Exploring Representational Alignment with Human Perception Using Identically Represented Inputs

Vedant Nanda
MPI-SWS &
University of Maryland*

Ayan Majumdar
MPI-SWS

Camila Kolling
PUCRS†

John P. Dickerson
University of Maryland

Krishna P. Gummadi
MPI-SWS

Bradley C. Love
University College London &
The Alan Turing Institute

Adrian Weller
University of Cambridge &
The Alan Turing Institute

Abstract

We contribute to the study of the quality of learned representations. In many domains, an important evaluation criterion for safe and trustworthy deep learning is how well the invariances captured by representations of deep neural networks (DNNs) are shared with humans. We identify challenges in measuring these invariances. Prior works used gradient-based methods to generate identically represented inputs (IRI), i.e., inputs which have similar representations (on a given layer) of a neural network. If these IRIs look ‘similar’ to humans then a neural network’s learned invariances are said to align with human perception. However, we show that prior studies on the alignment of invariances between DNNs and humans are ‘biased’ by the specific loss function used to generate IRI. We show how different loss functions can lead to different takeaways about a model’s shared invariances with humans. We show that under an adversarial IRI generation process all models appear to have very little shared invariance with humans. We conduct an in-depth investigation of how different components of the deep learning pipeline contribute to learning models that have good alignment with human’s invariances. We find that architectures with residual connections trained using a self-supervised contrastive loss with ℓp ball adversarial data augmentation tend to learn the most human-like invariances.

1 Introduction

The ability to train deep neural networks (DNNs) which learn useful features and representations is key for their widespread use in a variety of domains [3,35]. In domains where deep learning models are being used for tasks that previously required human intelligence (e.g. image classification) and where safety and trustworthiness are important considerations, it is often important to assess the alignment of the learned representations with human perception. Such alignment assessments can crucially help in understanding and diagnosing issues such as lack of robustness to distribution shifts [49,61], adversarial attacks [45,18] or using undesirable features for a downstream task [2,54,51,4,25].

To study alignment with human perception, prior works have used the approach of representation inversion [38]. The key idea is the following: given an input to a neural network, the approach first finds identically represented inputs (IRIs), i.e. inputs which have similar representations on some given layer(s) of the neural network. In the second step, the inputs that are perceived similarly by the

*Correspondence to: vnanda@mpi-sws.org
†Work done during an internship at MPI-SWS

Preprint. Under review.
neural network are checked by humans for visual similarity. Thus, the approach relies on estimating whether a transformation of the inputs which is representation invariant to a neural network is also an invariant transformation to the human eye, i.e. it checks whether models and humans have shared or aligned invariances.

Prior works [11, 13] used gradient-based methods to generate IRIs for a given target input starting with a random seed input. These works revealed exciting insights: (a) Feather et al. [13] studied representational invariance for different layers of DNNs trained over ImageNet data (using the standard cross-entropy loss). They showed that while later layer representations of DNNs do not share any invariances with human perception, the earlier layers are somewhat better aligned with human perception. (b) Engstrom et al. [11] found that, unlike standard DNNs, adversarially robust DNNs, i.e., DNNs trained using adversarial training [37], learn representations that are well aligned with human perception, even in later layers [11, 27, 50]. However, these findings are contradicted when differently regularized methods are used for generating IRIs [38, 43, 42, 38]; these latter methods show that even later layers of standard DNNs learn human aligned representations. Also, these works do not formally define a measure that can quantify alignment with human perception beyond relying on visual perception of the images by the authors.

We seek to make sense of these confusing earlier results, and thereby to better understand alignment, and how the process used to generate IRIs influences insights about DNNs learning human-like invariances. We group existing generation processes into two broad categories: regularizer-free (as in [13]), where the goal is to find an IRI without any additional constraints; and human-leaning (as in [43, 42, 38]), where the goal is to find an IRI that is also visually human-comprehensible. Additionally, we propose and explore a new (third) broad category, adversarial, where the goal is to find an IRI that is visually (from a human perception perspective) far apart from the target input. We show that when we evaluate alignment of DNNs’ representations and human perception using IRI generation methods from these three categories, we arrive at very different conclusions. Specifically, compared to the regularizer-free IRI generation approach, the human-leaning IRI generation approach over-estimates a model’s alignment with human perception, while the adversarial approach under-estimates a model’s alignment with human perception. We also show how the alignment can be quantified reliably by designing simple visual perception tests that can be crowdsourced, i.e. used in human surveys.

Next, inspired by the prior works that suggest that changes in the model training pipeline (as in training adversarially robust DNNs [11, 27]) can lead to human-like invariant representations, we conduct an in-depth investigation to understand what parts of the deep learning pipeline are crucial in helping DNNs better learn human-like invariances. Our analysis shows that architectures play a crucial role. For CNN like models, architectures with residual connections are better. We also show that in certain cases, using data augmentation during training is key to learning aligned representations. Additionally, we find that when the same architectures are trained with a self-supervised contrastive loss (e.g., SimCLR [6]), the learned representations – while typically having lower accuracies than their fully supervised counterparts – have high shared invariances with human perception. Prior work has suggested that robust training induces a human prior over learned representations [11], however, our results suggest that the story is more nuanced and that carefully combining other techniques with robust training can lead to even better representations.

We highlight the following contributions:

• We offer insights into the challenges of measuring shared invariances of DNNs with human perception. We show how different losses used for generating IRIs lead to different conclusions about a model’s shared invariances with human perception, thus helping to make sense of conflicting earlier works that do not engage with these challenges.

• We show how using an adversarial IRI generation loss, we can (almost always) discover invariances of DNNs that do not align with human perception, thus showing that there is scope to design better mechanisms to learn representations that are more aligned with human perception.

• We conduct an in-depth study of how loss functions, architectures, data augmentations and training paradigms contribute to learning human-like shared invariances.
| Seed ($x_0$) | Regularizer-free | Human-aligned Regularizer | Adversarial Regularizer |
|-------------|------------------|--------------------------|------------------------|
| Target ($x_t$) | Model | Result ($x_r$) |
| Standard |  |  |  |
| AT $\ell^2$, $\epsilon = 1$ |  |  |  |
| Standard |  |  |  |
| AT $\ell^2$, $\epsilon = 1$ |  |  |  |

Figure 1: **Representation Inversion for different kinds of $R$; For ImageNet trained ResNet50**

Images are generated by starting from $x_0$ and solving Eq 2 with different kinds of regularizers. For the vanilla ResNet50, with regularizer free and adversarial inversion, $x_r$ looks perceptually much closer to $x_0$ than $x_t$, even though from the model’s point of view, $x_r$ and $x_t$ are the same. However, with the human-aligned regularizer, we see that $x_r$ contains some information like color patterns of $x_t$. For adversarially robust ResNet50 [37, 55] even though regularizer-free and human-aligned inversions look perceptually similar to $x_t$, for the adversarial regularizer even these models produce $x_r$ that looks nothing like $x_t$.

