HOW DATA AUGMENTATION SHAPES NEURAL REPRESENTATIONS

Tianxiao He
Department of Computer Science
New York University
New York, NY 10012 USA
th3129@nyu.edu

Alex H. Williams
Center for Neural Science
New York University
New York, NY 10012 USA
aw4614@nyu.edu

Sarah E. Harvey
Flatiron Institute
New York, NY 10010, USA
sharvey@flatironinstitute.org

ABSTRACT

Data augmentation is widely recognized for improving generalization in deep networks, yet its impact on the geometry of learned representations remains poorly understood. In this work, we characterize how different data augmentation strategies reshape internal representations in neural networks. Using tools from shape analysis, we embed network hidden representations into a metric space where distance is invariant to scaling, translation, rotation and reflection. We show that increasing augmentation strength leads to reproducible trajectories in this space, and that different augmentation types steer representations in distinct directions. Moreover, we investigate how neural representation shapes are distorted along data augmentation trajectories, and show that insights from neural geometry can predict which representations provide the most improvement when ensembling models. Our results reveal shared geometric patterns across architectures and seeds, and suggest that analyzing shape-space trajectories offers a principled tool for understanding and comparing data augmentation methods.

1 INTRODUCTION

The highly complex and overparametrized nature of deep neural networks can result in many combinations of hyperparameters that achieve similar task performance. While hyperparameters such as learning rate, batch size, and momentum are known to be crucial for training stability and generalization (Smith, 2018), and recent work shows that training dynamics influence representational geometry (Alleman et al., 2024), there is little direct study on how hyperparameter choices change the geometry of learned neural representations. In particular, when a single hyperparameter is isolated, how are the learned hidden representations steered through neural representation space across models, as a function of the hyperparameter?

A particularly rich area to study this phenomenon is data augmentation. It is a foundational technique for improving the performance and generalization capabilities of deep networks by diversifying existing datasets with simple transformations (Shorten & Khoshgoftaar, 2019), such as rotation, cropping, color jitter and addition of gaussian noise. Data augmentation is often interpreted as an implicit regularization technique (Dao et al., 2019), leading to more robust internal representations (Rebuffi et al., 2021), or the incorporation of prior knowledge about invariances in the task (Lenc & Vedaldi, 2015; Chen et al., 2020). Despite their centrality in modern training pipelines, data augmentations are often tuned using simple heuristics or searches that optimize task performance. However, many distinct combinations of augmentation hyperparameters, such as strength, probability, or spatial extent, can yield near-identical accuracy on held-out data. This degeneracy suggests that performance metrics alone may obscure meaningful differences in how models learn to represent the world.

While the benefits of data augmentation intuitively stem from enriching the training data, a deeper understanding remains elusive. For example, do distinct augmentation methods (e.g. image rotations and random cropping) sculpt hidden layer computations in fundamentally different ways, or are their effects are redundant? How do two augmentation methods interact when they are simultaneously applied? More broadly, does diversity in representational geometry predict the extent of functional improvements when models are ensembled across different hyperparameter values? Can two augmentations synergistically interact or interfere with each other, and can this be observed from their hidden representational geometry?

Studying how data augmentation shapes the geometry of neural network representations may also offer a path toward more interpretable comparisons between artificial and human cognition. Prior work suggests that training neural networks on transformed images as in data augmentation may help replicate human performance and representations (Yoshihara et al., 2023). However, determining if and when such modifications move network representations toward or away from human representations first requires a more sophisticated understanding of our representational similarity metrics as a function of training parameters. In particular, it may be important to understand representational similarity not just as a scalar value, but in terms of trajectories and angles in representational space.

Here, we leverage *metric spaces* on the geometry of hidden layer representations (Williams et al., 2021; Lange et al., 2023) as a first step towards addressing these questions. This formulation embeds the hidden layer representations in a smooth manifold, and allows us to apply techniques from Riemannian geometry to study the effect of hyperparameter changes on the eventual learned representation. Any changes in hyperparameters or augmentation strength, type, or composition are then conceptualized as operations on the final learned representations that move it within this manifold, and are endowed with a distance and direction. This enables a systematic comparison of how augmentation strength and types reshape the representation geometry across network layers, initializations, and architectures. Our main contributions are:

- We propose a geometric framework for analyzing the effects of data augmentation hyperparameters on learned neural representations
- We demonstrate that increasing data augmentation magnitude steers representations along structured trajectories in shape space that are reproducible across network layers, random initializations, and architectures
- We show that greater divergence between trajectories of different augmentations predicts larger gains from model ensembling
- We find that data augmentation displaces shape landmarks non-uniformly, with effects that depend on network depth

2 Related Work

Data augmentation Data augmentation is a central technique in modern machine learning pipelines, used to increase data diversity, improve generalization, and enhance robustness (Shorten & Khoshgoftaar, 2019; Wang et al., 2025; Bishop, 1995; Dao et al., 2019). It is generally thought that augmentations shape neural computation by "collapsing" the manifold of representations along nuisance dimensions, so that augmented views of the same input are mapped to similar responses (Lenc & Vedaldi, 2015). The broader principle of view-invariant representations can be traced back decades within the computer vision and visual neuroscience literature (Olshausen et al., 1993; Riesenhuber & Poggio, 1999; DiCarlo & Cox, 2007).

Importantly, the intuition that training with augmentations "collapses" augmented views onto a single point only pertains to the representational geometry of the augmented dataset—it does *not* tell us how the manifold of representations over clean images is altered by training with augmentations. This distinction is particularly important for augmentation techniques that produce out-of-distribution training samples (e.g. adding a large amount of Gaussian noise), with the goal of improving performance of unseen within-distribution samples. In this paper, we investigate how different augmentations alter representations of non-augmented "clean images" in the test set. Furthermore, we study whether these changes in representational geometry are consistent across different random weight initializations as well as across different network architectures. To our knowledge, few (if any) papers touch on these issues.

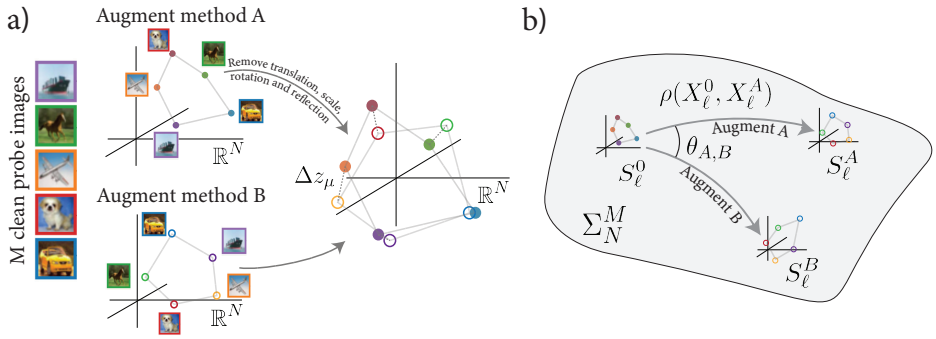

Figure 1: **Embedding representations into shape space enables more meaningful comparison.** **(a)** After two models are trained end-to-end using two different data augmentation methodologies, M unaltered images from the test dataset are randomly selected as probe images. The Riemannian shape distance Equation (3) between these two representations is measured as the angular distance after translation, scale, rotation, and reflection variability are discarded. **(b)** The representation shapes with the Riemannian shape distance form a metric space in which we may measure distance, compute geodesics, and measure angles.

Representational similarity Comparing neural network representations is essential for understanding training dynamics, transferability, and model equivalence. Representational similarity metrics aim to characterize relationships between hidden layer activation patterns across layers and models. Metrics like centered kernel alignment (CKA) measure pairwise similarity between activation matrices using kernel-based alignment (Kornblith et al., 2019), and are widely used to study how layer representations change across seeds or architectures (Nguyen et al., 2020). More recent work has proposed using formal metric spaces to capture more nuanced geometric structure in representations (Williams et al., 2021) which allows for more advanced analysis methods on representations such as regression and clustering. (Lange et al., 2023) leverage this idea to study the trajectories of representations through layers of deep neural networks, and also introduce concepts from Riemannian geometry to study these paths. Our work is philosophically complementary but orthogonal to theirs, as we study the trajectories of neural representations *across models* as hyperparameters are tuned.

3 Methods

In this section, we define our notion of a metric space for neural representations, and define the distance between representations. We then discuss geodesics of this metric and how to measure angles between trajectories of representations. Lastly, we discuss inspecting the landmark displacements as a window into how shapes are distorted along trajectories in shape space.

3.1 Problem setup

Let $f_L^A : \mathcal{Z} \mapsto \mathcal{Y}$ represent the function of a feedforward neural network with $L \in \mathbb{N}$ layers, where $\mathcal{Z}$ is the input space, $\mathcal{Y}$ is the output space, and the network is trained using data augmentation strategy A. The neural network $f_L^B : \mathcal{Z} \mapsto \mathcal{Y}$ is identical in architecture and initialization to f_L^A, but is trained using data augmentation strategy B. Functional changes induced by this change in training protocol are easily measured by computing the change in network generalization performance.

