Linking in Style: Understanding learned features in deep learning models

Maren H. Wehrheim[1,2], Pamela Osuna-Vargas[1], and Matthias Kaschube[1,2]

[1] Frankfurt Institute for Advanced Studies (FIAS), Frankfurt, Germany
[2] Department of Computer Science and Mathematics, Goethe University Frankfurt, Frankfurt, Germany
{wehrheim, osuna, kaschube}@fias.uni-frankfurt.de

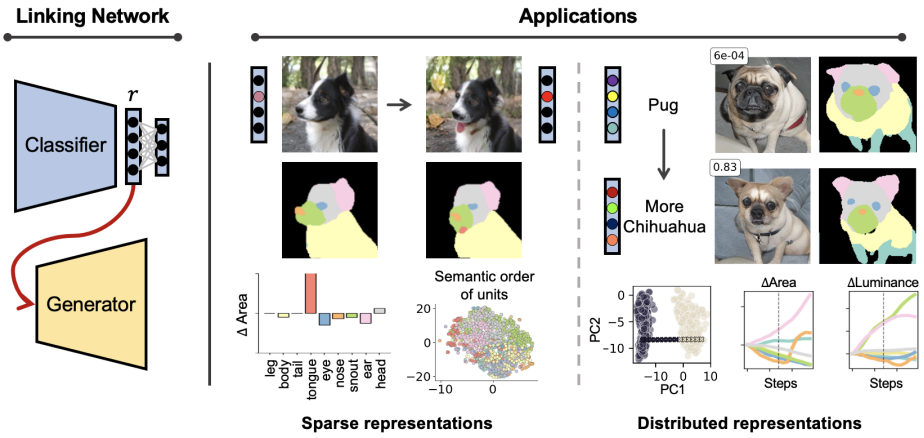

Fig. 1: Visualization and systematic quantification of a classifier's learned representations. **Left:** We introduce a linking network (red arrow) that links an activation pattern $r \in R$ in the penultimate layer of a classifier to the latent space of StyleGAN-XL [57], thereby visualizing the representations learned by the classifier. Building on these visualizations, we propose a pipeline to automatically and objectively analyze a large number of learned representations in R by evaluating the changes between images caused by perturbations in R. We show two applications of how our method can be used to understand learned features in deep learning models. **Middle:** We systematically 'tune' the activations of all units in R separately to obtain a comprehensive overview across sparsely encoded representations across thousands of units. **Right:** The linking network can visualize counterfactual examples and our quantification pipeline reveals trajectories that provide insights into learned concepts relevant for the classifier's decision.

Abstract. Convolutional neural networks (CNNs) learn abstract features to perform object classification, but understanding these features remains challenging due to difficult-to-interpret results or high computational costs. We propose an automatic method to visualize and systematically analyze learned features in CNNs. Specifically, we introduce a linking network that maps the penultimate layer of a pre-trained classifier to the latent space of a generative model (StyleGAN-XL), thereby

enabling an interpretable, human-friendly visualization of the classifier's representations. Our findings indicate a congruent semantic order in both spaces, enabling a direct linear mapping between them. Training the linking network is computationally inexpensive and decoupled from training both the GAN and the classifier. We introduce an automatic pipeline that utilizes such GAN-based visualizations to quantify learned representations by analyzing activation changes in the classifier in the image domain. This quantification allows us to systematically study the learned representations in several thousand units simultaneously and to extract and visualize units selective for specific semantic concepts. Further, we illustrate how our method can be used to quantify and interpret the classifier's decision boundary using counterfactual examples. Overall, our method offers systematic and objective perspectives on learned abstract representations in CNNs. `https://github.com/kaschube-lab/LinkingInStyle.git`

1 Introduction

Deep learning models learn abstract concepts in their hidden layers when trained to perform a task. However, the models do not provide plausible explanations when they fail, thereby hampering their trustworthiness. Unraveling the learned concepts that influence a classifier's decisions can reveal inherent biases [20, 37] or identify failures in these models [48, 65].

Recent work has focused on interpreting deep learning models' behavior by explaining their weights, units, subnetworks, or latent representations [56]. Individual units in deep neural networks (DNNs) have been shown to be selective for single human-interpretable concepts such as faces, food, textures, or even multimodal concepts [3, 4, 18, 43, 53, 69, 71]. It has since been an open debate whether DNNs learn disentangled, sparse representations in individual units or whether representations are distributed across many units, a phenomenon often referred to as feature superposition [14, 22, 24] and hypothesized to contribute to adversarial vulnerability [15, 17].