## 2 Measuring Shared Invariance with Human Perception

Measuring the extent to which invariances learned by DNNs are shared by humans is a two step process. We first need to generate IRIs, i.e., inputs that are mapped to similar representations by the DNN. Then we need to assess if these inputs are considered similar by humans. More concretely, if invariances of a given DNN ($g_{model}$) are shared by humans ($g_{human}$) on a set of $n$-dimensional samples $X \in \mathbb{R}^{n \times d}$, then:

$$g_{human}(X^i) \approx g_{human}(X^j) \forall (X^i, X^j) \in S \times S ; S = \{X\} \cup \{X^i \mid g_{model}(X^i) \approx g_{model}(X)\}.$$

$S$ denotes the IRIs for $g_{model}$. There are three major challenges here:

- Access to representations in the brain, i.e., $g_{human}$ is not available.
- Due to the highly non-linear nature of DNNs, $S$ can be very hard to obtain.
- The fine-grained input space implies very many inputs $n$, making the choice of $X$ hard.

We address each of these below. We also show how prior works that do not directly engage with these points can miss important issues in their conclusions about shared invariances of DNNs and humans.

### 2.1 Measuring Alignment of Invariances Using $g_{human}$

Assuming we have a set of images with similar representations ($S$; discussed in Section 2.2), we must check if humans also perceive these images similarly. The extent to which humans think this set of images is similar defines how aligned the invariances learned by the DNN are with human perception. In prior works this has been done by either eye-balling IRIs [11] or by asking annotators to assign class labels to IRIs [13]; both approaches do not scale. Additionally, assigning class labels to IRIs limits $X$ to being samples from a data distribution containing human-recognizable images (i.e., $X$ cannot be sampled from any arbitrary distribution) with only a few annotations (e.g., asking annotators to assign one class label out of 1000 ImageNet classes is not feasible). To get past the issue of scalability and class labels, we propose the following as a measure of alignment between DNN’s and human’s invariances:
(b) Hard ImageNet Clustering

A value close to 33% for clustering means random assignment and indicates no alignment. We see that LPIPS and humans rank models similarly in both 2AFC and clustering setups, thus showing that LPIPS is a reliable proxy for judging perceptual similarity of IRIs. These experiments were conducted on IRIs generated using regularizer-free loss in Eq 2. The variance reported for LPIPS is for different backbone networks that are available for LPIPS.

![Examples of images used in the experiments](Image 249x328 to 338x423)

Figure 2: [Survey Prompts for AMT workers] In the 2AFC (left) setting we ask the annotator to choose which of the two images (x_t or x_0) is perceptually closer to the query image (x_r). In the clustering setting (center and right) we show 3 images from the dataset (target images, x_t) in the columns and for each of these, we generate x_r_1 \in S_{x_t} and x_r_2 \in S_{x_t}. Each of these is shown across the rows. The task here is to match each image on the row with the corresponding target image on the column.

$$\text{Alignment} = \frac{|A|}{\sum_{x_t \in X} |S_{x_t}|},$$

where $|A|$ is the number of matches and $|S_{x_t}|$ is the number of images in the set $S_{x_t}$.

$$A = \{x_r \mid ||g_{human}(x_t) - g_{human}(x_r)|| < ||g_{human}(x_0) - g_{human}(x_r)|| \forall x_t \in X, x_r \in S_{x_t}, \}
S_{x_t} = \{x_r \mid g_{model}(x_r) \approx g_{model}(x_t) \forall x_t \in X \}$$

Table 1: [CIFAR10 and ImageNet Surveys To Confirm Efficacy of LPIPS] We use LPIPS to simulate a human in both 2AFC and Clustering setups described in Section 2.1 and compare it with AMT worker’s responses. A value close to 33% for clustering means random assignment and indicates no alignment. We see that LPIPS and humans rank models similarly in both 2AFC and clustering setups, thus showing that LPIPS is a reliable proxy for judging perceptual similarity of IRIs. These experiments were conducted on IRIs generated using regularizer-free loss in Eq 2. The variance reported for LPIPS is for different backbone networks that are available for LPIPS.
We also explore a third kind of \( \text{IRI} \) where \( \lambda \) is regularizer-free. These methods do not use a regularizer, as also noted by \cite{[43]}. We identify the following distinct categories of IRIs based on the choice of \( \lambda \).

To ensure the efficacy of LPIPS as a proxy for human judgements, we deploy two types of surveys on Amazon Mechanical Turk (AMT) to also elicit human similarity judgements. Prompts for these surveys are shown in Fig. 2. We received approval from the Ethical Review Board of our institute for this survey. Each survey consists of 100 images plus some attention checks to ensure the validity of our responses. The survey was estimated to take 30 minutes (even though on average our annotators took less than 20 minutes), and we paid each worker 7.5 USD per survey.

**Clustering** In this setting, we ask humans to match the IRIs \((x_r)\) on the row to the most perceptually similar image \((x_t)\) on the column (each row can only be matched to one column). A prompt for this type of a task is shown in Fig. 2b & 2c. Once we get these responses, a quantitative measure of alignment can be calculated by measuring the fraction of \(x_r\) that were correctly matched to their respective \(x_t\). For ImageNet, we observed that a random draw of three images (e.g., Fig. 2c) can often be easy to match to based on how different the drawn images \((x_t)\) are. Thus, we additionally construct a “hard” version of this task by ensuring that the three images are very “similar” (as shown in Fig. 2b). We leverage human annotations of ImageNet-HSJ \cite{[50]} to draw these similar images. More details can be found in Appendix A.

**2AFC** This is the exact test used to generate \( \mathcal{A} \). In this setting we show the annotator a reconstructed image \((x_r)\) and ask them to match it to one of the two images shown in the options. The images shown in the options are the seed \((x_0)\), i.e., starting value of \(x\) in Eq. 2 and the original image \((x_t)\). Since \(x_r\) and \(x_t\) are IRIs for the model (by construction), alignment would imply humans also perceive \(x_r\) and \(x_t\) similarly. See Fig. 2a for an example of this type of survey.