However, here we are interested in studying the changes to hidden representations induced by a change in data augmentation protocol. We define the *representation mapping* from the input domain to a vector of activations at a hidden layer ℓ with N units as $f_\ell^A : \mathcal{Z} \mapsto \mathbb{R}^N$ and similarly $f_\ell^B : \mathcal{Z} \mapsto \mathbb{R}^N$. How similar are the activations produced in hidden layers by $f_\ell^A(\cdot)$ and $f_\ell^B(\cdot)$? More specifically, to what extent has the change in training data induced by the data augmentation parameter change $A \to B$ changed the representations formed by $f_\ell^A(\cdot)$ and $f_\ell^B(\cdot)$, over a representative collection of probe inputs $z_1, \ldots, z_M \in \mathcal{Z}$? We proceed by measuring neural responses $f_\ell^A(z_1) \ldots f_\ell^A(z_M)$ and stacking them row-wise into a representation matrix $X_\ell^A \in \mathbb{R}^{M \times N}$. Like-

wise, we form a matrix $X_\ell^B \in \mathbb{R}^{M\times N}$ from the second network's responses, $f_\ell^B(z_1)\dots f_\ell^B(z_M)$. Intuitively, one can view these matrices as approximations to each network's input to layer ℓ mapping over a discrete set of M points. The representation matrices X_ℓ^A can be visualized as M points in an N-dimensional space (Fig. 1 a).

Measuring the distance between representations X_ℓ^A and X_ℓ^B is complicated by the fact that even though the architectures of the two systems are identical, once the networks are trained with different augmentation protocol there is no inherent correspondence between activations at layer ℓ. We will compare representations using Riemannian shape distance, which coupled with the set of R representations $\{X_i \in \mathbb{R}^{M\times N}, i = 1, ..., R\}$ forms a metric space with desirable geometric properties, and is related to kernel-based measures of representational similarity (Harvey et al., 2024). In particular, embedding representations in this metric space will allow us to measure the distances as well as directions of non-trivial transformations to the representations as distances, and define a notion of shortest paths and angles between representation trajectories.

3.2 Riemannian shape distance

We can intuitively define the notion of *shape* as the geometrical information remaining when scale, location, and global rotations are treated as nuisance transformations and discarded from an object (Kendall, 1977). This notion can be made mathematically precise by sampling a finite set of locations from the object that serve as a points of correspondence between shapes, which are often called *landmarks* Fig. 1 (a). We will study the shape of the activation patterns in deep neural networks, with the landmarks corresponding to M probe inputs (images), in N dimensions. Measuring similarity or distance between neural representations up to these nuisance transformations could be desirable for artificial neural networks, as this posits that the units themselves are not meaningful (Elhage et al., 2022). Indeed, different models may represent the same information in rotated bases and remain functionally equivalent. Furthermore, orthogonal transformations of the activations do not effect the information that is linearly decodable from the representations, which is sometimes treated as a probe of layer function (Alain & Bengio, 2017).

Translation is removed from the representation matrices by multiplying by the centering matrix, $C = I_M - \frac{1}{M}1_M 1_M^T$ and scale is removed by dividing by the Frobenius norm, to yield the pre-shape associated with the ith representation, Z_i:

$$Z_i = \frac{CX_i}{\|CX_i\|_F} \tag{1}$$

As $\|Z_i\|_F = 1$, the pre-shapes lie on a unit-radius hypersphere in $\mathbb{R}^{(M-1)N}$. The *shape* of the representation S_i is defined by the set of all rotated versions of the pre-shape– it is an equivalence class under the action of the nuisance transformations:

$$S_i = \{Z_i O : O \in \mathcal{O}(N)\} \tag{2}$$

where $O(N)$ is the orthogonal group of rotations and reflections in N-dimensions, which is the set of $N \times N$ matrices satisfying $O^T O = OO^T = I_N$ and $\det O = \pm 1$. All pre-shapes connected by orthogonal transformations in $\mathcal{O}(N)$ form a *fibre* on the pre-shape hypersphere, which corresponds with the shape. *Shape space* Σ_N^M is the space of all shapes with M landmarks in $\mathbb{R}^N$. We will be largely concerned with the shapes associated with representations at particular layers ℓ as a result of data augmentation method A, S_ℓ^A (Fig. 1 b). Each shape may be visualized by a member of its shape set (e.g. the original activations measured from the deep network), but we need to specify a distance metric between the shapes themselves to define the non-Euclidean shape metric space.

The *Riemannian shape distance* is a canonical way to measure distance between shapes in shape space. It is the smallest great circle distance between two pre-shapes, minimized over orthogonal transformations of the pre-shapes:

$$\rho(X_i, X_j) = \arccos\Big(\sup_{O\in\mathcal{O}(N)} \mathrm{Tr}[Z_j^\top Z_i O]\Big) \tag{3}$$

The Riemannian shape distance is an intrinsic geodesic distance through shape space. It is measured in radians, and is the smallest angle between pre-shapes Z_i and Z_j on the hypersphere that can be achieved by N-dimensional rotations of the pre-shapes. The solution to the optimization problem in

Equation (3) is available in closed form as

$$\rho(X_i, X_i) = \arccos\left(\sum_{k=1}^{N} \sigma_k\right) \tag{4}$$

where $Z_j^\top Z_i = U\Sigma V^\top$ is the singular value decomposition of $Z_j^\top Z_i$ and σ_k are the diagonal elements of Σ.

In Section 4 we explore the effect of data augmentation methods on the geometry of neural representations by way of their trajectories in shape space, which is enabled by measuring Riemannian shape distances between representations. This metric space approach also allows the use of methods from Riemannian geometry to study how different hyperparameter changes effect representation shape, some of which we detail below.

Shape geodesics. Given shapes S_0 and S_1 in shape space, let corresponding preshape Z_1^* denote the optimally aligned version of Z_j under an orthogonal transformation $Q \in O(m)$ that maximizes inner product with Z_1. The geodesic distance is the Riemannian shape distance in Equation (3), and the Riemannian shape geodesic is obtained by projecting the great circle on the preshape sphere:

$$\gamma(t) = \frac{\sin((1-t)\rho)}{\sin\rho} Z_0 + \frac{\sin(t\rho)}{\sin\rho} Z_1^*, \quad t \in [0, 1].$$

This geodesic traces the shortest smooth deformation from one shape to another, ignoring overall orientation, scale and reflection.

Geodesic angles. Let γ_1 and γ_2 be two shape geodesics between the shape S_0 and the shapes S_1 and S_2, respectively. Their optimally aligned preshapes to initial preshape Z_0 are Z_1^* and Z_2^*, and the initial tangent vectors are

$$V_i = \frac{\rho_i}{\sin\rho_i}\left(Z_i^* - \cos(\rho_i) Z_0\right), \text{ with } \rho_i = \arccos(\operatorname{Tr} Z_0^\top Z_i^*).$$

for $i = 1, 2$. These tangent vectors represent the initial direction of each geodesic path in the shape manifold.

The angle between the two shape geodesics is the angle between their tangent vectors at the initial shape S_0:

$$\theta_{1,2} = \angle(\gamma_1, \gamma_2) = \arccos\left(\frac{\langle V_1, V_2\rangle}{\|V_1\|_F \cdot \|V_2\|_F}\right). \tag{5}$$

This angle quantifies how differently the two shapes begin to deform away from the same reference shape.

Landmark displacement. We can also investigate *how* shapes are being deformed as data augmentation is applied at varying levels by measuring the landmark-level displacement between two shapes. The displacement for each landmark $\Delta z_\mu \in \mathbb{R}^N$, $\mu \in \{1, ..., M\}$, is computed as the Euclidean displacement vector between corresponding landmarks across two optimally aligned preshapes (Fig. 1 a). These displacement vectors are the rows of the matrix $\Delta Z = Z_\ell^A - Z_\ell^B O^*$, with $O^* = \arg\sup_{O \in \mathcal{O}(N)} \operatorname{Tr}[(Z_\ell^A)^\top Z_\ell^B O]$. We may then study which landmarks are maximally or minimally distorted along a trajectory in shape space, and which landmarks are being distorted in similar directions.

4 Experiments

Here we observe and quantify the effect of data augmentation processes during learning on the distribution of representational shapes in shape metric space after learning. To study image representations, we train standard ResNet18 (He et al., 2016) on the CIFAR-10 dataset for image classification, while controlling the data augmentation methods applied during training. We build on a standard ResNet training recipe (cropping, flipping, and normalization), and apply stochastic transformations to all training images, including geometric and photometric transformations, noise, and occlusion using standard PyTorch data augmentation methods. The same type of augmentation is applied across runs with varying magnitudes, such as rotation degree, cutout location, or noise amplitude.

