EXPLORING PERCEPTUAL STRAIGHTNESS IN LEARNED VISUAL REPRESENTATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Humans have been shown to use a "straightened" encoding to represent the natural visual world as it evolves in time (Hénaff et al. 2019). In the context of discrete video sequences, "straightened" means that changes between frames follow a more linear path in representation space at progressively deeper levels of processing. While deep convolutional networks are often proposed as models of human visual processing, many do not straighten natural videos. In this paper, we explore the relationship between network architecture, robustness, biologically-inspired filtering mechanisms, and representational straightness in response to time-varying input; we identify curvature as a useful way of evaluating neural network representations. We find that (1) adversarial training leads to straighter representations in both CNN and transformer-based architectures but (2) this effect is task-dependent, not generalizing to tasks such as segmentation and frame-prediction, where straight representations are not favorable for predictions. Finally, (3) biologically-inspired elements increase straightness in the early stages of a network, but do not guarantee increased straightness in downstream layers of CNNs. Our results suggest that the ability of a model to straighten is a useful and easily computed measure of representational robustness and stability, as well as a marker of similarity between human and machine representations.

1 INTRODUCTION

Visual input from the natural world evolves over time, and one can think of that change over time as a trajectory in some representation space. For humans, this trajectory changes at different levels of processing, from input at the retina to brain regions such as V1 and finally to perception, as Figure 1 illustrates. We can ask about the shape of that trajectory and consider that there might be advantages to a straighter, less curved, trajectory. Hénaff et al. (2019) demonstrated that trajectories are straighter in human perceptual space than in pixel space. They suggested that a straighter representation may be useful for visual tasks that require extrapolation, such as predicting the future.

Learning a useful visual representation is one of the major goals of computer vision. Properties like temporal stability, robustness to transformations, and task flexibility – all of which characterize human vision – are often desirable in computer vision representations. Yet, many existing computer vision models still fail to capture aspects of human vision, despite achieving high accuracy on visual tasks like recognition (Feather et al., 2019; Hénaff et al., 2019). In Hénaff et al. (2019) it was found that, while biologically-inspired V1-like transformations yield straighter representations compared to the input domain, popular computer vision models such as the original ImageNet-trained AlexNet (Krizhevsky et al., 2017) do not.

In an effort to achieve these favorable human-vision properties for computer vision models, there has been much work dedicated to incorporating various aspects of human vision into deep neural networks. These include modifying network architectures to mimic that seen in the human visual system Huang & Rao (2011), incorporating filter banks matching the receptive field properties of visual neurons Dapello et al. (2020), as well as enforcing activation properties similar to that seen in visual cortex with properties such as sparsity Wen et al. (2016). Another promising avenue in this area has been in modifying training with adversarial examples. In directly targeting areas of vulnerability, adversarially robust networks show increased representational robustness, more closely aligning them with their human counterparts (Engstrom et al., 2019b).

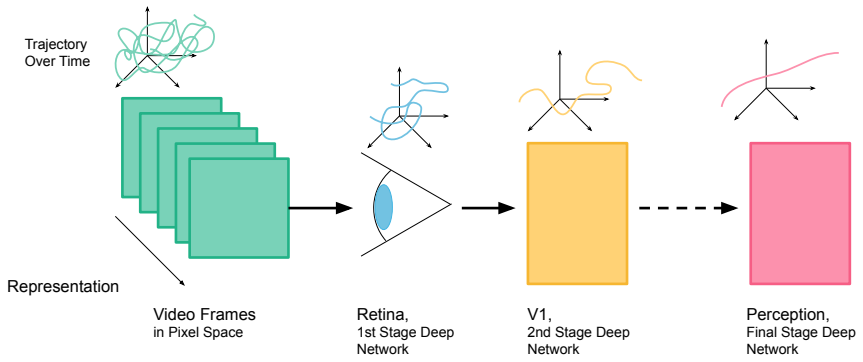

Figure 1: Illustration of the representation of a discrete video sequence becoming progressively straighter as information is processed through a visual processing pipeline, starting from the highly nonlinear trajectory of typical video frames in pixel space.

In this paper, we explore what makes some learned visual representations straighter than others. We evaluate a variety of network architectures, both biologically and non-biologically inspired, for representational straightness across layers; we then ask whether training for adversarial robustness in both CNN and transformer-based architectures may lead to the straighter representations generated by human vision. Because DNNs learn an early representation that differs from what is known about human vision, we also ask if hard-coding that early representation might lead to a trained network with more straightening downstream. Overall, we find that most deep learning models do not employ straightened representations; however, with modified training schemes (notably adversarial training), many models are capable of relying on a straightened representations. This capability to utilize a straightened representation may be useful for learning more robust and stable image and video processing systems.

2 Previous Work

Deep neural networks have been proposed as models of human visual processing, owing to their ability to predict neural response patterns (Yamins & DiCarlo, 2016; Rajalingham et al., 2015; Kell & McDermott, 2019). As such, there has been much effort to improve the alignment of deep networks to human vision by incorporating known aspects of the human visual system. Some of these include: simulating the multi-scale V1 receptive fields of early vision (Dapello et al., 2020), adding foveation using a texture-like representation in the periphery at a CNN's input stage (Deza & Konkle, 2020), and incorporating activation properties of visual neurons such as sparsity Olshausen & Field (1997); Wen et al. (2016). Predictive coding, often attributed to biological networks Huang & Rao (2011), has been incorporated into deep networks trained to perform tasks such as video frame prediction Lotter et al. (2016), using layers that propagate error-signals. The desire to evaluate the effectiveness of these techniques at creating models of the human visual system, motivated the creation of measures like BrainScore Schrimpf et al. (2020) that compare models to humans.

One important area in understanding how humans and DNNs differ lies in their response to adversarial examples (Elsayed et al., 2018; Ilyas et al., 2019; Feather et al., 2022; Dapello et al., 2021). Adversarial examples, which modify images with changes that are imperceptible to humans, can cause a network to misclassify an image (Goodfellow et al., 2014; Szegedy et al., 2013). Adversarial training (Madry et al., 2017) improves the misclassfication problem and has been suggested to help networks learn visual representations that are more perceptually aligned with humans Engstrom et al. (2019b); Ilyas et al. (2019). If adversarial training leads to models that are more aligned with human perception, perhaps they also learn straight representations like humans do. Adversarial training schemes, however, are not biologically plausible, and recent work has identified mechanisms that are better supported by vision science (Dapello et al., 2020; Guo et al., 2022; Dapello et al., 2021).

In the context of representation learning in computer vision, adversarially robust models have also been shown to do better at transfer learning than their non-adversarially robust counterparts (Davchev et al., 2019), and adversarially robust features can be used directly for tasks like image generation and in-painting (Santurkar et al., 2019). Adversarial training schemes have also been developed for tasks other than classification like semantic segmentation (Xu et al., 2021). In this paper, we build on work around adversarial robustness by evaluating if increasing this robustness leads to straightened representations like those found in human spatiotemporal processing.