### 2.2 Generating IRIs

Even if we assume a finite sampled set \( X \sim \mathcal{D} \) (discussed in Section 2.3), there can be many samples in \( \mathcal{S} \) due to the highly non-linear nature of DNNs. However, we draw on the insight that there is often some structure to the set of IRIs, that is heavily dependent on the IRI generation process. Prior work on understanding shared invariance between DNNs and humans \cite{[?, e.g.,][38, 43, 42, 40, 41]} has used representation inversion \cite{[38]} to generate IRIs. However, IRIs generated this way depend heavily on the loss function used in representation inversion, as demonstrated by \cite{[43]}. Fig. 1 shows how different loss functions can lead to very different looking IRIs. We group these losses used in the literature to generate IRIs into two broad types: **regularizer-free** (used by \cite{[11, 13]}), and **human-leaning** (used by many works on interpretability of DNNs including \cite{[38, 43, 42, 40, 41]}).

We also explore a third kind of adversarial regularizer, that aims to generate controversial stimuli \cite{[17]} between a DNN and a human.

Representation inversion is the task of starting with a random seed image \( x_0 \) to reconstruct a given image \( x_t \in X \) from its representation \( g(x_t) \) where \( g(\cdot) \) is the trained DNN. The reconstructed image \((x_r)\) is same as \(x_t\) from the DNN’s point of view, i.e., \( g(x_t) \approx g(x_r) \). This is achieved by performing gradient descent on \( x_0 \) (in our experiments we use SGD with a learning rate of 0.1) to minimize a loss of the following general form:

\[
\mathcal{L}_x = \frac{||g(x_t) - g(x)||_2}{||g(x_t)||_2} + \lambda \cdot \mathcal{R}(x)
\]

(2)

where \( \lambda \) is an appropriate scaling constant for regularizer \( \mathcal{R} \). All of these reconstructions induce representations in the DNN that are very similar to the given image \((x_t)\), as measured using \( \ell_2 \) norm. Depending on the choice of seed \( x_0 \) and the choice of \( \mathcal{R} \), we get different reconstructions of \( x_r \) thus giving us a set of inputs \( \{x_r, x_{r_1}, \ldots, x_{r_k}\} \) that are all mapped to similar representations by \( g(\cdot) \).

Doing this for all \( x_t \in X \), we get the IRIs \( S = \{X, X_{r_1}, \ldots, X_{r_k}\} \).

In practice we find that the seed \( x_0 \) does not have any significant impact on the measurement of shared invariance. However, the choice of \( \mathcal{R} \) does significantly impact the invariance measurement (as also noted by \cite{[43]}). We identify the following distinct categories of IRIs based on the choice of \( \mathcal{R} \).

**Regularizer-free.** These methods do not use a regularizer, i.e., \( \mathcal{R}(x) = 0 \).

---

1For all evaluations we report the average over 4 different backbones used to calculate LPIPS including the finetuned weights released by the authors. More details in Appendix A.2.
### CIFAR10

| Training | Model       | REG.-FREE | HUMAN-ALIGNED | ADVERSARIAL | CLEAN ACC. | ROBUST ACC. |
|----------|-------------|-----------|---------------|-------------|------------|-------------|
| AT       | ResNet18    | 63.25 ± 0.26 | 79.00 ± 2.19 | 0.33 ± 0.07 | 80.77      | 50.92       |
|          | VGG16       | 0.25 ± 0.43 | 41.41 ± 16.74 | 1.00 ± 1.41 | 79.84      | 48.36       |
|          | InceptionV3 | 23.25 ± 0.56 | 64.75 ± 24.17 | 3.00 ± 4.24 | 81.57      | 51.02       |
|          | DenseNet121 | 82.75 ± 20.07 | 86.25 ± 14.50 | 1.33 ± 1.89 | 83.22      | 52.86       |
| Standard | ResNet18    | 0.00 ± 0.00 | 21.09 ± 13.51 | 1.33 ± 1.89 | 94.94      | 0.00        |
|          | VGG16       | 0.00 ± 0.00 | 21.88 ± 14.82 | 0.00 ± 0.00 | 93.63      | 0.00        |
|          | InceptionV3 | 0.00 ± 0.00 | 21.88 ± 17.54 | 0.33 ± 0.47 | 94.59      | 0.00        |
|          | DenseNet121 | 0.00 ± 0.00 | 26.56 ± 16.90 | 0.00 ± 0.00 | 95.30      | 0.00        |

| Training | Model       | REG.-FREE | HUMAN-ALIGNED | ADVERSARIAL | CLEAN ACC. | ROBUST ACC. |
|----------|-------------|-----------|---------------|-------------|------------|-------------|
| AT       | ResNet18    | 42.00 ± 38.33 | 46.75 ± 39.37 | 0.33 ± 0.47 | 53.12      | 31.02       |
|          | ResNet50    | 51.00 ± 34.89 | 45.75 ± 37.39 | 14.00 ± 3.74 | 62.83      | 38.84       |
|          | VGG16       | 55.50 ± 34.14 | 55.50 ± 38.29 | 11.00 ± 3.74 | 56.79      | 34.46       |
| Standard | ResNet18    | 0.00 ± 0.00 | 17.00 ± 28.30 | 0.00 ± 0.00 | 69.76      | 0.01        |
|          | ResNet50    | 0.00 ± 0.00 | 16.25 ± 26.42 | 0.00 ± 0.00 | 76.13      | 0.00        |
|          | VGG16       | 0.00 ± 0.00 | 0.00 ± 0.00   | 0.00 ± 0.00 | 73.36      | 0.16        |

Table 2: [CIFAR10 and ImageNet Model Alignment Results for Different Regularizers] For different regularizers, we see that ranking of models can look very different. For example, for Adversarially Trained (AT) Resnet18 vs InceptionV3 on CIFAR10, we see that regularizer-free inversion leads to Resnet18 being significantly more aligned, but the trend is much less pronounced for the human-aligned regularizer. We also find that alignment can vary quite a bit between different architectures – all of which achieve similar clean and robust accuracies.

**Human-aligned regularizer.** This kind of a regularizer purposefully puts constraints on $x$ such that the reconstruction has some “meaningful” features. They use $R(x) = TV(x) + ||x||_p$ where $TV$ is the total variation in the image. Intuitively this penalizes high frequency features and smoothens the image to make it look more like natural images. Achieve a similar kind of high frequency penalization by blurring $x$ before each optimization step. We combine both these frequency-based regularizers with pre-conditioning in the Fourier domain and robustness to small transformations. More details can be found in Appendix [A.4]. Intuitively a regularizer from this category generates IRIs that have been “biased” to look meaningful to humans.