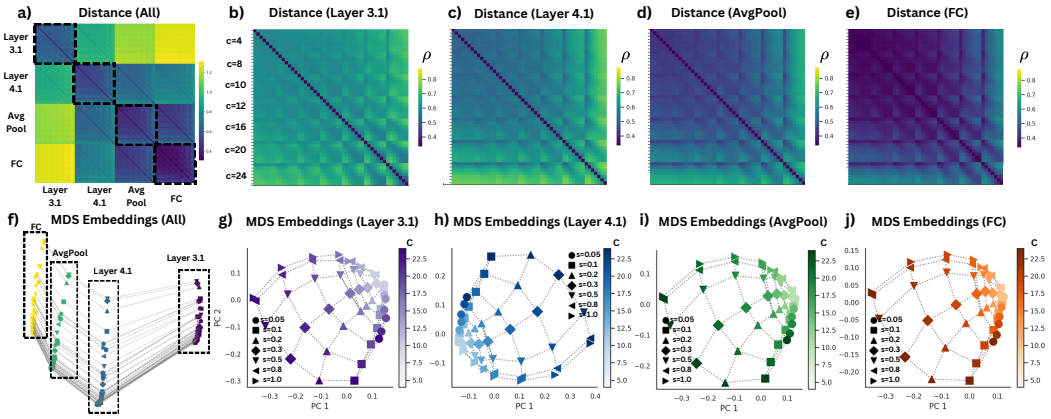

Figure 2: **Data augmentation produces stereotyped layer trajectories in shape space across models.** We analyze the effect of patch gaussian augmentation on representations learned by ResNet18 trained on Cifar10. (a) Riemannian shape distance matrix between trained representations from different layers (blocks) and different data augmentation parameters (within-layer variation). (b-e) Diagonal blocks of panel A in isolation, showing repeated structure to layer geometry variation over different patch Gaussian hyperparameters cutout size c, and noise variance s. (f) Projection onto the first two principal components of a 200-dimensional Euclidean embedding of the ensemble of representations with the distance matrix from panel A, explaining 60% of the variance. (g-j) Similar to panel f, the projections onto the first two principal components of individual Euclidean embeddings of within-layer distance matrices in b-e.

The internal representations were collected from a sampling of hidden layers after training, in response to $M = 10,000$ clean (unaugmented) probe images, constituting the entire CIFAR-10 test set. Activations produced by convolutional layers of size $M \times C \times H \times W$, where C is the number of channels and $H \times W$ are the spatial dimensions, were flattened into representation matrices $X_i \in \mathbb{R}^{M \times CHW}$. Full training details, such as hyperparameters and transformation parameters, are listed in appendix A. Representations formed by hidden layers are often extremely large in terms of the number of activations $N = CHW$, and the decomposition required to compute Equation (4) can become computationally expensive. If $N > 1000$, dimensionality of the representations was reduced by discarding all but the top 1000 principal components, with the observation that in all representations studied this projection preserved at least 75% of the variance.

Data augmentation steers representations along stereotyped trajectories in shape space. We first investigated the effects of the patch Gaussian augmentation proposed by (Lopes et al., 2019). A useful feature of this augmentation method is that it is defined by two hyperparameters that can be independently and gradually varied: the patch size c, which specifies the number of pixels in a square image patch corrupted by Gaussian noise of variance, s. When c equals the width and height of the input images the method reduces to additive Gaussian noise, while for large s, it resembles "Cutout" with patch size c (DeVries & Taylor, 2017). Patch Gaussian augmentations therefore define a convenient two-dimensional space of image augmentations that interpolate between well-established methods used in practice.

We analyzed representations in four layers of ResNet18 across 49 independent end-to-end training runs initialized with different random seeds. We varied c and s over a grid of values with $c \in \{4, 8, 10, 12, 16, 20, 24\}$ and $s \in \{0.05, 0.1, 0.2, 0.3, 0.5, 0.8, 1.0\}$. For this ensemble of representations, we may construct a distance matrix D in which element D_{ij} is equal to the Riemannian shape distance measured between representations i and j (Fig. 2 a). The diagonal blocks of this matrix correspond with the distances among representations from the same layer, wherein representation i was trained with patch gaussian parameters $\{c, s\}$, while representation j was trained independently with patch gaussian parameters $\{c', s'\}$ (Fig. 2 b). An overall trend may be observed that earlier hidden layers are subject to larger changes in representation shape as a result of data augmentation, however, we also observe distinct sub-block structure which correlates with changes in the augmentation parameters. We then use multidimensional scaling (MDS) to embed each representation in a 200-dimensional Euclidean space, and use principal components analysis (PCA) to

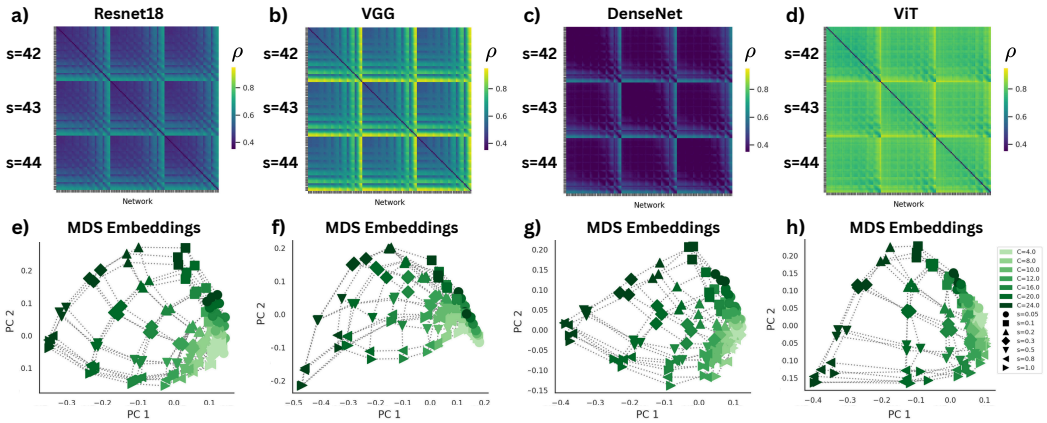

Figure 3: **Effects of data augmentation hyperparameters on geometry are consistent across architectures and initializations.** (a) Riemannian shape distance matrix for representations at the avgpool layer of ResNet18 trained under different values of patch Gaussian augmentation parameters (cutout size c, noise variance s), and three different random seeds. The repeating block structure indicates that representation geometry is determined by augmentation parameters rather than seed. (b–d) Analogous shape distance matrices and MDS-PCA projections for the penultimate layer of (b) VGG, (c) DenseNet, (d) and ViT. (e-h) MDS and PCA embeddings of the shape distance matrix for (e) Resnet18, (f) VGG, (g) DenseNet, (h) ViT. Despite differences in architecture, the structured distribution of representation shapes in shape space is largely preserved, revealing augmentation-induced geometric signatures that are robsust across models and seeds.

display the main axes of variation among the representation shapes, a visualization method we will refer to as MDS-PCA (Williams et al., 2021). The top two PCs for the embedding of all layers of all models is shown in Fig. 2 c, and the global clustering of layers in one dimension is visible. When restricted to the within-layer distance matrices, consistent relational structure can be observed among the layers corresponding to different hyperparameter tuples $\{c, s\}$, suggesting that changes to the data augmentation smoothly steers the representations in shape space in ways that are visible once the effect has been isolated. Figures A-1 and A-2 replicate this experiment using augmentation pairs Cutout with random crop, and rotation with sheer, and different geometric patterns can be observed for different augmentation methods. Figure A-3 replicates this experiment with WideResNet50 fine tuned using the Imagenet dataset.

The effect of data augmentation on representation shape is consistent over random seeds and observed across architectures. In complement with previously observed "butterfly effects" in which small perturbations early in training lead to training trajectories that drive models into distinct loss minima (Altıntaş et al., 2025), we observe that training with different augmentation hyperparameters converge to geometrically distinct learned representations, however, the distribution of these representations in shape space is conserved across random initializations and structured according to parameter values. Fig. 3 a demonstrates this effect for applying the patch Gaussian data augmentation method during end-to-end training of ResNet18, the same model as shown in Fig. 2, at the avgpool layer. The colormap in Fig. 3 a represents the Riemannian shape distance matrix between different values of patch Gaussian parameters $\{c, s\}$ and between different random seeds. A diagonal block of this matrix replicates the experiment of Fig. 2 at the avgpool layer, while the off-diagonal blocks measure the distance between representations formed as a result of training from a different random seed, for the same collection of $\{c, s\}$ parameters. Fig. 3 b-d repeat this analysis at the penultimate layer across several vision architectures. The repeating block structure observed in every model architecture indicates that the distinct patterning of representations in shape space as a function of these hyperparameters is preserved across seeds. We can visualize this phenomenon using the same MDS-PCA technique as Fig. 2, which are shown in the bottom row of Fig. 3.

