Francisco J R Ruiz, and Isabel Valera, editors, *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 10703–10717. PMLR, 2022. URL https://proceedings.mlr.press/v151/nix22a.html.

[41] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023.

[42] Pingbo Pan, Siddharth Swaroop, Alexander Immer, Runa Eschenhagen, Richard E Turner, and Mohammad Emtiyaz Khan. Continual deep learning by functional regularisation of memorable past. In *Advances in Neural Information Processing Systems*, volume 2020-Decem, 2020.

[43] Hieu Pham, Zihang Dai, Qizhe Xie, and Quoc V. Le. Meta Pseudo Labels. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 11553–11563, mar 2021. ISSN 10636919. doi: 10.1109/CVPR46437.2021.01139. URL `https://arxiv.org/abs/2003.10580v4`.

[44] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical Text-Conditional Image Generation with CLIP Latents, April 2022. URL `http://arxiv.org/abs/2204.06125`. arXiv:2204.06125 [cs].

[45] Ahmad Rashid, Vasileios Lioutas, and Mehdi Rezagholizadeh. MATE-KD: Masked adversarial text, a companion to knowledge distillation. In *ACL-IJCNLP 2021 - 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, Proceedings of the Conference*, pages 1062–1071, 2021. ISBN 9781954085527. doi: 10.18653/v1/2021.acl-long.86.

[46] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*, pages 5389–5400. PMLR, 2019.

[47] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

[48] Evgenia Rusak, Lukas Schott, Roland S. Zimmermann, Julian Bitterwolf, Oliver Bringmann, Matthias Bethge, and Wieland Brendel. A simple way to make neural networks robust against diverse image corruptions. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12348 LNCS:53–69, jan 2020. ISSN 16113349. doi: 10.48550/arxiv.2001.06057. URL `https://arxiv.org/abs/2001.06057v5`.

[49] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding, May 2022. URL `http://arxiv.org/abs/2205.11487`. arXiv:2205.11487 [cs].

[50] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.

[51] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826, 2016. doi: 10.1109/CVPR.2016.308.

[52] Michalis K Titsias, Jonathan Schwarz, Alexander G. de G. Matthews, Razvan Pascanu, and Yee Whye Teh. Functional Regularisation for Continual Learning with Gaussian Processes. 2019. URL `http://arxiv.org/abs/1901.11356`.

[53] Ilya Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, and Alexey Dosovitskiy. MLP-Mixer: An all-MLP Architecture for Vision. *Advances in Neural Information Processing Systems*, 29:24261–24272, may 2021. ISSN 10495258. doi: 10.48550/arxiv.2105.01601. URL `https://arxiv.org/abs/2105.01601v4`.

[54] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. pages 10347–10357, jul 2020. ISSN 2640-3498. URL https://proceedings.mlr.press/v139/touvron21a.htmlhttp://arxiv.org/abs/2012.12877.

[55] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 2017-Decem, pages 5999–6009, 2017.

[56] Riccardo Volpi, John Duchi, Hongseok Namkoong, Vittorio Murino, Ozan Sener, and Silvio Savarese. Generalizing to Unseen Domains via Adversarial Data Augmentation. *Advances in Neural Information Processing Systems*, 2018-Decem:5334–5344, may 2018. ISSN 10495258. doi: 10.48550/arxiv.1805.12018. URL https://arxiv.org/abs/1805.12018v2.

[57] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pages 10506–10518, 2019.

[58] Huan Wang, Suhas Lohit, Michael Jones, and Yun Fu. Knowledge Distillation Thrives on Data Augmentation. dec 2020. doi: 10.48550/arxiv.2012.02909. URL https://arxiv.org/abs/2012.02909v1http://arxiv.org/abs/2012.02909.

[59] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6023–6032, 2019.

[60] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*, 2017. URL https://github.com/szagoruyko/attention-transfer.

[61] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond Empirical Risk Minimization. *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, oct 2017. doi: 10.48550/arxiv.1710.09412. URL https://arxiv.org/abs/1710.09412v2.

[62] Minjia Zhang, Niranjan Uma Naresh, and Yuxiong He. Adversarial Data Augmentation for Task-Specific Knowledge Distillation of Pre-trained Transformers. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10):11685–11693, 2022. ISSN 2159-5399. doi: 10.1609/aaai.v36i10.21423. URL www.aaai.org.

[63] Xiaofeng Zhang, Zhangyang Wang, Dong Liu, and Qing Ling. DADA: Deep Adversarial Data Augmentation for Extremely Low Data Regime Classification. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2019-May:2807–2811, may 2019. ISSN 15206149. doi: 10.1109/ICASSP.2019.8683197. URL https://arxiv.org/abs/1809.00981v1.

[64] Xinyu Zhang, Qiang Wang, Jian Zhang, and Zhao Zhong. Adversarial AutoAugment. dec 2019. doi: 10.48550/arxiv.1912.11188. URL https://arxiv.org/abs/1912.11188v1http://arxiv.org/abs/1912.11188.

[65] Long Zhao, Ting Liu, Xi Peng, and Dimitris Metaxas. Maximum-entropy adversarial data augmentation for improved generalization and robustness. In *Advances in Neural Information Processing Systems*, volume 2020-Decem, 2020. URL https://github.com/garyzhao/ME-ADA.

[66] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation, 2017.