**Adversarial regularizer.** We propose a new regularizer to generate IRIs while intentionally making them look perceptually dissimilar from the target, i.e., $R = -||g_{human}(x_t) - g_{human}(x)||_p$ (negative sign since we want to maximize perceptual distance between $x$ and $x_t$). We leverage LPIPS (Learned Perceptual Image Patch Similarity), a widely used perceptual distance measure, to approximate $||g_{human}(x_t) - g_{human}(x)||_p$. LPIPS uses initial layers of an ImageNet trained model (finetuned on a dataset of human similarity judgements) to approximate perceptual distance between images which makes it differentiable and thus can be easily plugged into Eq. 2. Thus, the regularizer used is $R(x) = -\text{LPIPS}(x, x_t)$. IRIs generated using this regularizer can be thought of as controversial stimuli – they’re similar from the DNN’s perspective, but distinct from a human’s perspective.

### 2.3 Choice of inputs $X$

In order to overcome the challenge of choosing $X$, we assume $X$ to be sampled from a given data distribution $\mathcal{D}$. In our experiments, we try out many different distributions, including the training data distribution and random noise distributions, and find that takeaways about a alignment of model’s invariances with humans do not depend heavily on the choice of $\mathcal{D}$. Some examples of $X$ sampled from the data distribution and noise distributions (two random Gaussian distributions, $\mathcal{N}(0, 1)$ and $\mathcal{N}(0.5, 2)$), along with the corresponding IRIs are shown in Fig. 1. Interestingly, the human-aligned regularizer, which explicitly tries to remove high-frequency features from $x$, fails to reconstruct an $x_t$ that itself consists of high-frequency features.
2.4 Evaluation and Takeaways

For each model, we randomly picked 100 images from the data distribution along with a seed image with random pixel values. For each of the 100 images, we do representation inversion using one regularizer each from regularizer-free, human-aligned, and adversarial.

**Reliability of using LPIPS** Table 1 shows the results for the surveys conducted with AMT workers. Each survey was completed by 3 workers. For a well aligned model, the scores under 2AFC and Clustering should be close to 1, while for a non-aligned model scores under 2AFC should be close to 0, and scores under Clustering should be close to a random guess (i.e., about 33%). We see that LPIPS (with different backbone nets, e.g., AlexNet, VGG) orders models similar to human annotators for both the survey setups, thus showing that it’s a reliable proxy.

**Reliability of Human Annotators** In Table 1, we make three major observations: 1) variance between different annotators is very low; 2) scores under Human 2AFC and Human Clustering order different models similarly; and finally, 3) even though accuracy drops for the “hard” version of ImageNet task, the relative ordering of models remains the same. These observations indicate that alignment can be reliably measured by generating IRIs and does not depend on bias in annotators. Note that AMT experiments were only performed on IRIs generated using the regularizer-free loss in Eq 2.

**Impact of regularizer** Table 2 shows the results of Alignment (Eq 1) for different regularizers for IRI generation. We evaluated multiple architectures of both standard and adversarially trained [37] CIFAR10 and ImageNet models. We find that under different types of regularizers, the alignment of models can look very different. We also see that adversarial regularizer makes alignment bad for almost all models, thus showing that for the worst pick of IRIs the alignment between learned invariances and human invariances has a lot of room for improvement. Conversely, the human-aligned regularizer overestimates the alignment.

**Impact of X** In the case of OOD targets (x_t) we see that humans are still able to faithfully judge similarity, yielding the same ranking of models as in-distribution targets. Some results for human judgements about similarity of IRIs for out of distribution samples are shown in Table 5, Appendix A.3. As seen in Fig 4 (Appendix A.3), human-aligned regularizer does not work well for reconstructing noisy targets. This is because such regularizers explicitly remove high-frequency features from reconstructions [43] and thus struggle to meaningfully reconstruct targets that contain high-frequency features. Hence, all results in Table 5, Appendix A.3 are reported on IRIs generated using regularizer-free loss.

**Impact of x_0** We repeat some of the experiments with other starting points for Eq 2 and find that results are generally not sensitive to the choice of x_0. Results are included in Appendix A.5.

3 What Contributes to Learning Invariances Aligned with Humans

There have been many enhancements in the deep learning pipeline that have lead to remarkable generalization performance [33, 22, 60, 58, 6, 30, 26]. In recent years there have been efforts to understand how invariances in representations learnt by such networks align with those of humans [16, 23, 13]. However, how individual components of the deep learning pipeline affect the invariances learned is still not well understood. Prior works claim that adversarially robust models tend to learn representations with a “human prior” [27, 11, 56]. This leads to the question: how do other factors such as architecture, training paradigm, and data augmentation affect the invariances of representations? We explore the answer to these questions in this section. As shown in Section 2.4, the adversarial regularizer consistently gives bad alignment for all models – while this shows that there’s room for improvement in the learned representations, it’s not useful in comparing different components. We report results for all models on IRIs generated using the adversarial regularizer in Appendix A.4 and show that (almost) all models evaluated in our work show bad alignment when measured using adversarial regularizer. As also noted in Section 2.4, the human-aligned regularizer overestimates alignment and often has a string ‘bias’ enforced by the regularizer. All evaluations in this section are thus based on regularizer-free IRIs.

\[4\] This was conducted only using IRIs from regularizer-free inversion.
3.1 Architectures and Loss Functions

We test the alignment of different DNNs trained using various loss functions – standard cross-entropy loss, adversarial training (AT), and variants of AT (TRADES [66], MART [64]). Both TRADES and MART have two loss terms – one each for clean and adversarial samples, which are balanced via a hyperparameter $\beta$. We report results for multiple values of $\beta$ in Fig 3a and find that the alignment of standard models (blue squares) is considerably worse than the robust ones (triangles and circles). However, the effect is also influenced by the choice of model architecture, e.g., for CIFAR10, for all robust training losses, VGG16 has significantly lower alignment than other architectures.