Recent efforts have also been dedicated to understanding the representations that form the decision boundaries in DNNs trained for visual object classification [25, 32, 33, 63]. Counterfactual explanations present an empirical perspective for interpreting how deep learning models make decisions. Consider an instance of a given class (e.g., an image of a dog), a *counterfactual example* represents a slightly altered version of that instance such that the classifier predicts a target class (e.g., cat). Crucially, the change in the original instance should be minimal and human-interpretable to be effective. This excludes adversarial examples [8, 68, 70], where small pixel perturbations change the prediction but remain unrecognized by humans.

For computer vision applications, explainability methods greatly benefit from providing human-comprehensible visualizations of single examples. However, human visual inspection is inherently subjective and only possible for a few features, prohibiting an unbiased and systematic evaluation of the high-dimensional

representations in CNNs. Studying all potential configurations of representations poses an intricate combinatorial challenge, hence visual inspection soon becomes infeasible and cannot provide a comprehensive and objective understanding of learned features in hidden layers.

Generative adversarial networks (GANs) [19] are characterized by a latent space that is continuous and semantically structured, enabling the visualization of feature representations, including counterfactual examples [39, 62]. However, as GANs were usually trained with a single data category to ensure the generation of high-quality images, their ability to visualize learned representations in classification models required extensive (re-)training and remained infeasible for multi-class categorization problems. Only recently, the StyleGAN-XL [57] allows to generate images of all ImageNet classes from a single latent space.

In this work, we present a broadly applicable method to objectively and systematically analyze features encoded in the penultimate layer of CNNs trained for object classification. This is achieved in two steps: Firstly we establish an efficient feature visualization tool based on a pre-trained StyleGAN-XL that can be flexibly linked to various pre-trained CNNs (overcoming extensive (re-)training strategies of several previous studies). Specifically, we introduce a linking network that connects the penultimate layer, here termed *representation space*, of a CNN to the latent space of a pre-trained StyleGAN-XL. Linking these two spaces allows us to visualize arbitrary feature dimensions in the classifier. Secondly, we establish methods for an automatic assessment of learned features in the classifier's representation space using unsupervised tracking methods [54] and few-shot image segmentation [64]. We envision our pipeline to offer novel research applications and show examples in Sec. 4. First, we analyze and quantify the features encoded in each of the several thousand units of the penultimate layer to build summary statistics of a classifier's learned concepts. This also enables us to reveal class-relevant units encoding human-interpretable features, shedding new light on the recurring question of whether features are represented in individual units in a rather disentangled or superimposed fashion. Second, we probe the classifier's decision boundary to identify and interpret the most relevant features underlying classification. Our contributions are as follows:

- A simple and easy-to-train linking network to visualize learned representations in CNNs.
- An automated pipeline to quantify these high-dimensional representations enabling their systematic analysis and objective characterization via summary statistics.
- We highlight two applications of our method: i) to reveal learned abstract concepts in single units and ii) to examine a classifier's decision boundaries.

2 Related Work

Explainable AI (XAI) aims to provide human-understandable explanations for the features learned and decisions made by an AI system. In this context, some research argues for the relevance of sparse representations to encode abstract

features within single units [2,5,11,12,16,41,47], others highlight the importance of distributed highly robust representations [13,40,45]. A variety of methods that explain the learned representations in pre-trained models exist [1, 56], including GradCAM [58], DeepLIFT [61] or LIME [55]. These methods usually visualize single features or saliency maps but do not quantify the *what* and *how*, e.g., larger eyes or different color, without additional user input.

GANs generate near photorealistic images [7, 19, 31, 34–36] and manipulating their latent code smoothly alters features of the generated image [26, 28, 29, 50, 59, 60, 66]. Recent work shows that semantic concepts in GANs allow to generate image segmentation masks, using, for example, unsupervised clustering [49,67], few-shot learning [64], or self-supervised contrastive approaches [44]. Representations of (pre-trained) classifiers have previously been used to guide the generative process or to build meaningful latent spaces [6,9,39,62]. In recent work, GANs have been used to visualize changes in single attributes for counterfactual examples of a classifier [39,62]. Lang et al. [39] incorporate a GAN in the training procedure of the classifier and then extract user-defined attributes that change the classifier's output. However, this approach is computationally expensive and does not allow to interpret single units in the classifier.

Previous work on GAN-based image editing or counterfactual explanations in computer vision often focuses on visualizing learned representations or relies on user input to quantify the learned concepts [21,23,28,29,46,50,59,60,66]. [30] and [52] define a set of attributes a priori such that a manipulation induces a change in the predicted category. Other methods rely on text guidance to generate difficult images for the classifier [51] or identify the classifier's sensitivity to certain features [42]. However, methods that enable an objective and comprehensive study of learned(class-relevant) features using photorealistic visualizations and flexibly allowing for an analysis of any combination of units are still lacking.