[67] Dominik Zietlow, Michael Lohaus, Guha Balakrishnan, Matthäus Kleindessner, Francesco Locatello, Bernhard Schölkopf, and Chris Russell. Leveling Down in Computer Vision: Pareto Inefficiencies in Fair Deep Classifiers. 2022. URL http://arxiv.org/abs/2203.04913.

A Setup Details

Our experiments on MNIST were meant to reproduce Nix et al. [40] and thus follow their setup exactly, using the same training setup and model architectures.

A.1 CIFAR10 Experiments

Training We train on the entire CIFAR10 dataset (excluding 10% held-out as validation set) for 300 epochs with a batch-size of 256. As an optimizer, we use Adam [30] with a learning rate of 0.0003 and an L2-regularization of $2{\cdot}10^{-9}$. Our training begins with a linear warmup of the learning rate for 20 epochs. The validation accuracy is monitored after every epoch and if it has not improved for 20 consecutive epochs, we decay the learning rate by a factor of 0.8 and restore the previously best performing model. The training is stopped prematurely if we decay five times.

Models The different models we use generally follow the standard architecture and settings know from the literature. For the ViT, we use a smaller variant of it on the CIFAR task. It consists of six layers and eight attention heads throughout the network. The dropout rate is set to 0.1 and the hidden dimension is chosen as 512 in all places.

KD and HARD After initial experiments on MNIST, we decided to use a softmax temperature of 5.0 for all experiments involving KD. We furthermore rely solely on the KL-Divergence loss to optimize our model. For the experiments with our augmentation framework, we have the same settings as before for the student (KD) training and separate settings for the augmentor training. There we have different settings depeding on whether we use the VAE augmentor (or the Affine augmentor). There we reduce the batch-size to 160 (128) and a learning-rate of 0.0001 (0.05). We initialize both augmentors to perform an identity transformation, i.e. the VAE is taken pretrained from Child [10]. The thresholds for switching are set as $\ell_{\min} = 10\%(5\%)$ and $\ell_{\max} = 60\%(40\%)$. The train modi are switched if the threshold is surpassed for 5 consecutive iterations. Both λ_{s} and λ_{t} are set to 1 for the experiments. For the experiment ResNet101* $\rightarrow$ ResNet18, we found a slightly different setting to be more effective with $\ell_{\min}$ =5% and $\ell_{\max} = 40\%$ and a switch only happening if the threshold is surpassed for 10 consecutive iterations.

A.2 ImageNet Experiments

Baseline Training In general, all our ImageNet experiments follow a similar setup. We train with a batch-size of 512 samples using the Lion optimizer [8] with a linear learning-rate warmup to a defined initial learning-rate. Afterwards, we anneal the learning-rate following a cosine schedule [36] with a final value of 0. The training runs for 200 epochs for all ResNet18 experiments and 150 epochs for the ViT-S experiments. Throughout the training, the validation accuracy is monitored on a heldout set consisting of samples randomly chosen from training set, making up 1% of the total number of samples. The validation performance is used to pick the best performing epoch throughout training for final evaluation and the best hyperparameters during grid-search. We train at a resolution of 224 pixels with random resizing and cropping, as well as random horizontal flips applied in all trainings. All training runs are performed with automatic mixed precision and 8bit optimization [16].

Student Training After a grid-search, we found that for the standalone student training, an optimization with learning-rate 0.0001 with weight decay 0.1 for the ResNet18 student and learning-rate 0.00005 with weight decay 0.001 for the ViT-S student worked best. For both students, we apply light augmentations during training with Mixup ($\alpha = 0.2$) [61] and CutMix ($\alpha = 1.0$) [59]. For the ViT-S baseline, we additionally apply Trivial-Augment [39] and randomly erase pixels from the input image [66] with a probability of 0.1. We optimize the standard cross-entropy loss with additional label-smoothing [51] mixed in with a factor of 0.1.

KD and HARD As described in the main paper, the configuration for the KD experiments (including HARD) mainly differ in the choice of augmentation, as well as learning-rate and weight-decay. The plain KD experiments use Mixup ($\alpha = 1.0$) and CutMix ($\alpha = 1.0$) as well as AugMix [24] augmentation. The softmax temperature was chosen as 1.0 in prior experiments and kept for all experiments. The learning-rate for all KD and HARD experiments was chosen through a grid-search

to be 0.0001 in all cases and weight-decay is 0.001 in most cases, except for HARD experiments with a ResNet18 student where a weight-decay of 0.05 is used.

B Knowledge Distillation Results from the Literature

We (re-)evaluated student and teacher models from two high-performing KD experiments [6, 41] in the literature on both in-domain and out-of-domain test sets.

Table 5: In-domain and out-of-domain performance for two KD experiments from the literature. Showing that the gap in out-of-domain evaluations is larger than in-domain.

		In Domain			Out of Domain				
Model		Val	ReaL	V2	Im-A	Im-R	Im-C	Sketch	Style
BiT ResNet152 (Teacher)	[6]	82.9	87.8	72.0	31.9	49.2	51.0	37.4	16.9
BiT ResNet50 (Distilled)		82.8	87.5	72.5	25.1	45.3	51.8	31.6	15.1
Dino Teacher (Teacher)	[41]	86.5	89.7	78.4	76.1	79.3	27.3	62.8	34.6
ViT-S/14 (Distilled)		81.2	86.7	71.2	34.4	55.1	53.4	42.2	13.5