3.2 Data Augmentation

Hand-crafted data augmentations are commonly used in deep learning pipelines. If adversarial training – which augments adversarial samples during training – generally leads to better aligned representations, then how do hand-crafted data augmentations affect invariances of learned representations? For adversarially trained models, we try with and without the usual data augmentation (horizontal flip, color jitter, and rotation). Since standard models trained with usual data augmentation show poor alignment (Section 3.1), we try stronger data augmentation (a composition of random flip, color jitter, grayscale and gaussian blur, as used in SimCLR [6]) to see if hand-crafted data augmen-

| ADV. TRAINING | RESNET18 | DENSENET121 | VGG16 | INCEPTIONV3 |
|---------------|----------|-------------|-------|-------------|
| USUAL DATA AUG | 76.50±15.91 | 93.50±9.60 | 0.25±0.43 | 24.25±25.17 |
| NO DATA AUG | 30.00±12.02 | 93.75±8.20 | 1.00±1.73 | 12.25±20.08 |

| STANDARD | RESNET18 | DENSENET121 | VGG16 | INCEPTIONV3 |
|----------|----------|-------------|-------|-------------|
| STRONG DATA AUG | 0.00±0.00 | 1.00±1.73 | 0.00±0.00 | 0.00±0.00 |
| USUAL DATA AUG | 0.00±0.00 | 0.00±0.00 | 0.00±0.00 | 0.00±0.00 |

Table 3: [CIFAR10 Models; Effect of Data Augmentation] For certain models, e.g., adversarially trained resnet18, data augmentation is crucial in learning aligned representations.
tations can improve alignment. Table 3 shows how hand-crafted data augmentation can be crucial in learning aligned representations for some models (e.g., adversarially trained ResNet18 benefits greatly from data augmentation). In other cases data augmentation never hurts the alignment. We also see that standard models do not gain alignment even with stronger hand-crafted data augmentations. CIFAR100 and ImageNet results can be found in Appendix C with similar takeaways.

3.3 Learning Paradigm

Since data augmentations (both adversarial and hand-crafted) along with residual architectures (like Resnet18) help alignment, self-supervised learning (SSL) models – which explicitly rely on data augmentations – should learn well aligned representations. This leads to a natural question: how do SSL models compare with the alignment of supervised models? SimCLR [6] is a widely used contrastive SSL method that learns 'meaningful' representations without using any labels. Recent works have built on SimCLR to also include adversarial data augmentations [29, 5]. We train both the standard version of SimCLR (using a composition of transforms, as suggested in [5]) and the one with adversarial augmentation on CIFAR10 and compare their alignment with the supervised counterparts. More training details are included in Appendix C. Additionally, we also train SimCLR without the color distortion transforms – which were identified as key transforms by the authors [6] – to see how transforms that are crucial for generalization affect alignment. Fig 3b shows the results when comparing self-supervised and supervised learning. We see that SimCLR when trained with both hand-crafted and adversarial augmentations has the best alignment, even outperforming the best adversarially trained supervised model in initial and middle epochs of training. We also see that removing color based augmentations (DA - color) does not have a significant impact on alignment, thus showing that certain DA can be crucial for generalization but not necessarily for alignment.

To summarize, various learning parameters in addition to robust training affect representation alignment, and the nature of these effects is quite nuanced. While we provide initial insights, we leave a more comprehensive study of the effects of these training parameters on alignment for future work.

4 Related Work

Robust Models Several methods have been introduced to make deep learning models robust against adversarial attacks [46, 44, 52, 19, 62, 37, 66, 64, 7]. These works try to model a certain type of human invariance (small changes input that do not change human perception) and make the model also learn such an invariance. Our work on the other hand, aims to evaluate what invariances have already been learned by a model and how they align with human perception. Interestingly, models that are robust to adversarial perturbations were also found to learn features that align with human perception [11] and possess perceptually aligned gradients [27, 56]. In our work, we systematically study alignment and show that many other factors (discussed in Section 3) are also important to learn better aligned representations.

Representation Similarity There has been a long standing interest in comparing neural representations [34]. More recently, there have been advancements that efficiently calculate the similarity of learned representations of two neural nets [48, 39, 31, 63]. A similar line of work explores the alignment of features across networks [36]. While these works are related to ours in that they compare two systems of cognition, they assume complete white-box access to both neural networks and are thus able to fully analyze both sets of representations. These methods cannot be applied to our case since we want to measure the similarity between an artificial neural network and a human neural network, with only black box access to the latter.

DNNs and Human Perception Neural networks have been used to measure quality [11, 15] and perceptual closeness [67] in the image space in. Recent work has looked at measuring the alignment of human and neural network perception by eliciting similarity judgements from humans and comparing it with the outputs of neural nets [50]. Our work, however, explores alignment in the opposite direction, i.e., we measure if inputs that a network sees the same are also same for humans. Thus, the kind of similarity we look for lies entirely in the pixels of the image, whereas [50] check if contextual similarity of images (e.g., a cigarette and beer bottle being similar because both have a shared context of being age restricted) is manifested in DNNs. [13, 11] are closest to our work as they also evaluate alignment from model to humans, however as discussed in Section C, these works do not discuss the
effects of loss function used to generate IRIs and neither do they contribute to an understanding
of what components in the deep learning pipeline lead to learning human-like invariances.

**Explainability/Interpretability** Many prior works that evaluate learned representations often do so
with the goal of explaining the learned features of a DNN [38, 43, 28, 8, 9]. While our work heavily
uses techniques proposed in these works, our goal is to measure how invariances in DNNs align with
human perception.

5 Conclusion and Broader Impacts

Our work offers insights into how measures of alignment can vary based on different evaluation
schemes. Still, we believe that when it is done carefully, measuring alignment is a useful model
evaluation tool which can give insights beyond those offered by traditional metrics such as clean and
robust accuracy, thereby enabling better alignment of models with humans which will be broadly
beneficial. We recognize that there are potentially worrying use cases against which we must be
vigilant, such as taking advantage of alignment to advance work on deceiving humans. Human
perception is complex, nuanced and discontinuous [59], which poses many challenges in measuring
alignment of DNNs with human perception [20]. In this work, we take a step towards defining and
measuring alignment of DNNs with human perception. Our proposed method is a necessary but not
sufficient condition for alignment with human perception and, thus, must be used carefully and be
supplemented with other checks, including domain expertise. By presenting this method, we hope for
a better understanding and auditing of DNNs.

Acknowledgements

AW acknowledges support from a Turing AI Fellowship under grant EP/V025379/1, The Alan
Turing Institute, and the Leverhulme Trust via CFL. VN, AM, and KPG were supported in part by an
ERC Advanced Grant “Foundations for Fair Social Computing” (no. 789373). VN and JPD were
supported in part by NSF CAREER Award IS-1846237, NSF D-ISR Award #2039862, NSF Award
CCF-1852352, NIH R01 Award NLM013039-01, NIST MSE Award #20126334, DARPA GARD
#HR00120200007, DoD WHS Award #HQ003420F0035, ARPA-E Award #4334192 and a Google
Faculty Research Award. BCL was supported by Wellcome Trust Investigator Award WT106931MA
and Royal Society Wolfson Fellowship 183029. All authors would like to thank Nina Grgić-Hlača
for help with setting up AMT surveys.