There have been various efforts to incorporate constraints that reduce curvature and increase straightness in temporal representation learning. Slow feature analysis (Berkes & Wiskott, 2005) encourages a learned representation to change slowly over time, but does not encourage straightness explicitly. Another early attempt to add a straightness constraint to representation learning (Goroshin et al., 2015) focused on video frame prediction for unsupervised learning. We follow in the same vein of this work, evaluating straightness as a metric for encouraging a robust, stable, representation when learning to solve tasks.

3 METHODS

3.1 MEASURING STRAIGHTNESS AS LOW CURVATURE

Representational straightness can be evaluated as a reduction in curvature. For a temporal sequence, such as a video, curvature is defined as the angle between the vectors representing the *difference* between consecutive frames. Let $\mathbf{x}$ refer to a representation of a video of length T, with x_t being a representation of one frame of a video at time step t. The representation may be at any stage of the processing pipeline, from a vector of raw input pixels from the video frame, to the activations of a network's hidden layer. Then, v_t represents the difference between successive frames:

$$v_t = x_t - x_{t-1} \tag{1}$$

$$\hat{v}_t = \frac{v_t}{\|v_t\|} \tag{2}$$

We can find the curvature at time t by finding the angle between successive $\hat{v}_t$, which we call c_t:

$$c_t = \arccos\left(\hat{v}_t \cdot \hat{v}_{t+1}\right) \tag{3}$$

The global curvature of a video sequence is then simply the mean angle over all time steps. Note there are $T-1$ time steps due to curvature being evaluated on frame differences:

$$\text{Global curvature} := \frac{1}{T-1}\sum_{t=0}^{T-1} c_t \tag{4}$$

This is the formulation proposed by Hénaff et al. (2019). One can compute this global curvature for any representation of a video sequence over time, either on the vector of pixels (likely not very straight), or one can apply it to a representation of that video, e.g. at any layer of a neural network model. In this paper, we often report change in curvature with respect to the pixel-value input to show if a model's representation is more straight at later stages of processing. Change in curvature is the representational curvature minus the pixel curvature.

Note that curvature is distinct from simple cosine similarity in that curvature is calculated on frame differences (v_t), whereas cosine similarity depends on the angle between the frame vectors themselves (x_t). Curvature can be thought of as a first-order variant of cosine similarity (see Figure 2 and Section A.1). For a complete description of how curvature was determined for each model, see Section A.2.2.

4 CURVATURE AND ROBUSTNESS IN CONVOLUTIONAL OBJECT RECOGNITION MODELS

We tested a variety of models for output curvature, to investigate the relationship between model type and curvature of the output layer. As shown in Figure 3, we found non-adversarially trained image recognition models to have the highest output curvature. All adversarially trained models

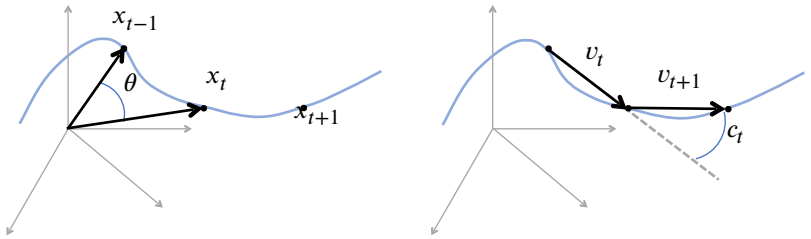

Figure 2: Illustration of how the curvature measure is distinct from cosine similarity. (Left) Three points are sampled along a trajectory in time (x_{t-1}, x_t, x_{t+1}). The angle θ between neighboring x samples is their cosine similarity. (Right) Curvature c_t is the angle between v_1 and v_{t+1}, where v_t is the difference between x_t and x_{t-1}.

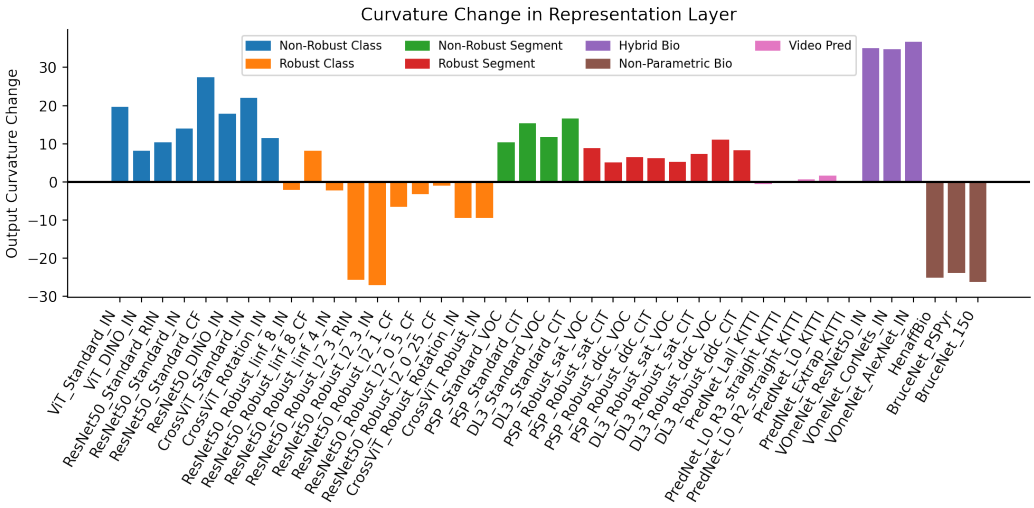

Figure 3: **Output Representation layer curvature for a variety of network architectures and training schemes.** The x-axis ticks describe different models and are formatted as: [architecture]_[training scheme]_[training dataset]. The datasets are IN:ImageNet (Deng et al., 2009), RIN:RestrictedImageNet (Santurkar et al., 2019),CF:CIFAR10 (Krizhevsky et al., 2014), VOC:PASCAL VOC 2012 (Everingham et al., 2010), CIT:Cityscapes (Cordts et al., 2016), KITTI:KITTI (Geiger et al., 2012). Exact model details can be found in the supplemental data. The y-axis shows curvature of each model at the output representation layer relative to the curvature of the input video. For image classification models the output representation is the penultimate layer (just before classification). This is sometimes referred to as the output layer.

have lower curvature than their non-adversarially trained counterparts, as well as overall, with the majority reducing output curvature *below* that of the input pixels. Self-supervised DINO (Caron et al., 2021) models have similar output curvature values to their supervised counterparts – despite DINO models having been shown to have more semantically meaningful feature correspondences.