Different data augmentation methods move representations in different directions in shape space We now study the effect of single data augmentation methods applied individually on representation shape geometry. The embedding of representations into shape space established in Sec-

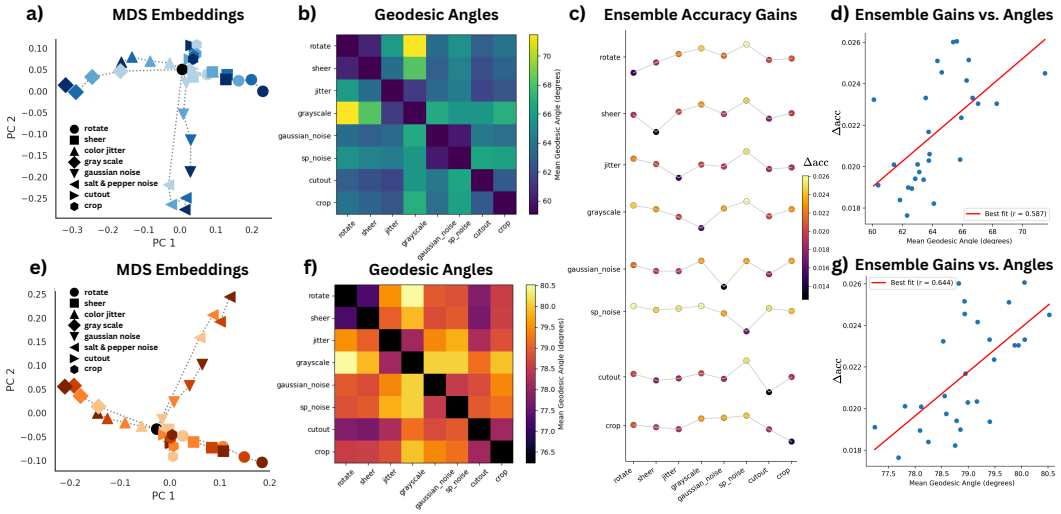

Figure 4: **Different augmentation types produce distinct representational changes, whose diversity predicts ensemble gains.** (a) MDS-PCA embedding of representation shapes for ResNet18 layer 2.1 trained with 8 distinct augmentation types applied individually. Each trajectory connected with dotted lines corresponds to a single augmentation method applied at different magnitudes listed in Table 2. Increasing augmentation strength moves the model representations away from the representation shape measured after training with no data augmentation (black circle). (b) Heatmap of pairwise geodesic angles between the augmentation trajectories, measured with the un-augmented model as the vertex as in Fig. 1 (c) Improvement in test accuracy of ensembled models over average accuracy of model ensemble. Models trained separately with different augmentation methods, then ensembled pairwise across methods. (d) Scatter plot showing the correlation between geodesic angle and the improvement in ensemble accuracy across pairs of augmentations, suggesting that geometric diversity is related with larger ensemble gains. (e,f,g) Analogous MDS-PCA embedding and geodesic angle measurements for AvgPool layer corresponding to panels (a,b,d).

tion 3 allows us to study geometric properties of the representation *trajectories* through shape space, as well as distances moved. Fig. 4 (a,e) show MDS-PCA visualizations of these trajectories for eight different classical data augmentation methods in ResNet18, applied at four different magnitudes (dotted lines connect points of adjacent augmentation hyperparameter magnitude listed in appendix Table 2). (a) shows trajectories of representation shapes collected from layer 2.1, while (e) repeats the same analysis on the penultimate avgpool layer. All trajectories begin at the black point representing the unaugmented learned representation. The mean geodesic angle for two trajectories is computed as the angle in the tangent space of the unaugmented shape formed by geodesics between the unaugmented shape and steps along augmentation trajectories (Equation (5)), averaged over all steps in the trajectories. Fig. 4 (b) and (f) show heatmaps of the mean geodesic angles measured between all augmentation trajectories for layer 2.1 (b) and avgpool (f). Clustering by augmentation type (geometric, photometric, noise, occlusion) is observed in both the MDS-PCA trajectory visualizations and the geodesic angles. The dimensionality of shape space is known (Dryden & Mardia, 2016) to be $N(M-1)-1-N(N-1)/2 \approx 10^6$ for our deep neural network representations, even after dimensionality reduction was performed on the activations. In this high dimensional space, two randomly chosen directions in the tangent space of a particular shape will have an angle very close to 90° and observations by chance of significantly acute angles are overwhelmingly rare (Vershynin, 2018), suggesting that the measured angles between $\sim 60^\circ$ and $\sim 81^\circ$ could indicate non-trivial similarity and difference between trajectory types. As a practical test of this idea, we measure the test accuracy of ensembles of pairs of data augmentation methods. Each ensemble consists of two sets of trained models, $\mathcal{M}_A, \mathcal{M}_B$, from each data augmentation trajectory in Fig. 4 a . Models are combined using soft voting by averaging their predicted class probabilities: $p_{\text{ens}}(y \mid x; A, B) = \frac{1}{|\mathcal{M}_A|+|\mathcal{M}_B|} \sum_{f \in \mathcal{M}_A \cup \mathcal{M}_B} p_f(y \mid x)$, where $p_f(y \mid x) = \text{softmax}(z_f(x))_y$ for model f. Then the benefit of ensembling is quantified using the ensemble accuracy gain relative to the average constituent accuracy $\Delta\text{Acc}(A, B) = \text{Acc}_{\text{ens}}(A, B) - \frac{1}{|\mathcal{M}_A|+|\mathcal{M}_B|} \sum_{f \in \mathcal{M}_A \cup \mathcal{M}_B} \text{Acc}(f)$. We find that

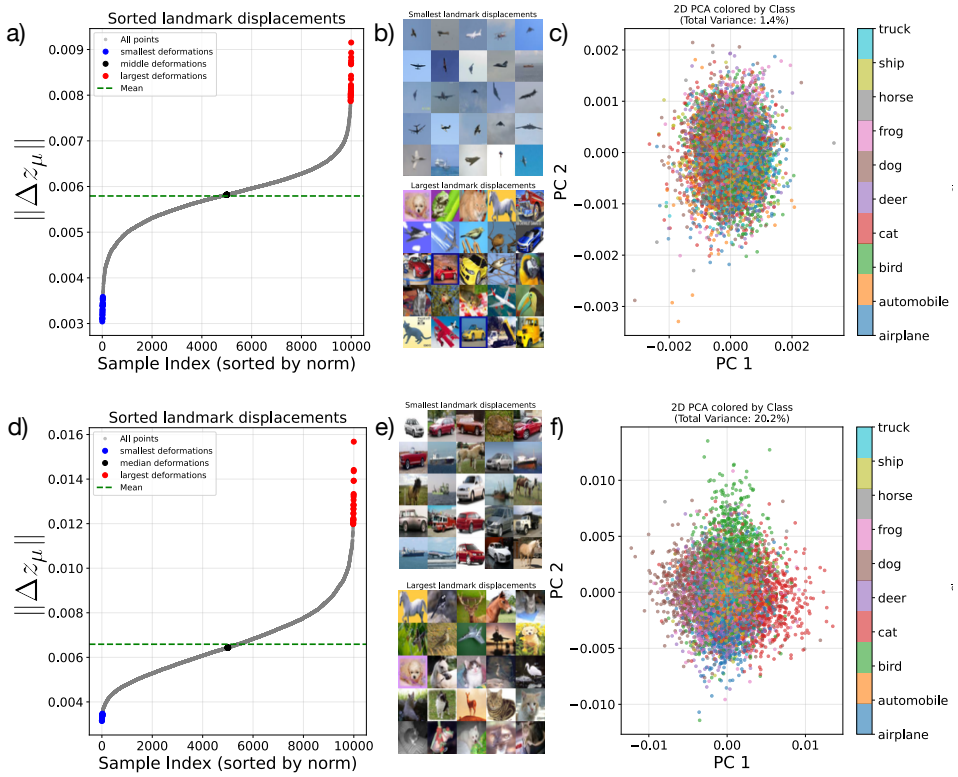

Figure 5: **Data augmentation produces layer-dependent landmark distortions.** Top row contains Resnet18 representations under cutout data augmentation measured after layer 2.1, while bottom row corresponds with avgpool representations. (a, d) Sorted displacement magnitudes across $10,000$ probe images. A small subset of landmarks exhibit the largest (red) and smallest (blue) movement in shape space. (b, e) Probe images from the CIFAR-10 test set corresponding to highlighted landmarks in panels (a) and (d). Image tiles show minimal (top) and maximal (bottom) distortion magnitude images, measured along a step of the trajectory induced by increasing CutOut augmentation from $C = 0$ to $C = 4.0$. (c, f) Projection onto the first two principal components of landmark distortion matrix ΔZ, colored by CIFAR-10 class.

ensemble gains of augmentation pairs correlate with the mean geodesic angle between their trajectories (Fig. 4), suggesting that improvements from ensembling are connected with the geometric diversity of the learned representations of the individual models.