References

[1] Seyed Ali Amirshahi, Marius Pedersen, and Stella X Yu. Image quality assessment by comparing
cnn features between images. *Journal of Imaging Science and Technology*, 60(6):60410–1,
2016.
[2] Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *Proceedings
of the European conference on computer vision (ECCV)*, pages 456–473, 2018.
[3] 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.
[4] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in
commercial gender classification. In *Conference on fairness, accountability and transparency*,
pages 77–91. PMLR, 2018.
[5] Tianlong Chen, Sijia Liu, Shiyu Chang, Yu Cheng, Lisa Amini, and Zhangyang Wang. Ad-
versarial robustness: From self-supervised pre-training to fine-tuning. In *Proceedings of the
IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 699–708, 2020.
[6] 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.
[7] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized
smoothing. In *International Conference on Machine Learning*, pages 1310–1320. PMLR, 2019.
[8] Alexey Dosovitskiy and Thomas Brox. Generating images with perceptual similarity metrics based on deep networks. *Advances in neural information processing systems*, 29:658–666, 2016.

[9] Alexey Dosovitskiy and Thomas Brox. Inverting visual representations with convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4829–4837, 2016.

[10] Logan Engstrom, Andrew Ilyas, Hadi Salman, Shibani Santurkar, and Dimitris Tsipras. Robustness (python library), 2019.

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

[12] William Falcon. Pytorch lightning. https://github.com/PyTorchLightning/pytorch-lightning, 2019.

[13] Jenelle Feather, Alex Durango, Ray Gonzalez, and Josh McDermott. Metamers of neural networks reveal divergence from human perceptual systems. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[14] Gustav Theodor Fechner. Elements of psychophysics, 1860. 1948.

[15] Fei Gao, Yi Wang, Panpeng Li, Min Tan, Jun Yu, and Yani Zhu. Deepsim: Deep similarity for image quality assessment. *Neurocomputing*, 257:104–114, 2017.

[16] 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. *arXiv preprint arXiv:1811.12231*, 2018.

[17] Tal Golan, Prashant C. Raju, and Nikolaus Kriegeskorte. Controversial stimuli: Pitting neural networks against each other as models of human cognition. *Proceedings of the National Academy of Sciences*, 117(47):29330–29337, 2020.

[18] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2015.

[19] Shixiang Gu and Luca Rigazio. Towards deep neural network architectures robust to adversarial examples. *arXiv preprint arXiv:1412.5068*, 2014.

[20] Olivia Guest and Bradley C Love. What the success of brain imaging implies about the neural code. *Elife*, 6:e21397, 2017.

[21] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[23] Katherine Hermann, Ting Chen, and Simon Kornblith. The origins and prevalence of texture bias in convolutional neural networks. *Advances in Neural Information Processing Systems*, 33:19000–19015, 2020.

[24] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007.

[25] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems*, 2019.

[26] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[27] Simran Kaur, Jeremy M. Cohen, and Zachary C. Lipton. Are perceptually-aligned gradients a general property of robust classifiers? 2019.
[28] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In International conference on machine learning, pages 2668–2677. PMLR, 2018.

[29] Minseon Kim, Jihoon Tack, and Sung Ju Hwang. Adversarial self-supervised contrastive learning. arXiv preprint arXiv:2006.07589, 2020.

[30] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.

[31] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In International Conference on Machine Learning, pages 3519–3529. PMLR, 2019.

[32] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[33] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 25, 2012.

[34] Aarre Laakso and Garrison Cottrell. Content and cluster analysis: assessing representational similarity in neural systems. Philosophical psychology, 13(1):47–76, 2000.

[35] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. nature, 521(7553):436–444, 2015.

[36] Yixuan Li, Jason Yosinski, Jeff Clune, Hod Lipson, and John Hopcroft. Convergent learning: Do different neural networks learn the same representations?, 2016.

[37] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks, 2019.

[38] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them, 2014.

[39] Ari S. Morcos, Matthra Raghu, and Samy Bengio. Insights on representational similarity in neural networks with canonical correlation. In Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada, pages 5732–5741, 2018.

[40] Alexander Mordvintsev, Christopher Olah, and Mike Tyka. Inceptionism: Going deeper into neural networks, 2015.

[41] Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 427–436, 2015.

[42] Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom in: An introduction to circuits. Distill, 2020. https://distill.pub/2020/circuits/zoom-in.

[43] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. Distill, 2017. https://distill.pub/2017/feature-visualization.

[44] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In Proceedings of the 2017 ACM on Asia conference on computer and communications security, pages 506–519, 2017.

[45] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In 2016 IEEE European symposium on security and privacy (EuroS&P), pages 372–387. IEEE, 2016.

[46] Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In 2016 IEEE symposium on security and privacy (SP), pages 582–597. IEEE, 2016.

[47] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, and R. Garnett, editors, Advances in Neural Information Processing Systems 32, pages 8024–8035. Curran Associates, Inc., 2019.
[48] Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS’17, page 6078–6087, 2017.

[49] 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.

[50] Brett D. Roads and Bradley C. Love. Enriching imagenet with human similarity judgments and psychological embeddings. In IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021, pages 3547–3557. Computer Vision Foundation / IEEE, 2021.

[51] Amir Rosenfeld, Richard Zemel, and John K Tsotsos. The elephant in the room. arXiv preprint arXiv:1808.03305, 2018.

[52] Andrew Slavin Ross and Finale Doshi-Velez. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In Thirty-second AAAI conference on artificial intelligence, 2018.

[53] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. International journal of computer vision, 115(3):211–252, 2015.

[54] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. arXiv preprint arXiv:1911.08731, 2019.

[55] Hadi Salman, Andrew Ilyas, Logan Engstrom, Ashish Kapoor, and Aleksander Madry. Do adversarially robust imagenet models transfer better? In Advances in Neural Information Processing Systems, 2020.

[56] Shibani Santurkar, Andrew Ilyas, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Image synthesis with a single (robust) classifier. In Advances in Neural Information Processing Systems, volume 32, 2019.

[57] Ali Shafahi, Mahyar Najibi, Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! arXiv preprint arXiv:1904.12843, 2019.

[58] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.

[59] Brian J Stankiewicz and John E Hummel. Categorical relations in shape perception. Spatial vision, 10(3):201–236, 1996.