4.1 Adversarial Attack Type, Strength, and Curvature

Given the increased straightness seen in adversarially robust image recognition / object detection models, we investigated the relationship between the type and strength of adversarial attack and the resulting curvature of the model's output. To evaluate the effect of these attacks on curvature, we compare a set of ResNet50 networks (He et al., 2016), trained on CIFAR-10 (Krizhevsky et al., 2014), ImageNet (Deng et al., 2009), and Restricted ImageNet (a subset of ImageNet (Engstrom et al., 2019b; Ilyas et al., 2019)), with and without adversarial training and measure the output

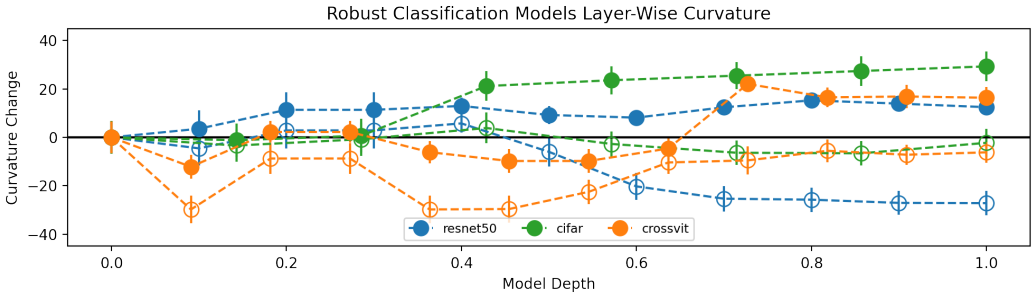

Figure 4: **Deep network models vary in curvature for each layer over model depth.** Curvature is shown for a variety of model types. Filled circles indicate non-adversarially trained models, and open circles indicate adversarially-trained robust models. All adversarial models show output curvature below zero, indicating output straightening. All non-adversarial models have output curvature above zero, indicating output representation is *less* straight than the input pixels; All adversarial models shown have output curvature below zero, indicating output representation is *more* straight than the input pixels.

curvature (Figure -5). The adversarially trained models are trained using projected gradient descent with l_2 or l_∞ norms at different perturbation levels (Madry et al., 2017; Engstrom et al., 2019a). We use ResNet50 adversarially trained with projected gradient descent because that model type has been shown to be more aligned with human perception (Engstrom et al., 2019b; Feather et al., 2022; Harrington & Deza, 2022). During projected gradient descent, adversarial examples are created to train the model to correctly classify them. The adversarial example is created by adding noise to an image. The amount of noise is bounded by a norm (usually l_2 or l_∞ set to a level ϵ). We include some models trained on CIFAR-10 to understand how perturbation level interacts with straightness (CIFAR-10 models are faster to train, so more models are available).

We find that, as predicted, output curvature is highest for non-adversarially trained networks. l_∞ attacks decrease output curvature, with larger values of ϵ leading to decreased curvature. l_2-attacked networks have the straightest output curvatures, however strength of attack does not greatly affect the output curvature.The l_2 norm models may achieve straighter representations than the l_∞ models because they allow for greater ϵ in training. Among l_2 models, higher ϵ values may need to be tested to see a difference in curvature among l_2 norm trained models. Overall, we show that robustness to larger perturbations leads to straighter representations.

Note that while we show that curvature is predictive of adversarial robustness, this does not translate to improved accuracy on the test set (See Section A.4). We argue however, that while curvature is not predictive of test accuracy directly, adversarial robustness is associated with many other model benefits, including representational stability and alignment with human perception.

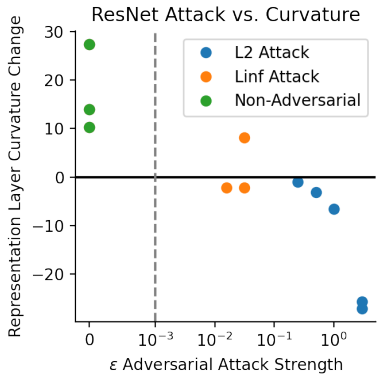

Figure 5: White-box adversarial attacks reduce curvature in the output layer of ResNet. Increased attack strength (ϵ) decreases curvature, with l_2 attacks most reducing curvature. Data on symlog x-scale to show $\epsilon = 0$: data to left of line on linear scale to show $\epsilon = 0$, and right of line on log scale.

5 Curvature in Visual Transformer Models for Object Recognition

Vision transformers (ViT) have very different architectures than CNNs. They replace convolutional elements with self-attention on visual tokens and achieve state of the art performance on a variety of visual tasks. In addition, these networks are more adversarially robust than standard-trained CNNs, which is attributed to their higher-level and more

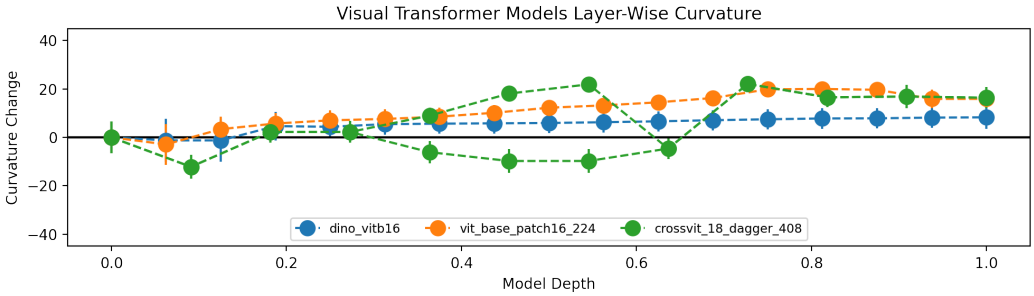

Figure 6: **Visual Transformer (ViT) models vary in curvature for each layer over model depth.** All ViT models (standard and DINO, base-patch-16) show output curvature above that of the pixel input. For DINO and standard ViT models, this increase in curvature is consistent throughout the layers. For CrossViT (18 dagger 408) the representation forks in the middle stage. We observe that one path increases in curvature, while the other decreases below input, but after recombination output is curved above pixel input.

generalizable features (Shao et al., 2021). Despite this, we find that standard-trained ViT (ViT Base, patch 16) does not reduce curvature on the natural video sequences tested (Fig 3). Self-supervised training of ViT (base, patch 16) using the DINO method (Caron et al., 2021) also results in a more curved representation than the input space. However, this DINO trained ViT has a more straight representation than the baseline ViT, despite the two models sharing the same architecture. This suggests that self-supervised training may be better than supervised training for obtaining straight representations.

No transformer model except for CrossViT daggar (Chen et al., 2021) reduced curvature in any layer compared to the pixel input (Fig 6). The structure of CrossViT models is unique in that they split into multiple paths, and the reduction in curvature we found interestingly only happens in one pathway of the model and does not persist after the two paths recombine. A possible explanation for this is that the CrossViT is able to straighten in the part of the model that splits because that pathway utilizes a multi-scale representation, a known aspect of human vision representations such as V1 that which we show in 8 to induce straightening.