Landmark displacements show layer-dependent structure Lastly, we examine *how* individual probe inputs (landmarks) are displaced as changes in augmentation strength steer representation shapes. Each landmark $\mu = 1, ..., M$ is displaced by $\Delta z_\mu \in \mathbb{R}^N$ (Fig. 1), whose magnitudes and directions show which aspects of the shape are distorted along a trajectory. This serves as a window into the layer-by-layer mechanism by which the data augmentation process changes representations. Fig. 5 and Fig. 6 study properties of the landmark displacements for ResNet18, comparing unaugmented representations to those trained with Cutout augmentation of size 5. Panels on the top row correspond with analysis of representations from layer 2.1, an earlier hidden layer, while the bottom row show the same analysis for the penultimate avgpool representations. After sorting the landmarks by the magnitude of their displacement, we see that a small number landmarks are moved significantly more or less than the average displacement (Fig. 5 a, d). Image tiles in Fig. 5 (b) and (e) (and see also Fig. A-4, Fig. A-5 and Fig. A-6) show that maximally and minimally moved landmarks seem statistically related to each other in a way that is layer-dependent, with class relationships emerging in the layer layer (e). Principal components analysis is used in Fig. 5 (c) and (f) to study the primary axes of variation of each landmark displacement matrix ΔZ, as a way of probing whether groups of landmarks tend to be displaced in similar directions. We see from coloring the points by CIFAR-10 class (Fig. 5 (c) and (f)) that a relationship between class and landmark displacement for

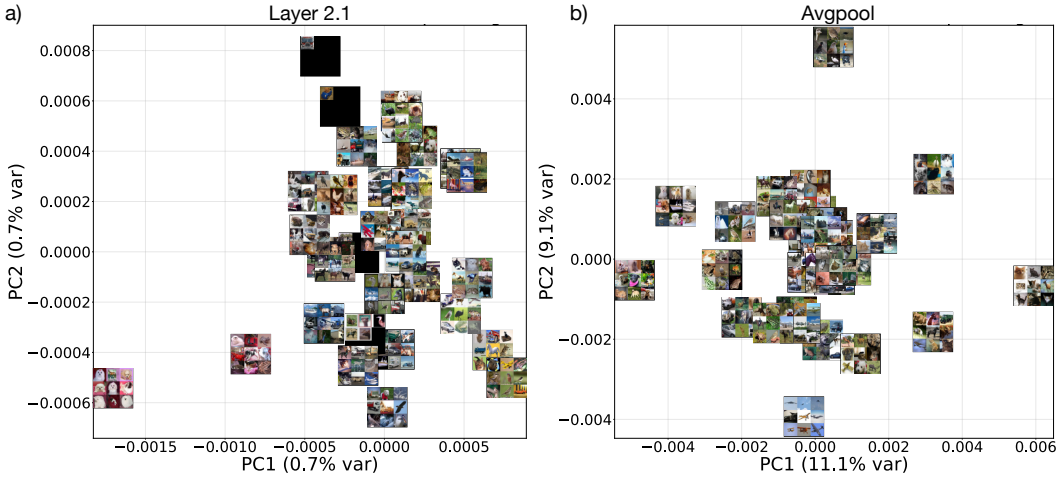

Figure 6: **K-means clustering of landmark distortions shows related landmarks may move in similar directions.** (a) The ResNet18 landmark distortion matrix of Fig. 5 a-c, ΔZ, is clustered using K-means and centroids are projected onto the first two PCs of ΔZ. Sample images from each cluster are plotted over the centroid in small tiles. (b) Same analysis as (a) except using landmark distortion matrix between avgpool layers. K is set using the elbow method. (a) and (b) are printed as full-page figures in Fig. A-7 and Fig. A-8.

this augmentation method emerges in the later layer representations, with this example indicating largest variability along axes pushing cat landmarks apart from dogs and airplanes away from birds. However, in Fig. 6, we perform K-means clustering on the ΔZ matrices for layer 2.1 and avgpool, in order to probe which sets of landmarks tend to move together under Cutout augmentation. We find that landmarks distorted in similar directions in layer 2.1 may share lower-level features or statistics, while in the avgpool layer images of the same class tend to be moved together (although subclusters within a class and merged classes are found). We conjecture that the properties and patterns of the landmark distortions could help elucidate the mechanisms by which data augmentation or other hyperparameters deform representation shape layer by layer, and this would be an exciting avenue for future work.

5 Discussion

In this work, we apply and validate a framework for analyzing learned representations using Riemannian shape distances, providing a principled geometry for studying how data augmentaiton, or hyperparameters in general, steers learned representation *across trained models*. By removing nuisance transformations, this framework isolates meaningful changes in representational structure that are induced by the training protocol. The embedding of neural representations in a well-defined metric space allows us to conceptualize changes in data augmentation hyerparameters as driving hidden representations along trajectories in shape space, which we find are consistent even across models independently trained end-to-end from different seeds and reproducible in different architectures. These trajectories are not random, but instead exhibit geometric consistency: representations of networks trained with stronger augmentations shift further along smooth paths, and distinct augmentation types steer representations in reliably different directions. This suggests that these representations of unaltered images encode a kind of geometric "signature" of the data augmentation training protocol. Interestingly, we find that the angle between data augmentation trajectories correlates with the improvement in accuracy achieved by ensembling the models trained with those methods. This perhaps suggests that an ensemble of geometrically diverse models in shape space could provide the improvements of ensemble learning with fewer overall models. Beyond data augmentation, this approach opens new directions for investigating how other aspects of the training pipeline, such as optimization, architecture, or supervision, shape learned geometry.

REFERENCES

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes, 2017. URL https://openreview.net/forum?id=ryF7rTqgl.

Matteo Alleman, Jack W Lindsey, and Stefano Fusi. Task structure and nonlinearity jointly determine learned representational geometry, 2024. URL https://arxiv.org/abs/2401.13558.

Gül Sena Altıntaş, Devin Kwok, Colin Raffel, and David Rolnick. The butterfly effect: Neural network training trajectories are highly sensitive to initial conditions. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=L1Bm396P0X.

Christopher M. Bishop. Training with noise is equivalent to tikhonov regularization. *Neural Computation*, 7(1):108–116, 1995. doi: 10.1162/neco.1995.7.1.108. URL https://doi.org/10.1162/neco.1995.7.1.108.

Shuxiao Chen, Edgar Dobriban, and Jane H Lee. A group-theoretic framework for data augmentation. *Journal of Machine Learning Research*, 21(245):1–71, 2020.

Tri Dao, Albert Gu, Alexander J. Ratner, Virginia Smith, Christopher De Sa, and Christopher Ré. A kernel theory of modern data augmentation, 2019. URL https://arxiv.org/abs/1803.06084.

Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

James J DiCarlo and David D Cox. Untangling invariant object recognition. *Trends in cognitive sciences*, 11(8):333–341, 2007.

Ian L. Dryden and Kanti V. Mardia. *Statistical Shape Analysis: With Applications in R*. John Wiley & Sons, 2nd edition, 2016. ISBN 978-0-470-69962-1.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Sam McCandlish, Jared Kaplan, Martin Wattenberg, Carol Chen, and Roger Grosse. Toy models of superposition. *arXiv preprint arXiv:2209.10652*, 2022. URL https://arxiv.org/abs/2209.10652.

Sarah E. Harvey, Brett W. Larsen, and Alex H. Williams. Duality of bures and shape distances with implications for comparing neural representations. In Marco Fumero, Emanuele Rodolá, Clementine Domine, Francesco Locatello, Karolina Dziugaite, and Caron Mathilde (eds.), *Proceedings of UniReps: the First Workshop on Unifying Representations in Neural Models*, volume 243 of *Proceedings of Machine Learning Research*, pp. 11–26. PMLR, 15 Dec 2024. URL https://proceedings.mlr.press/v243/harvey24a.html.

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.

D G Kendall. The diffusion of shape. *Adv. Appl. Probab.*, 9(3):428–430, 1977.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited, 2019. URL https://arxiv.org/abs/1905.00414.

Richard D Lange, Devin Kwok, Jordan Kyle Matelsky, Xinyue Wang, David Rolnick, and Konrad Kording. Deep networks as paths on the manifold of neural representations. In Timothy Doster, Tegan Emerson, Henry Kvinge, Nina Miolane, Mathilde Papillon, Bastian Rieck, and Sophia Sanborn (eds.), *Proceedings of 2nd Annual Workshop on Topology, Algebra, and Geometry in Machine Learning (TAG-ML)*, volume 221 of *Proceedings of Machine Learning Research*, pp. 102–133. PMLR, 28 Jul 2023. URL https://proceedings.mlr.press/v221/lange23a.html.

Karel Lenc and Andrea Vedaldi. Understanding image representations by measuring their equivariance and equivalence. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

Raphael Gontijo Lopes, Dong Yin, Ben Poole, Justin Gilmer, and Ekin D Cubuk. Improving robustness without sacrificing accuracy with patch gaussian augmentation. *arXiv preprint arXiv:1906.02611*, 2019.

Thao Nguyen, Maithra Raghu, and Simon Kornblith. Do wide and deep networks learn the same things? uncovering how neural network representations vary with width and depth. *CoRR*, abs/2010.15327, 2020. URL https://arxiv.org/abs/2010.15327.

BA Olshausen, CH Anderson, and DC Van Essen. A neurobiological model of visual attention and invariant pattern recognition based on dynamic routing of information. *Journal of Neuroscience*, 13(11):4700–4719, 1993. ISSN 0270-6474. doi: 10.1523/JNEUROSCI.13-11-04700.1993. URL https://www.jneurosci.org/content/13/11/4700.

Sylvestre-Alvise Rebuffi, Sven Gowal, Dan A. Calian, Florian Stimberg, Olivia Wiles, and Timothy Mann. Data augmentation can improve robustness, 2021. URL https://arxiv.org/abs/2111.05328.

Maximilian Riesenhuber and Tomaso Poggio. Hierarchical models of object recognition in cortex. *Nature neuroscience*, 2(11):1019–1025, 1999.

Connor Shorten and Taghi Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6, 07 2019. doi: 10.1186/s40537-019-0197-0.

Leslie N. Smith. A disciplined approach to neural network hyper-parameters: Part 1 – learning rate, batch size, momentum, and weight decay. *CoRR*, abs/1803.09820, 2018. URL https://arxiv.org/abs/1803.09820.

Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2018. ISBN 9781108415194.

Zaitian Wang, Pengfei Wang, Kunpeng Liu, Pengyang Wang, Yanjie Fu, Chang-Tien Lu, Charu C. Aggarwal, Jian Pei, and Yuanchun Zhou. A comprehensive survey on data augmentation, 2025. URL https://arxiv.org/abs/2405.09591.

Alex H Williams, Erin Kunz, Simon Kornblith, and Scott Linderman. Generalized shape metrics on neural representations. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 4738–4750. Curran Associates, Inc., 2021.

Sou Yoshihara, Taiki Fukiage, and Shin'ya Nishida. Does training with blurred images bring convolutional neural networks closer to humans with respect to robust object recognition and internal representations? *Frontiers in Psychology*, 14, 2023. doi: 10.3389/fpsyg.2023.1047694. URL https://www.frontiersin.org/journals/psychology/articles/10.3389/fpsyg.2023.1047694/full.

A APPENDIX

A.1 TRAINING DETAILS

Models are trained on CIFAR-10 using four different architectures and the standard train/test split. Unless otherwise specified, training hyperparameters are fixed for each architecture and are reported in 1. Hyperparameters are selected such that they reach a reasonable classification performance while also maintaining a smooth and stable learning curve.

Table 1: Training Hyperparameters for CIFAR-10 Models

Hyperparameter	**ResNet-18**	**VGG-16**	**DenseNet-40**	**ViT-Tiny**
Optimizer	SGD	SGD	SGD	AdamW
Momentum	0.9	0.9	0.9	–
Weight Decay	5×10^{-4}	5×10^{-4}	0.01	5×10^{-2}
Batch Size	256	256	256	256
Epochs	100	100	100	50
Learning Rate Schedule	One-cycle	One-cycle	One-cycle	One-cycle
Max Learning Rate	0.03	0.05	0.01	0.001
Warmup Epochs	0	0	5	5

The data augmentations used in our experiments for Fig. 4 are listed in Table 2. For each experiment, we fix the augmentation type and sweep over a predefined set of magnitudes, while keeping the model architecture, optimizer, learning rate schedule, and number of epochs unchanged. All augmentations are applied stochastically on-the-fly during training.

Table 2: Data Augmentation Hyperparameters

Augmentation	**Value(s)**
Rotation	$\pm\theta \in \{4, 12, 18, 24\}$ degrees
Shear	$\pm s \in \{4, 12, 18, 24\}$ degrees
Color Jitter	$c \in \{0.1, 0.2, 0.3, 0.4\}$ for Brightness/Contrast/Saturation, hue $= c/4$
Gray Scale	Convert images to grayscale with $p \in \{0.03, 0.05, 0.08, 0.1\}$
Gaussian Noise	$\sigma \in \{0.01, 0.02, 0.030.04\}$
Salt-and-Pepper Noise	Pixel probability $p \in \{0.05, 0.1, 0.15, 0.2\}$
Random Crop	Scale $c \in \{0.6, 0.7, 0.8, 0.9\}$
Cutout	size k $\in \{2, 4, 6, 8\}$

A.2 SUPPLEMENTARY FIGURES

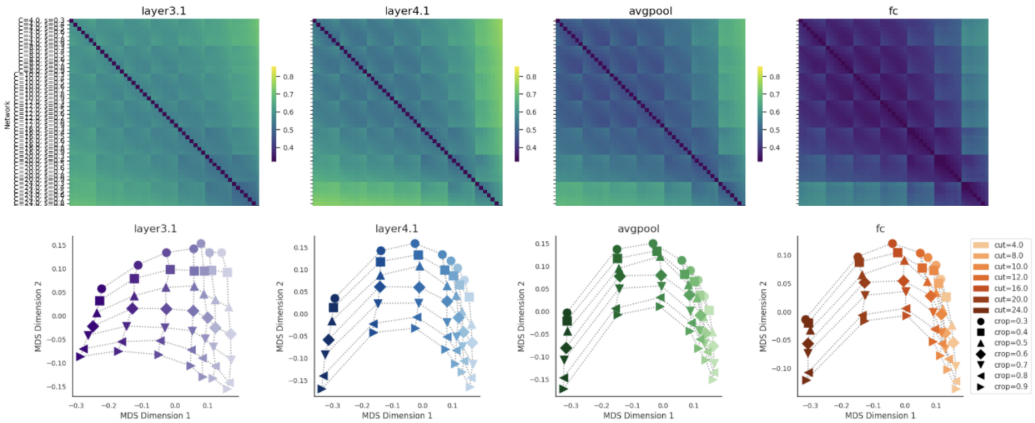

Figure A-1: Replicating the experiment of Fig. 2 also in ResNet18 but using the data augmentation methods Cutout and random cropping.

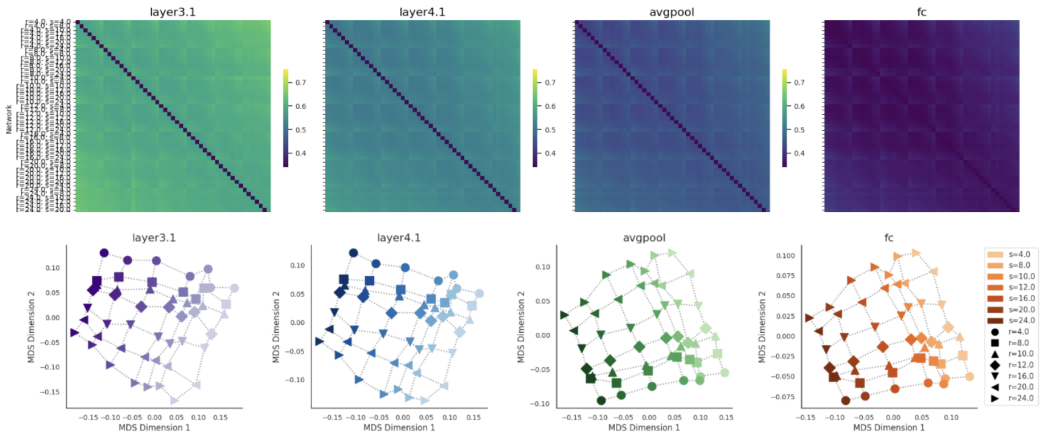

Figure A-2: Replicating the experiment of Fig. 2 also in ResNet18 but using the data augmentation methods rotation and sheer.

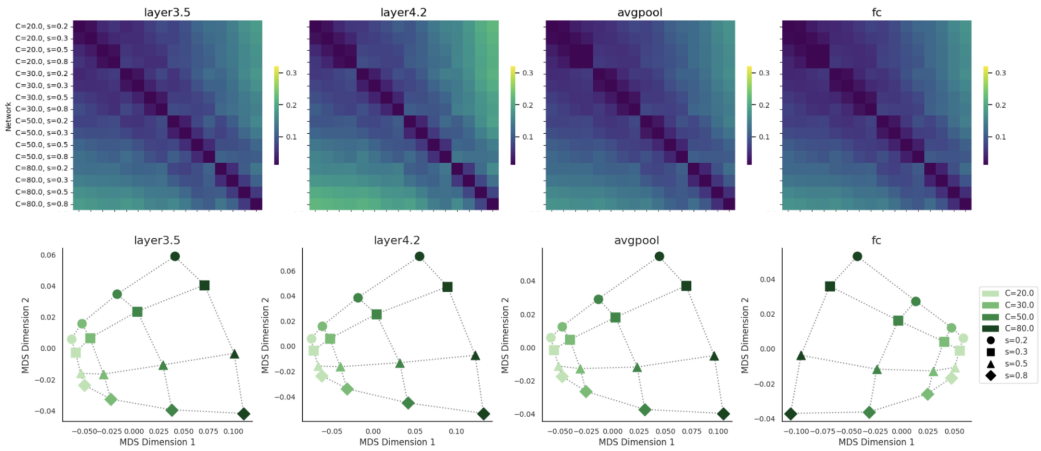

Figure A-3: Replicating the experiment of Fig. 2 in WideResNet50 but pretrained on the Imagenet dataset with clean images, and fine-tuned with data augmentation method patch Gaussian.