[60] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1–9, 2015.

[61] Rohan Taori, Achal Dave, Vaishaal Shankar, Nicholas Carlini, Benjamin Recht, and Ludwig Schmidt. Measuring robustness to natural distribution shifts in image classification. In Advances in Neural Information Processing Systems, 2020.

[62] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. arXiv preprint arXiv:1705.07204, 2017.

[63] Liwei Wang, Lunjia Hu, Jiayuan Gu, Yue Wu, Zhiqiang Hu, Kun He, and John Hopcroft. Towards understanding learning representations: To what extent do different neural networks learn the same representation. arXiv preprint arXiv:1810.11750, 2018.

[64] Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu. Improving adversarial robustness requires revisiting misclassified examples. In International Conference on Learning Representations, 2019.

[65] Ross Wightman. Pytorch image models. https://github.com/rwightman/pytorch-image-models, 2019.
[66] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In International Conference on Machine Learning, pages 7472–7482. PMLR, 2019.

[67] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In CVPR, 2018.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope? [Yes] We summarize our contribution towards the end of Section 1
   (b) Did you describe the limitations of your work? [Yes] See Section 5
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Section 5
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]

2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]

3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Code will be submitted with a README in the supplementary material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Appendix B
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] All plots in the paper have error bars
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix B

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Appendix B
   (b) Did you mention the license of the assets? [Yes] Wherever possible, we mention the license of the used assets in Appendix B
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you’re using/curating? [Yes] Consent was obtained via the data curation procedures for the given datasets [2, 53]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No]

5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [Yes] See Fig 2
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [Yes] We received clearance from our institute’s Ethical Review Board
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [Yes] See Section 2.1
We allotted

We estimated a completion time of about 30

\[ \text{[Adversarial IRIs]} \]

Table 4: [Adversarial IRIs] We observe that using the adversarial regularizer (described in Section 2.2) makes alignment for all models look bad. AT = Adversarial Training, DA = Data Augmentations. For details about standard SimCLR DA and DA without color, see Section C.

A Measuring Human Alignment via Representation Inversion

A.1 Measuring Human Perception Similarity

We recruited AMT workers with completion rate \( \geq 95\% \) and who spoke English. To further ensure that the workers understood the task, we added attention checks. For 2AFC task, this meant making the query image as the same image as one of the images in the option. In clustering setting, this meant making the image on the row same as one of the images in the columns. All the workers who took our survey passed the attention checks.

We estimated a completion time of about 30 minutes for each survey and thus paid each worker 7.5$. We allotted 60 minutes per survey, so workers are not forced to rush through the survey. Most of the workers were able to complete the task in less than 20 minutes. Our study was approved by the Ethical Review Board of our institute.

ImageNet Clustering Hard In order to create the hard ImageNet clustering task we use the human annotations of similarity between ImageNet images collected by ImageNet-HSJ authors [50]. This contains a matrix \((M)\) of similarity scores for each image in ImageNet validation set, where \(M_{ij}\) is an indication for similarity between \(i^{th}\) and \(j^{th}\) images. For each image \(i\) (randomly picked), we sample two more images that are the most similar to \(i\) as per \(M_i\). This creates a task that is much harder to perform for human annotators since the images on the columns look perceptually very similar (See Fig 2b for an example).

A.2 Using LPIPS as a proxy for \(g_{\text{human}}\)

In order to ensure that LPIPS [67] is a reliable proxy to simulate human perception we measured if LPIPS could simulate human annotators on two perceptual similarity task setups: 2AFC and Clustering, as described in Section C.1 For 2AFC, this meant using LPIPS to measure the distance
Table 5: [CIFAR10 and ImageNet In-Distr vs OOD Survey Results] We observe that even on OOD samples that look like noise, humans can still bring out the relative differences between models, e.g., densenet121 on CIFAR10 is still ranked best aligned model on targets sampled from both kinds of noise. Reduced accuracy of humans on noise shows that this identifying similarities between IRIs on OOD samples is a harder task than with in-distribution target samples.

between the query image and the two images shown in the options and then matching the query image to the one with lesser LPIPS distance. And similarly in Clustering, for each image on the row, we used LPIPS to measure its distance from each of the 3 images in the column and then matched it to the closest one. Our results (Table 1 in main paper) show that this can serve as a good proxy for human perception similarity. For all our experiments, we report averagae over 4 different LPiPS backbones: ImageNet trained Alexnet & VGG16, and both of the Imagenet trained Alexnet & VGG16 finetuned by the authors for perceptual similarity [https://github.com/richzhang/PerceptualSimilarity]

A.3 Role of Input Distribution

Fig 4 shows some examples of inputs sampled from different gaussians (the lighter ones are sampled from \( \mathcal{N}(0.5, 2) \) and darker ones from \( \mathcal{N}(0, 1) \)). We find that completing 2AFC and Clustering tasks on inputs that look like noise to humans is a qualitatively harder task than when doing this on in-distribution target samples.

However, remarkably, we observe that humans are still able to bring out the differences between different models, even when given (a harder) task of matching re-constructed noisy inputs. Table 5 shows the results of surveys conducted with noisy target samples. It’s worth noting that the accuracy of humans drop quite a bit from in-distribution targets, thus indicating that this is indeed a harder task.

A.4 Regularizers

For the human-aligned regularizers, we use the ones discussed in [43]. These fall into three broad categories: frequency penalization, transformation robustness, and pre-conditioning.

- **Frequency Penalization**: The goal is to explicitly penalize high-frequency features in the reconstruction \( (x_r) \). This is done by adding a regularizer of the form \( R(x) = TV(x) + ||x||_p \), where \( TV \) is the total variation and \( p = 1 \) [38]. A similar effect of frequency penalization can also be done by ensuring robustness of \( x_r \) to blurring, i.e., \( R(x) = ||x - \text{StopGradient(blur}(x))||_2^2 \) [41].

- **Transformation Robustness**: This ensures that \( x_r \) is such that the representation is same even if we slightly transform \( x_r \). This is achieved by replacing \( x \) with \( T(x) \) in Eq 2. We use \( T \) as a composition of color jitter, random scaling, and random rotation.
Figure 4: **[In vs Out of Distribution Samples]** Examples of reconstructions of in-distribution (bottom row) sample vs out-of-distribution samples for an ImageNet trained ResNet50 using the three regularizers mentioned in Section 2.2. Here the OOD targets are sampled from two separate random gaussians $\mathcal{N}(0, 1)$ (top row) and $\mathcal{N}(0.5, 2)$ (second row). We see that similar to the in-distribution sample, regularizer-free and adversarial inversions result in $x_r$ resembling and differing from $x_t$ respectively. Interestingly, for human-aligned regularizer, which explicitly tries to remove high-frequency features from $x_r$, fails to reconstruct an $x_t$ that itself consists of high-frequency features.