6 Curvature and Robustness in Image Segmentation Models

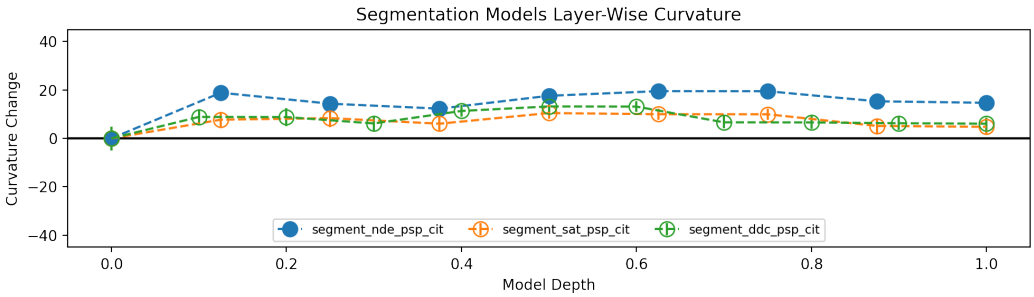

Figure 7: **Segmentation models vary in curvature for each layer over model depth.** Curvature is shown for various segmentation model types. Filled circles indicate non-adversarially trained models, and open circles indicate adversarially-trained robust models.

Output straightness over time should be a favorable, stabilizing property for object detection models, where objects often stay continuously in-frame over many frames of a video. For image segmentation models on the other hand, predictions lie close to low-level pixel space and change on a frame-by-frame basis. Therefore, for the task of predicting the dynamically changing segmentation

map of an input movie, straightness of the output representation would *not* be desirable (assuming the input movie contains motion). To test this prediction, we measured output curvature for a family of 12 image segmentation models from Xu et al. (2021), consisting of two different architectures, trained on two different datasets, with a non-adversarial as well as two different adversarial training schemes.

Unlike object detection models, where adversarial training greatly reduces output curvature, adversarial training in segmentation models only slightly reduces curvature compared to non-robust segmentation models (Figure 3), with all adversarially trained segmentation models displaying output curvature *higher* than the input pixel curvature. Investigating curvature change over model layers (Figure 7), reveals that this trend is consistent throughout all layers of the model. This highlights the importance of task on straightness, and supports the idea that output curvature reduction is not simply the result of any adversarial training scheme – only when temporal stability is beneficial does adversarial training promote straightness.

7 Curvature in Video Frame Prediction Models

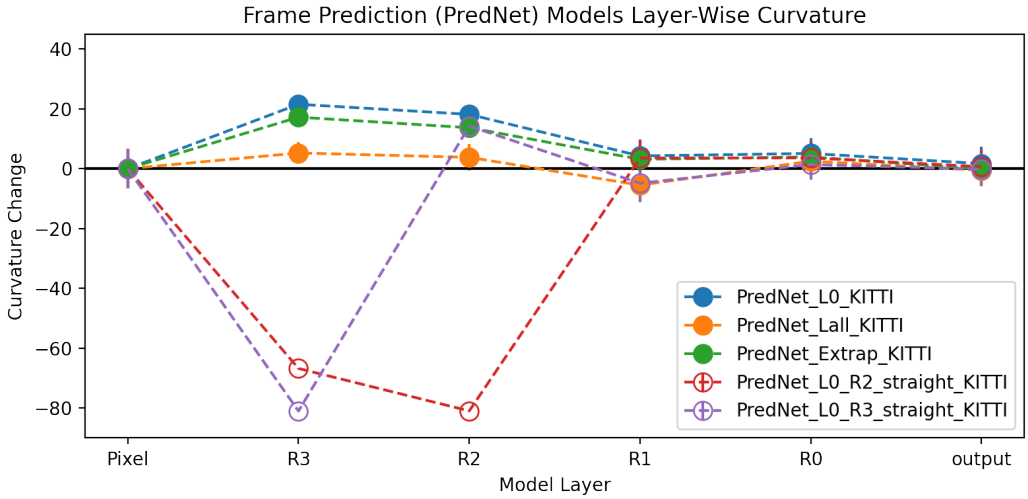

Figure 8: **Variants of PredNet models change in curvature for each layer over model depth.** Curvature is shown in filled circles for the following standard PredNet models: standard (PredNet_L0_KITTI), 5-frame future prediction (PredNet_Extrap_KITTI), and all-layer-loss (PredNet_Lall_KITTI). Shown in open circles are two PredNet models trained with straightness terms in the loss function (PredNet_L0_R2_straight_KITTI and PredNet_L0_R3_straight_KITTI). All models were trained on the KITTI dataset (Geiger et al., 2012).

Another DNN task for which output predictions are low level, is next-frame prediction for video sequences. As in image segmentation models, good predictions are expected to vary dynamically frame-to-frame, and representational straightness at and near the output layer is *not* a favorable quality in response to videos containing motion. Rather, for a frame-prediction network, a favorable quality would be equal curvature at the input and output. This is because for a well-performing model, the output prediction over many frames is the same as the input video; two identical videos have identical curvature. To evaluate this, we test the layer-evolution of curvature for three variants of PredNet (Lotter et al., 2016), a network inspired by predictive coding, which is trained to predict the next frame in a video sequence (Figure 8).

Indeed, for all pretrained variants of PredNet tested, the output layer maintains a very similar curvature value to that of its input frame, with final output curvature closer than all other models to that of the input (Fig 3). For all pretrained variants, the model curvatures strongly increase in the representation of the first model block, then re-straighten the representation throughout the rest of the network before returning to the original pixel curvature. We also investigated a variant of PredNet

which had been fine-tuned to predict multiple frames in the future (PredNet_Extrap_KITTI), which we find only slightly reduces model curvature, with a similar trajectory over layers.

For the variant of PredNet in which predictive error for all layers is incorporated into the loss during training (PredNet_Lall_KITTI), this increase is much less pronounced, pointing to a more temporally-stable representation for this network globally. This is in agreement with previous findings, showing that while this globally-trained PredNet slightly under-performs at the original frame-prediction task, when using its network backbone to perform a classification task, this least-curved PredNet model is the best performer. In other words, the network with the most straight representation performs best in the context of a different task, where temporal stability is favorable.