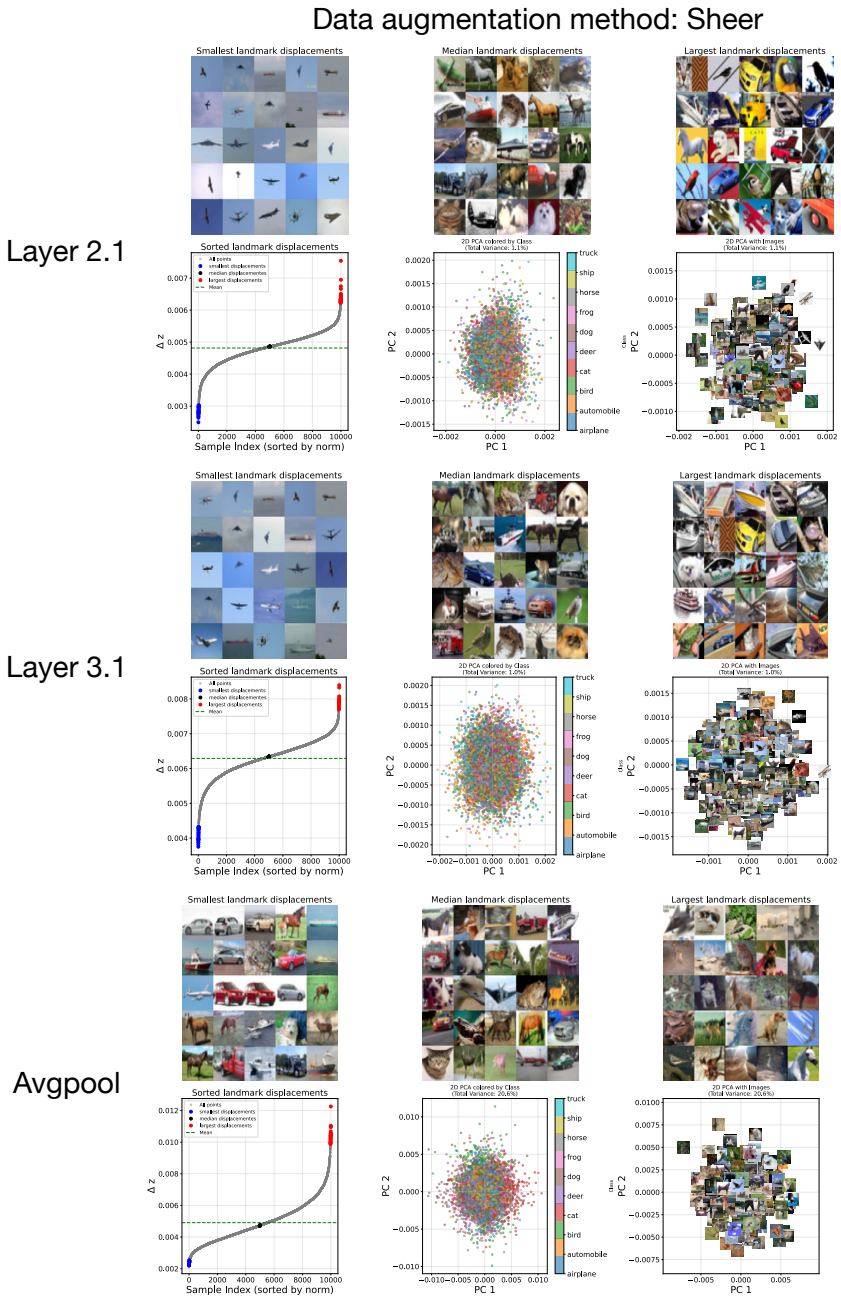

Figure A-4: Layer-by-layer variability in maximally, median, and minimally deformed landmarks. Landmark deviation analysis as in Fig. 5 comparing layers of ResNet18 trained on un-augmented images and trained using data augmentation method sheer at level s = 4 (one step along the sheer trajector in Fig. 4 a).

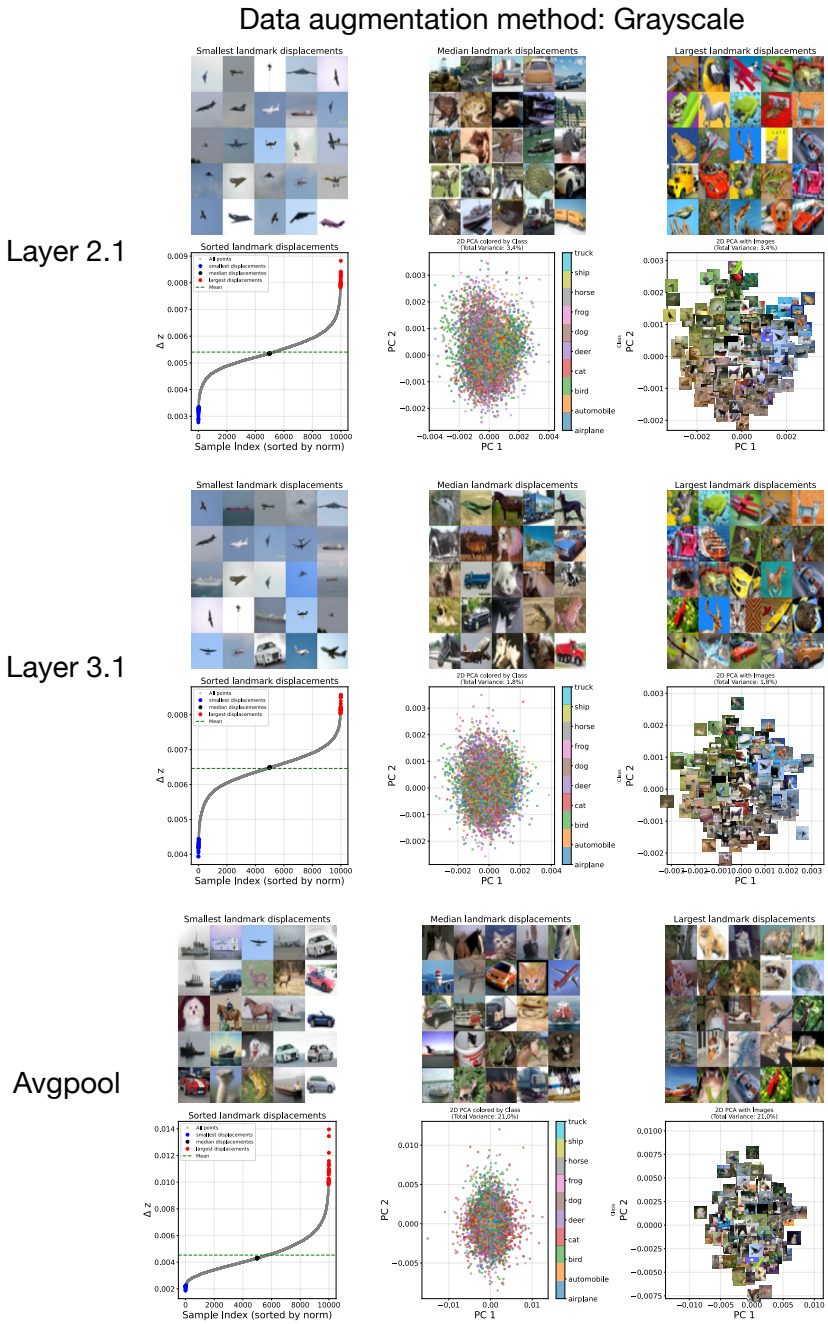

Figure A-5: Layer-by-layer variability in maximally, median, and minimally deformed landmarks. Landmark deviation analysis as in Fig. 5 comparing layers of ResNet18 trained on un-augmented images and trained using data augmentation method ‘grayscale’, with probability of image grayscale = 0.1 (one step along the grayscale trajectory of Fig. 4).