- **Pre-conditioning:** This involves taking gradient steps in the fourier domain, which decorrelates the pixels in $x_r$.

For our experiments we find that transformation robustness generates the best looking $x_r$ and thus we report results under human-learning regularizer based on $x_r$ generated using transformation robustness during representation inversion.

For adversarial regularizer, we report results in Table 4 and find that such a regularizer can make almost all models look like they have bad alignment.

### A.5 Role of $x_0$

We additionally report results for the adversarial regularizer where IRIs were generated from a separate seed. While experiments in the main paper reported for a seed sampled from $\mathcal{N}(0, 1)$, we report results here for a seed sampled from $\mathcal{N}(0, 0.01)$ in Table 4 and find that regardless of seed, adversarial regularizer makes all models look bad.

### B Model, Code, Assets, and Compute Details

#### B.1 Code and Assets

In our code we make use of many open source libraries such as timm [65], pytorch [47], pytorch-lightning [12], numpy [21], robustness [10], matplotlib [24], timm, pytorch-lightning and have an Apache 2.0 license. Numpy has a BSD 3-Clause License. Robustness has an MIT license. PyTorch’s license can be found here: https://github.com/pytorch/pytorch/blob/master/LICENSE and matplotlib’s here: https://github.com/matplotlib/matplotlib/blob/main/LICENSE/LICENSE. All these licenses allow free use, modification and distribution. We use publicly available academic datasets CIFAR10/100 [32] and ImageNet [53].

#### B.2 Models

**Supervised** We used VGG16, ResNet18, Densenet121 and InceptionV3 for experiments on CIFAR10 and CIFAR100. The “robust” version of these models were trained using adversarial training [37], with an $\ell_2, c$ of 1. All these models were trained using the standard data augmentations(a composition of RandomCrop, RandomHorizontalFlip, ColorJitter, RandomRotation). For ImageNet, we
used VGG16, ResNet18 and ResNet50 and the “robust” versions of these models were taken from [55] with an $\ell_2, \epsilon$ of 3. ImageNet models used slightly different data augmentations RandomHorizontalFlip, ColorJitter, Lighting

**Self-supervised** We used SimCLR [6] to train a ResNet18 backbone on CIFAR10 and CIFAR100. More details about different types of data augmentations in Section C.

**ImageNet** We used VGG16 (with batchnorm), ResNet18 and ResNet50 for ImageNet. The “robust” versions of these models were taken from [55], who trained these models using adversarial training with an $\ell_2$ epsilon of 3.

### B.3 Compute Details

We used our institute’s GPU cluster to run all experiments. Since our experiments involve standard models and datasets, these can be run on any hardware supported by PyTorch. In our case, we used 5 machines with 2 V100 Nvidia Tesla GPUs (32GB each, volta architecture) and a Nvidia dgx machine with 8 Nvidia Tesla P100 GPUs (16GB each). We estimate a total of 500+ GPU hours.

### C What Contributes to Good Alignment

**SimCLR training details** We used data augmentations shown to work best by the authors (a composition of RandomHorizontalFlip, ColorJitter, RandomGrayscale and GaussianBlur, as implemented in the original codebase [https://github.com/google-research/simclr]). We also train SimCLR models without the color augmentations (i.e. only RandomHorizontalFlip and GaussianBlur). Since color transforms were crucial for obtaining representations with good generalization performance, we wanted to analyze how removing augmentations crucial for generalization impacts alignment. Finally, we also train a variant of SimCLR with adversarial data augmentations, as proposed in some recent works [29, 5]. As opposed to traditional adversarial training, here we generate adversarial data augmentations for a model (g) by solving the following maximization for each input $x$:

$$\arg\max_{x'} ||g(x') - g(x)||_2 \text{ st } ||x - x'||_2 \leq \epsilon$$

For our experiments $\epsilon = 1$ for CIFAR10 and $\epsilon = 3$ for ImageNet (similar to supervised models).

**Architectures and Loss Function, CIFAR100 & ImageNet** Fig 6 shows results for CIFAR100 and ImageNet for standard and robust training. Similar to previous works, we find that robust models are better aligned with human perception. Interestingly, we find that the variance between different architectures that we observed for CIFAR10 does not exist for CIFAR100 and ImageNet, i.e., regardless of architecture, robustly trained models are well aligned with human perception. Indicating that (unsurprisingly) training dataset also plays a major role in alignment.

**SimCLR, CIFAR100** Fig 5 shows alignment of different types of SimCLR models throughout training. We observe a similar trend as CIFAR10, where adversarial data augmentation improves alignment.

### C.1 Data Augmentation, CIFAR100 & ImageNet

Since re-training ImageNet models with adversarial training is very resource intensive, we train ImageNet models using Free Adversarial Training (Free AT) [57]. Free AT only has an implementation for $\ell_{\infty}$ threat model, hence we train these models with $\ell_{\infty}, \epsilon = 4/255$ (for both with and without data augmentation). Table 6 shows that similar to CIFAR10, for some models, like ResNet18, data augmentation is crucial in learning aligned representations (despite being trained to be adversarially robust). For other models, data augmentation never hurts alignment (except InceptionV3 for CIFAR100).
Figure 5: ResNet18 backbone trained using SimCLR on CIFAR100.

| CIFAR100  | RESNET18 | DENSENET121 | VGG16 | INCEPTIONV3 |
|-----------|----------|-------------|-------|-------------|
| **USUAL DATA AUG** | 82.50±20.85 | 88.00±14.51 | 58.75±32.35 | 69.25±26.39 |
| **NO DATA AUG**  | 81.50±15.58 | 89.75±15.01 | 46.25±32.25 | 84.75±13.70 |

| ImageNet  | RESNET18 | RESNET50 |
|-----------|----------|----------|
| **USUAL DATA AUG** | 13.00±22.52 | 0.00±0.00 |
| **NO DATA AUG**  | 0.75±1.30 | 0.00±0.00 |

Table 6: **CIFAR100 & ImageNet Models all trained to be adversarially robust; Effect of Data Augmentation** Similar to CIFAR10, data augmentation is crucial for adversarially trained resnet18 to learn aligned representations.

Figure 6: Role of Loss Function in Alignment; CIFAR100 (left), and ImageNet (right)