We also tested the utility of straightness as a training signal, adding a curvature constraint to the first (R3) and second (R2) layers of PredNet during training to create "straight" PredNet models. We evaluate how the constraint affects the model's internal representation and the final frame prediction. For both models, curvature is greatly reduced in R2 and R3 compared to the input curvature (Fig 8). Because of PredNet's top-down and bottom-up feedback structure, the constraint on layer R3 does not affect any other layers of the model. Although we find that it is possible to train PredNet to straighten and still get next-frame predictions that has lower mean square error (MSE) than the copying the previous frame, we are not able to achieve equivalent or better prediction MSE with the straight models compared to the original on KITTI (Geiger et al., 2012) (original L0 PredNet:0.00687 MSE, straight R2 PredNet: 0.00976 MSE, compared to previous frame: 0.02122 MSE). See Fig 13 for example predictions. Again, while these straightened PredNet variants do not achieve state of the art, we argue that these models may display other advantages such as robustness and similarity to human representations.

8 Curvature in Biologically Inspired Models

We investigated straightness for a variety of both parametric (learned) and non-parametric biologically-inspired models. Given straightness is thought to increase over progressively deeper layers of visual processing, we align these networks along the visual processing areas they are most closely matched to (Figure 9). Some models match to visual areas explicitly in their architecture like the non-parametric Henaffbio (Hénaff et al., 2021), which is a two-stage model based on center-surround filters follows by oriented Gabor filters, and Brucenet (Brown et al., 2021), which is based on summary statistics of a steerable pyramid. This explicit match to biological vision also holds for the early layers VOneNetCornets (Dapello et al., 2020), which combines a V1 filter front-end and Cornets as a backbone (Cornets being a shallow neural network that has layers matched to regions of the human brain (Kubilius et al., 2019)). For the adversarially trained Visual Transformer network CrossViTRotAdv (Berrios & Deza, 2022), these layers are those best matched by BrainScore (a benchmark for how similar a model is to the human brain at object recognition Schrimpf et al. (2020)), for V4 and IT layers (Schrimpf et al., 2020).

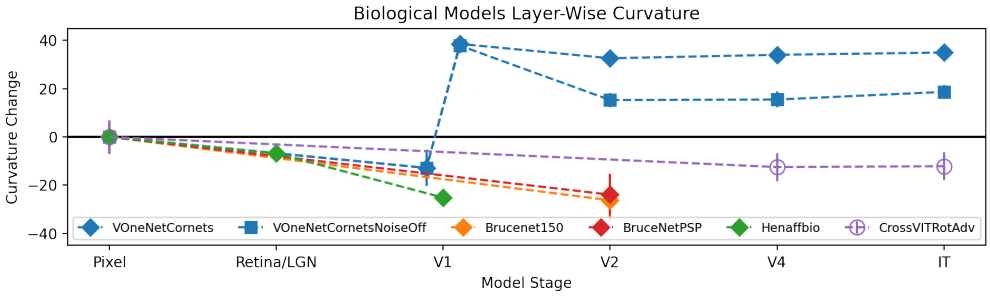

Figure 9: **Biologically-inspired network models result in straightened representations compared to input pixels.** Non-parametric multi-stage filter models Brucenet and Henaffbio progressively straighten. The VOneNetCornets model straightens the representation in its V1-block up until its noise layer, where curvature increases dramatically and does not recover in downstream convolutional blocks. Adversarially-trained CrossViT model shows progressive straightening.

For all biologically-inspired models except for VOneNetCornets, curvature progressively decreases through deeper network layers. For VOneNetCornets, curvature decreases up until the V1 layer in which a noise term capturing neural stochasticity is added; curvature then strongly increases, far above the pixel-curvature baseline. To determine if this increase in curvature was due to the added noise, we tested the same model with the noise term set to zero. While this reduced the downstream curvature after the V1 layer, this change did not eliminate the curvature increase present right after the V1 layer.

The increase in curvature for VOneNet at later layers suggest that making the front-end of a deep network more like biologically-inspired models does not suffice to get a straightened representation downstream in a deep network. This is interesting because VOneNet is is reported to be more adversarially robust to white-box attacks than a standard trained CNN. This suggests that adversarial *training*, not the property of adversarial robustness itself, leads to straightened representations in deep neural networks. Our finding supports Dapello et al. (2021) who found that the neural population geometry of adversarial trained models was noticeably different from VOneNet and other models trained with neural stochasticity mechanisms. However, it is puzzling that adversarial training, which is biologically implausible, would lead to straighter representations than biologically inspired mechanisms. More constraints or modifications may be needed to achieve straight representations with biologically plausible methods.

9 Discussion

We show that one can use the change in model representational curvature as a simple and computationally cheap metric for evaluating both image and video models across a variety of tasks. For a variety of image task models, we show that training with strong white-box adversarial attacks reduces output curvature. However, this is only true for tasks like image classification for which temporal stability is favorable, and not for semantic segmentation and video frame prediction. Moreover, we show that self-supervised training does not produce straightened representations. In evaluating both CNN and transformer-base architectures we show that curvature reduction is mainly a product of task and training procedure, not deep neural network architecture alone.

In evaluating curvature over layers in biologically inspired models, we show that biologically inspired mechanisms work to reduce curvature in a model's representation, even more so than adversarial training. However, the simple addition of non-parametric biologically inspired filtering mechanisms at the input of a model are insufficient to maintain output curvature.

In summary, we find that models with more known aspects of human vision like robustness have a greater ability to straighten natural videos. Our results suggests that the ability to straighten might be a useful and easily computed measure of how similar visual representations are to humans. These results identify representational curvature as a common thread between biologically inspired and adversarially robust models, and highlight the benefits and limitations of these techniques in creating temporally-stable representations.

10 Reproducibility

To reproduce our results, links to all the models analyzed can be found in the supplemental material in the Network Comparison Spreadsheet. Sources for the stimuli are in Sec A.2.2 and in the code base linked for the HenaffBio model in Network Comparison Spreadsheet. Straightness can be determined by calculating curvature using Sec 3 and A.1. The models, stimuli, and curvature function together should be sufficient to reproduce our results.

REFERENCES

Research Computing Center at University of Chicago. Chicago motion database, 2022. URL https://cmd.rcc.uchicago.edu.

Pietro Berkes and Laurenz Wiskott. Slow feature analysis yields a rich repertoire of complex cell properties. *Journal of vision*, 5(6):9–9, 2005.

William Berrios and Arturo Deza. Joint rotational invariance and adversarial training of a dual-stream transformer yields state of the art brain-score for area v4. *arXiv preprint arXiv:2203.06649*, 2022.

Rachel Brown, Vasha DuTell, Bruce Walter, Ruth Rosenholtz, Peter Shirley, Morgan McGuire, and David Luebke. Efficient dataflow modeling of peripheral encoding in the human visual system. *arXiv preprint arXiv:2107.11505*, 2021.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9650–9660, 2021.

Chun-Fu Richard Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 357–366, 2021.

Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3213–3223, 2016.

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. *Advances in Neural Information Processing Systems*, 33:13073–13087, 2020.