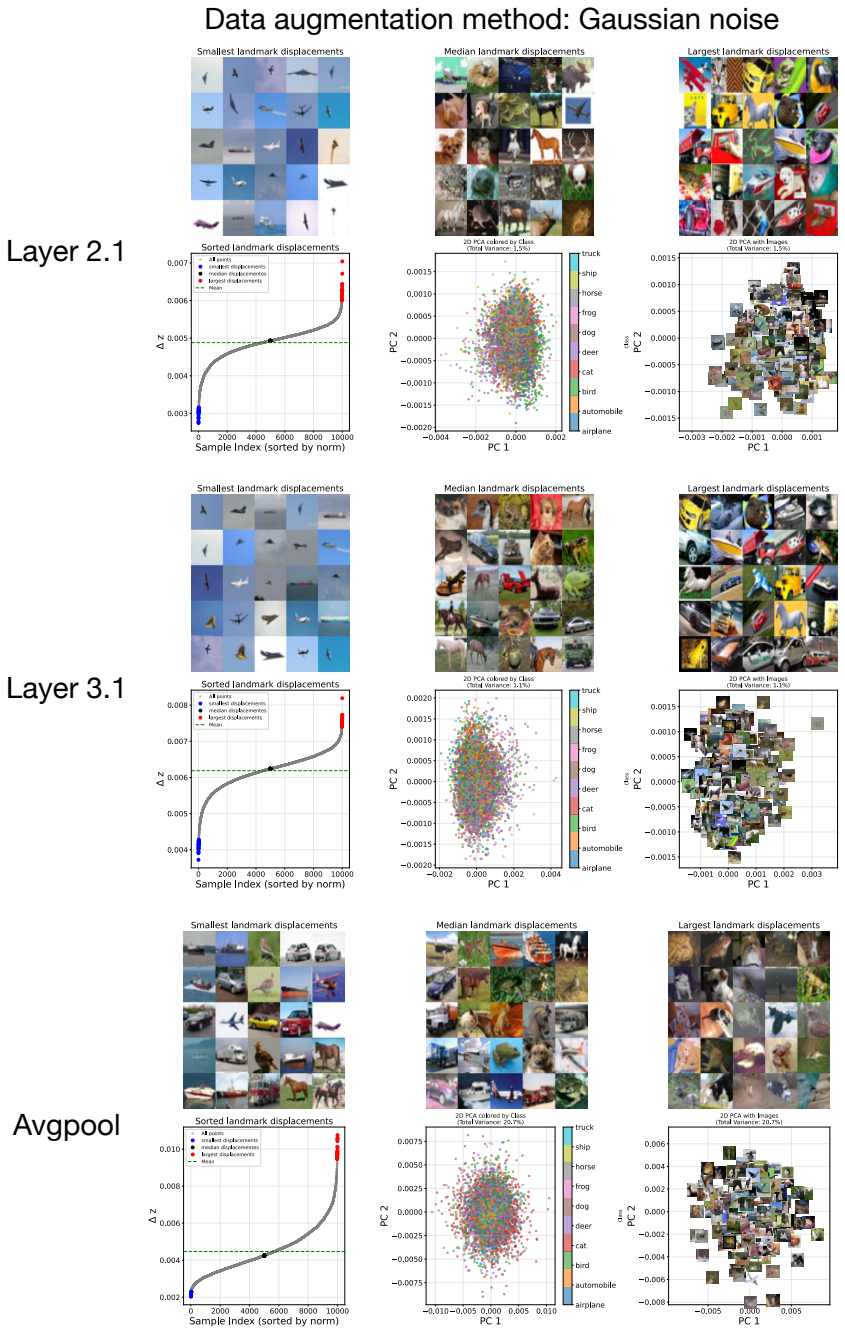

Figure A-6: Layer-by-layer variability in maximally, median, and minimally deformed landmarks. Landmark deviation analysis as in Fig. 5 comparing layers of ResNet18 trained on un-augmented images and trained using data augmentation method Gaussian noise at level s = 0.02

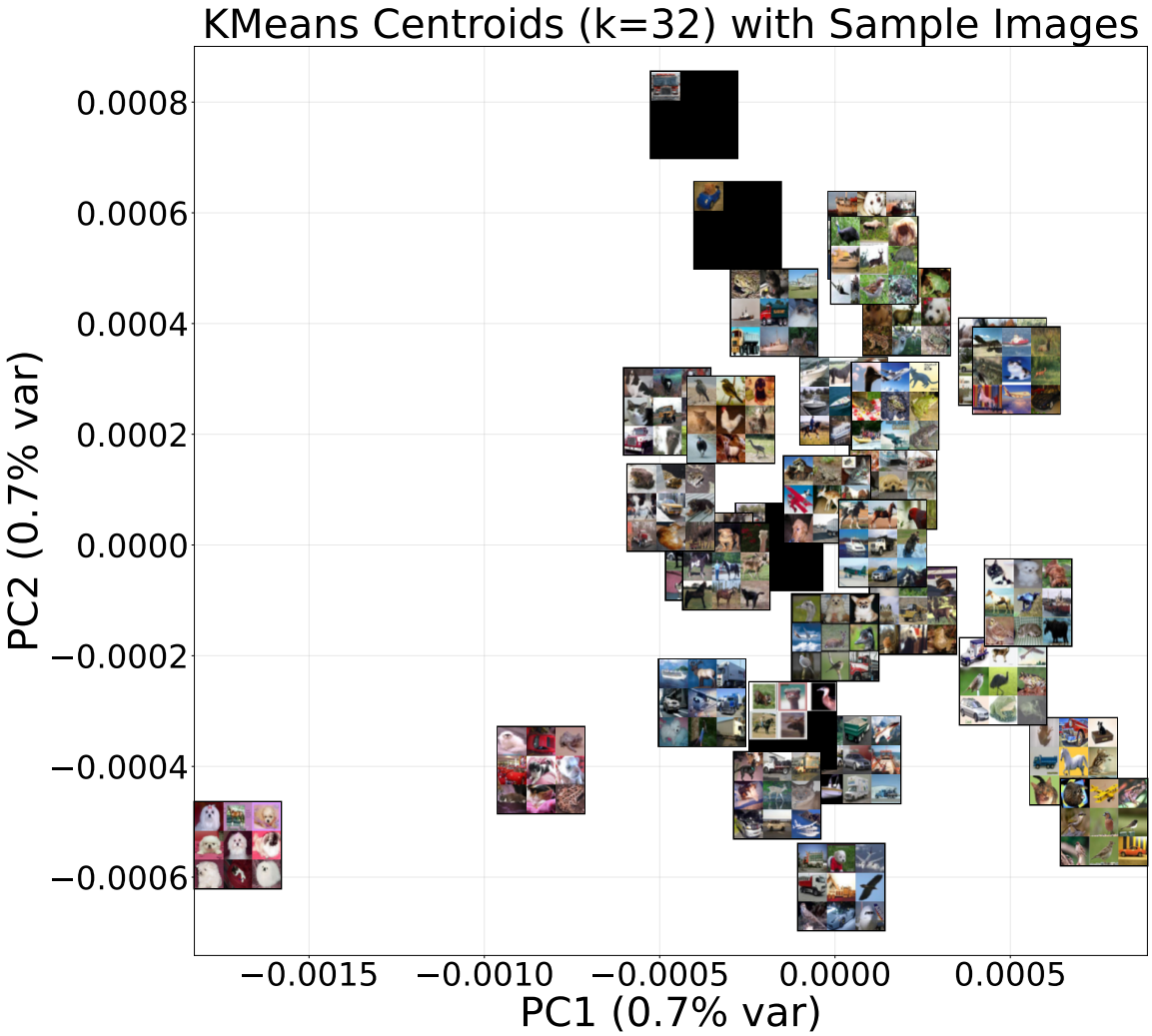

Figure A-7: Fig. 6 (a) is reproduced for readability.

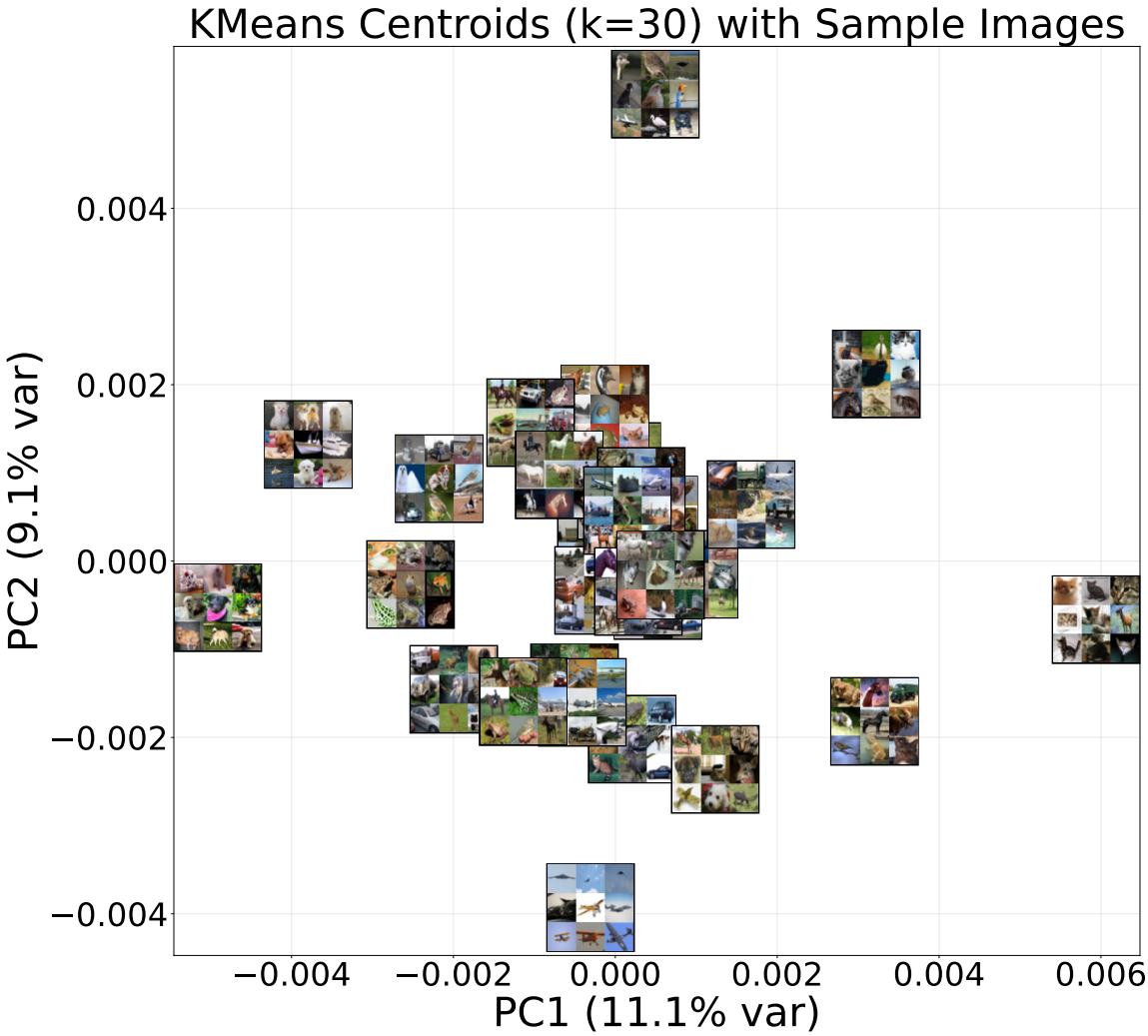

Figure A-8: Fig. 6 (b) is reproduced for readability.