Joel Dapello, Jenelle Feather, Hang Le, Tiago Marques, David Cox, Josh McDermott, James J DiCarlo, and SueYeon Chung. Neural population geometry reveals the role of stochasticity in robust perception. *Advances in Neural Information Processing Systems*, 34:15595–15607, 2021.

Todor Davchev, Timos Korres, Stathi Fotiadis, Nick Antonopoulos, and Subramanian Ramamoorthy. An empirical evaluation of adversarial robustness under transfer learning. *arXiv preprint arXiv:1905.02675*, 2019.

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*, pp. 248–255. Ieee, 2009.

Arturo Deza and Talia Konkle. Emergent properties of foveated perceptual systems. *arXiv preprint arXiv:2006.07991*, 2020.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

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.

Logan Engstrom, Andrew Ilyas, Hadi Salman, Shibani Santurkar, and Dimitris Tsipras. Robustness (python library), 2019a. URL https://github.com/MadryLab/robustness.

Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Brandon Tran, and Aleksander Madry. Learning perceptually-aligned representations via adversarial robustness. In *ArXiv preprint arXiv:1906.00945*, 2019b.

M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010.

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

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

Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3354–3361, 2012. doi: 10.1109/CVPR.2012.6248074.

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

Ross Goroshin, Michael F Mathieu, and Yann LeCun. Learning to linearize under uncertainty. *Advances in neural information processing systems*, 28, 2015.

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.

Anne Harrington and Arturo Deza. Finding biological plausibility for adversarially robust features via metameric tasks. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=yeP_zx9vqNm`.

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*, pp. 770–778, 2016.

Olivier J Hénaff, Robbe LT Goris, and Eero P Simoncelli. Perceptual straightening of natural videos. *Nature neuroscience*, 22(6):984–991, 2019.

Olivier J Hénaff, Yoon Bai, Julie A Charlton, Ian Nauhaus, Eero P Simoncelli, and Robbe LT Goris. Primary visual cortex straightens natural video trajectories. *Nature communications*, 12(1):1–12, 2021.

Yanping Huang and Rajesh PN Rao. Predictive coding. *Wiley Interdisciplinary Reviews: Cognitive Science*, 2(5):580–593, 2011.

Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. *Advances in neural information processing systems*, 32, 2019.

Alexander JE Kell and Josh H McDermott. Deep neural network models of sensory systems: windows onto the role of task constraints. *Current opinion in neurobiology*, 55:121–132, 2019.

Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. The cifar-10 dataset. *online: http://www. cs. toronto. edu/kriz/cifar. html*, 55(5), 2014.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

Jonas Kubilius, Martin Schrimpf, Kohitij Kar, Rishi Rajalingham, Ha Hong, Najib Majaj, Elias Issa, Pouya Bashivan, Jonathan Prescott-Roy, Kailyn Schmidt, et al. Brain-like object recognition with high-performing shallow recurrent anns. *Advances in neural information processing systems*, 32, 2019.

William Lotter, Gabriel Kreiman, and David Cox. Deep predictive coding networks for video prediction and unsupervised learning. *arXiv preprint arXiv:1605.08104*, 2016.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Bruno A Olshausen and David J Field. Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision research*, 37(23):3311–3325, 1997.

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 (eds.), *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc., 2019. URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

Rishi Rajalingham, Kailyn Schmidt, and James J DiCarlo. Comparison of object recognition behavior in human and monkey. *Journal of Neuroscience*, 35(35):12127–12136, 2015.

Shibani Santurkar, Dimitris Tsipras, Brandon Tran, Andrew Ilyas, Logan Engstrom, and Aleksander Madry. Computer vision with a single (robust) classifier. In *ArXiv preprint arXiv:1906.09453*, 2019.

Martin Schrimpf, Jonas Kubilius, Ha Hong, Najib J Majaj, Rishi Rajalingham, Elias B Issa, Kohitij Kar, Pouya Bashivan, Jonathan Prescott-Roy, Franziska Geiger, et al. Brain-score: Which artificial neural network for object recognition is most brain-like? *BioRxiv*, pp. 407007, 2020.

Kalpana Seshadrinathan, Rajiv Soundararajan, Alan C Bovik, and Lawrence K Cormack. A subjective study to evaluate video quality assessment algorithms. In *Human Vision and Electronic Imaging XV*, volume 7527, pp. 128–137. SPIE, 2010a.

Kalpana Seshadrinathan, Rajiv Soundararajan, Alan Conrad Bovik, and Lawrence K Cormack. Study of subjective and objective quality assessment of video. *IEEE transactions on Image Processing*, 19(6):1427–1441, 2010b.

Rulin Shao, Zhouxing Shi, Jinfeng Yi, Pin-Yu Chen, and Cho-Jui Hsieh. On the adversarial robustness of vision transformers. *arXiv preprint arXiv:2103.15670*, 2021.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Wei Wen, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Learning structured sparsity in deep neural networks. *Advances in neural information processing systems*, 29, 2016.

Ross Wightman. Pytorch image models. https://github.com/rwightman/pytorch-image-models, 2019.

Xiaogang Xu, Hengshuang Zhao, and Jiaya Jia. Dynamic divide-and-conquer adversarial training for robust semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7486–7495, 2021.

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

A APPENDIX

A.1 CURVATURE VS COSINE SIMILARITY

$$\text{cosine similarity (vectors)} := \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} \tag{5}$$

$$\text{cosine similarity (video frames)} := \cos(\theta) = \frac{x_t \cdot x_{t+1}}{\|x_t\|\|x_{t+1}\|} \tag{6}$$

$$\text{curvature} := c_t = \arccos(\hat{v}_t \cdot \hat{v}_{t+1}) \tag{7}$$

$$\text{cosine curvature} := \hat{v}_t \cdot \hat{v}_{t+1} = \frac{v_t \cdot v_{t+1}}{\|v_t\|\|v_{t+1}\|} = \cos(c_t) \tag{8}$$

A.2 MODELS

A.2.1 MODEL SOURCES

All deep neural networks we analyzed were pretrained. Exact links to code and weights can be found in the supplemental material in the Network Comparison Spreadsheet. The standard ImageNet-trained ResNet (He et al., 2016) model was downloaded from PyTorch's model zoo (Paszke et al., 2019). Adversarially robust ResNet models were all downloaded from (Engstrom et al., 2019a). The adversarially robust ResNets were trained using projected gradient descent. All ViT (Dosovitskiy et al., 2020) and standard trained CrossViT daggar (Chen et al., 2021) models were downloaded from the timm library (Wightman, 2019). All CrossViT daggar adversarially robust and rotationally invariant models were downloaded from the repository of Berrios & Deza (2022). The adversarially robust CrossViTs were trained with fast gradient sign method as stated in Berrios & Deza (2022). DINO models were downloaded from the DINO repository (Caron et al., 2021), while PredNet models were downloaded from Lotter et al. (2016).

A.2.2 MODEL ANALYSIS PROCEDURE

We showed each model the same 12 natural videos that were used in the psychophysics experiments of (Hénaff et al., 2019). The videos were taken from the Chicago Motion Database (at University of Chicago, 2022), the film 'Dogville', Lions Gate Entertainment (2003), and LIVE Video Quality Database (Seshadrinathan et al., 2010b;a). The videos were grayscale, consisting of 11 frames each of 512×512 pixels, capturing natural motion such as rippling ocean water or a person walking through a crowded street. We resized the video frames to be 224×224 for all deep networks and 256×256 for bio-models that use steerable pyramids. One limitation of this work is that we did not evaluate models on larger video datasets, but we wanted to use psychophysically validated stimuli for our analyses. For each model, we recorded its activations at intermediate and final layers for each video. We then found the global curvature for each stage of the model using equation 4 where we used the flattened model activations as the input x_t to the curvature procedure.

We compared the global curvature at each layer of the model to the curvature of the video in pixel space. Models that straighten are defined as models that have a lower global curvature at deeper layers. When comparing the curvature of different model layers, we chose not to reduce the dimensionality of each layer activation to be the same across stages. Although principle components analysis (PCA) was sometimes used in Hénaff et al. (2019) when expressing curvature, they did not use it in their analysis of deep networks. Furthermore, while an architecture's inherent dimensionality is likely relevant to a representation's curvature, we preferred not to introduce any additional transformations that would influence the measured curvature. We also found that performing PCA did not greatly affect trends like adversarial training leading to more straight representations See Figure 10 for examples of curvature for different numbers of principle components on two ResNet50 models.

A.2.3 COMPUTE

Our methods do not require large compute. All individual model analyses can be run on CPU. We used a single GPU to speed up getting the features activations at each layer to the order of minutes per model.

A.2.4 EFFECT OF PCA ON MODEL CURVATURE

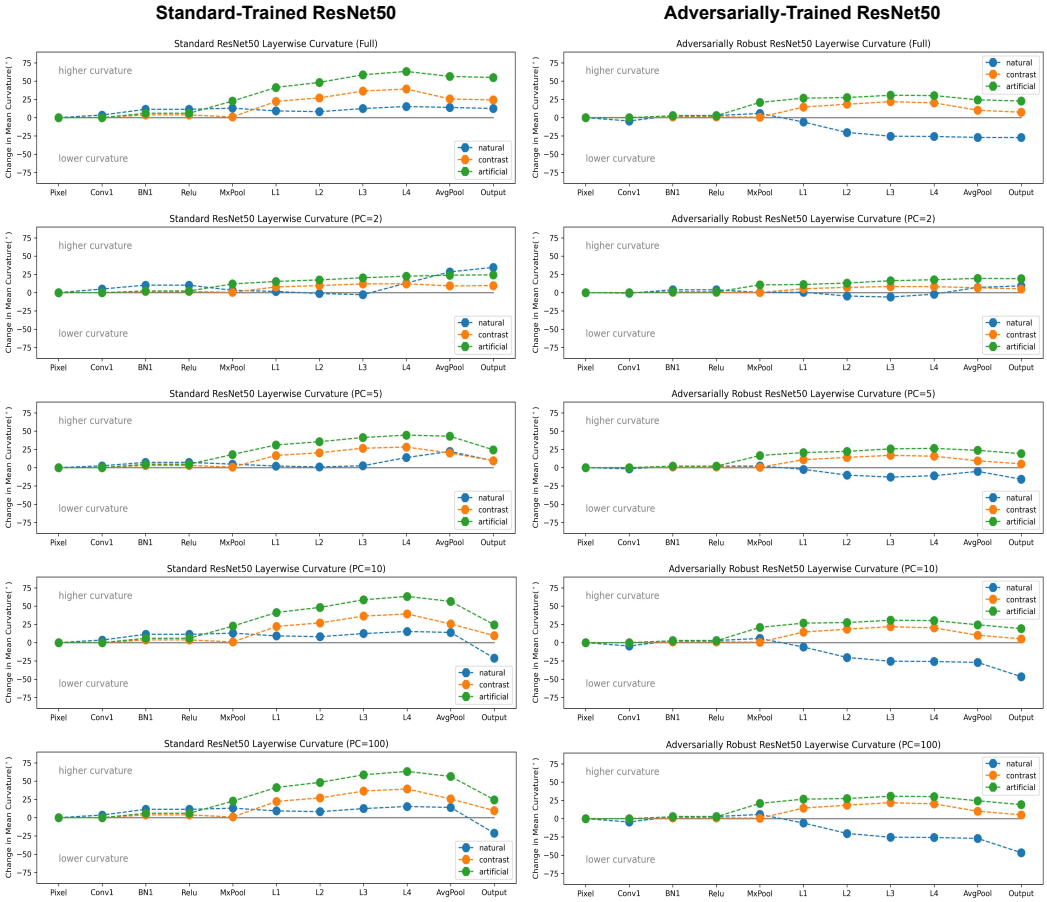

Figure 10: **Change in curvature for different numbers of principle components**. Each subplot shows layer-wise change in curvature. **Left** is a standard ResNet50 trained on ImageNet (Deng et al., 2009) classification. The natural line is the curvature measures on the natural videos from (Hénaff et al., 2019) that we used for all model analyses in the paper. The contrast and artificial sequences were additional video sequences psychophysically tested by (Hénaff et al., 2019) where humans tend to have little change in curvature for contrast but large increase in curvature for artificial. **Right** is an adversarially robust (projected gradient descent with l_2 norm and $\epsilon = 3$) ResNet50 trained on ImageNet. All rows below the top row show curvature when taking the first 2, 5, 10, and 100 principle components (PCs) of each representation layer activations. For PCs less than the number of frames in each video (11), we performed PCA on the pixel-level video in addition to the model activations when computing the change in curvature. For all PCs, the standard ResNet50 has a higher curvature for natural video sequences than the adversarially robust.

A.3 FINAL STAGE CURVATURES

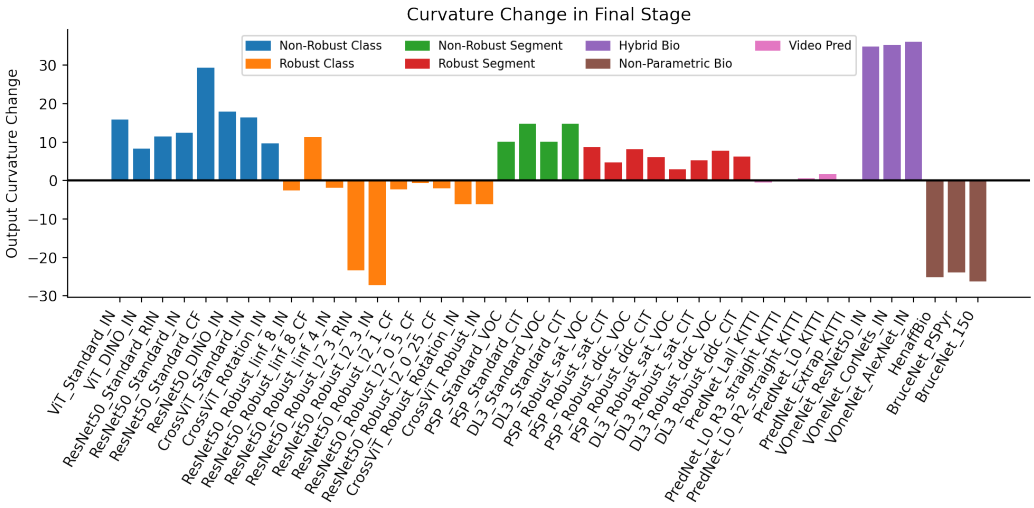

Figure 11: **Final stage curvature for a variety of network architecture and training schemes.** Model families show very similar curvature to that of their output representation layers 3.

The final stage of a network for image classification models is the last fully connected layer. We see similar trends in curvature across model families as in Fig 3. We chose to report the the curvature on the layer just before classification because this is the layer that would be most commonly used as the representational backbone for other tasks.

A.4 ADVERSARIAL ACCURACY AND CURVATURE

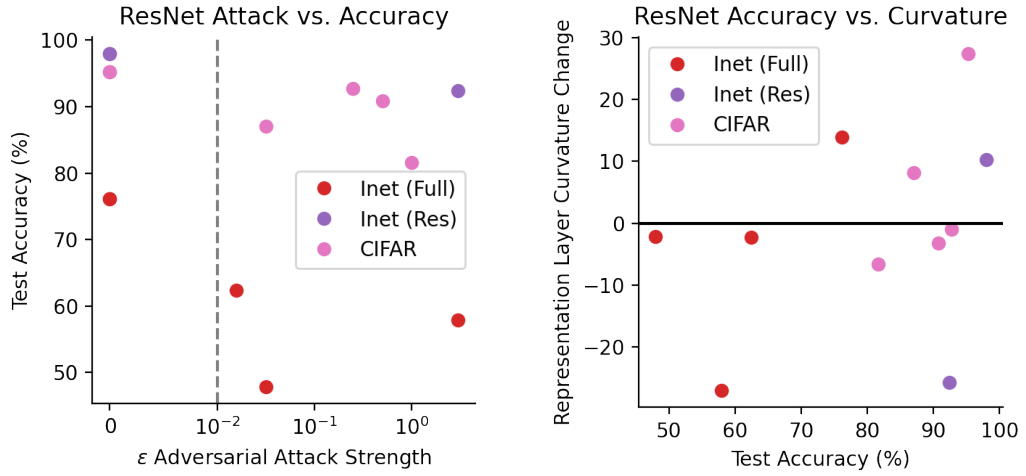

Figure 12: **Left**: While adversarial attacks with greater ϵ strength impart many desirable robustness properties on a network, adversarial training does not improve test accuracy, often decreasing test accuracy on the within-domain test set for a given model. (Data plotted on symlog scale.) **Right**: While stronger adversarial attacks decrease curvature, improved test accuracy for a model is not predictive of output curvature reduction. Rather, within a given model training/test set, increased test accuracy predicts a smaller curvature reduction in the output layer.

A.5 PREDNET PREDICTIONS

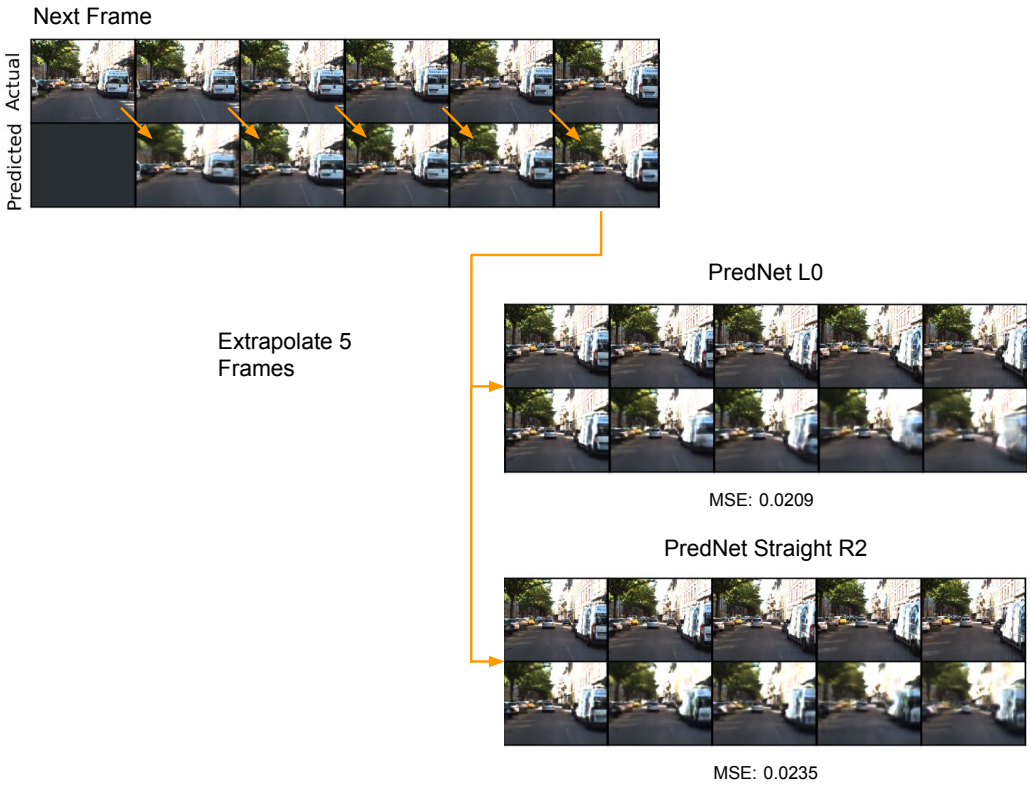

Figure 13: **Example Predictions for Original and Straighten PredNet**. Both models were trained on the KITTI dataset (Geiger et al., 2012). The straightened PredNet was trained with penalty on its R2 layer if there was high curvature.

A.6 NEGATIVE SOCIETAL IMPACTS

We believe there are few negative societal impacts of this paper. Our work was exploratory and did not introduce any new models. However, we note that the development of machine vision systems, which can operate at or above the level of performance of humans at certain tasks, may lead to negative societal impacts such as the loss of jobs and industries as human workers are replaced.