3 Method

Next, we describe our approach for uncovering learned representations in a classifier. The core of our method is a linking network that learns a mapping between the classifier and the latent space in StyleGAN-XL (Sec. 3.1). We then introduce a pipeline to automatically analyze learned concepts (Sec. 3.2). Finally, we propose different applications, demonstrating how our method can be used to understand single units as well as distributed representations relevant for a classifier's decision (Sec. 3.3).

3.1 Linking network

We introduce a linking network that establishes a connection between the classifier and the GAN to visualize learned representations in the classifier (Fig. 2 red arrow). We utilize the recently proposed pre-trained StyleGAN-XL [57], as it produces high-quality images and learns a single latent code across all classes without extra class-conditional input in the higher layers of the generator. For

a given style code $w \in W$, corresponding to a non-linear combination of a class embedding vector c and a random latent vector z, the generator G_s creates an image I. As we are interested in studying the internal representations of CNNs, we input the generated image into the classifier and extract the respective activation pattern r in the penultimate layer, which we call the representation layer R. Following this procedure, we generate 5,000 training instances of (w, r)-pairs for each class and train a network that maps the activations into W:

$$\tilde{w} = f(r), \tag{1}$$

where $\tilde{w}$ is the predicted w, f is the linking network and $r \in R$ is a specific activation vector. Thus, the linking network acts as a bridge between the classifier and the generative model and offers the possibility to perform a full cycle, that is $w \rightarrow I \rightarrow r \rightarrow \tilde{w} \rightarrow \tilde{I}$. In the simplest scenario, we use a linear regression model to fit a linear mapping between the two spaces based on the least-square distance (see Supplement for more complex linking networks).

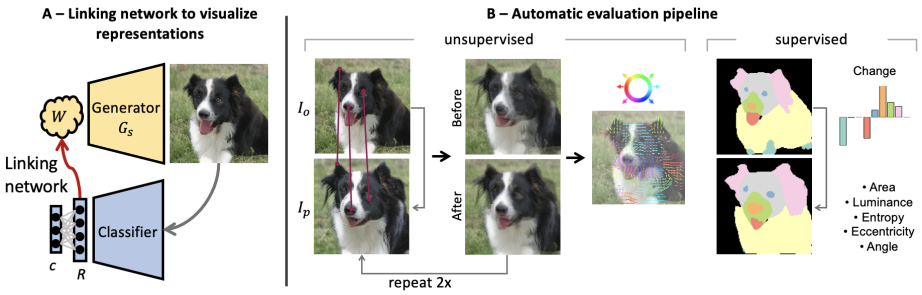

Fig. 2: Visualizing and quantifying learned features in CNNs. A) The generator G_s generates an image I from a given $w \in W$. I is input to the classifier from which the corresponding activation vector $r \in R$ is extracted. Using a set of (w, r)-pairs, we train a linking network (red arrow) to create a link between the classifier and the GAN. We then perturb the activation pattern r to visualize learned representations in R using the GAN. **B)** Automatic quantification of semantic concepts. Left (unsupervised): We introduce an unsupervised method to find matching points between images I_o and I_p. First, we use PUMP [54] to compute an affine transformation and align the two images to remove global changes such as translation or zoom (center, top: non-aligned images, bottom: aligned images). We then compute PUMP again to find local changes not accounted for by the affine transform and compute the vector field to visualize the changes. Right (supervised): We compute the segmentation mask for each image separately following [64]. Then, we quantify each semantic label in an image according to different evaluation metrics: area (shown here), luminance, entropy, eccentricity, and angle. For each metric, we compute the change induced by a perturbation in R.

3.2 Analyzing learned representations

CNNs learn abstract semantic concepts like eyes or faces. Whereas humans easily visually detect abstract concepts, quantifying them is challenging. In addition to visualizing examples of learned representations, we here also introduce two methods to objectively and systematically quantify semantic concepts learned by the classifier to increase the explainability of CNNs. Specifically, we introduce an unsupervised method that aids visual inspection and quantifies regional features. Additionally, we propose a supervised method to facilitate the interpretation of learned features using image segmentation (area, luminance, entropy, angle, eccentricity). The features analyzed by these two methods are revealed by comparing an original image I_o to an image I_p generated after perturbing I_o's representation r.

Unsupervised local descriptors matching We use an unsupervised motion tracking pipeline to reveal learned features by analyzing differences across images associated with perturbations in the classifier's activation space (Fig. 2B unsupervised). First, we use PUMP [54], an unsupervised method that finds pixel correspondences between I_o and I_p, to compute an affine transformation. We then apply the affine transformation to I_p, hence minimizing the effect of global displacement (rotation, scaling, etc.). Finally, we again compute PUMP to find the set of dense correspondences and visualize the vector field of local changes.

Semantic concept quantification through image segmentation We hypothesize that a classification model learns abstract semantic concepts (e.g., ear shape, fur type) to differentiate between classes. To capture changes in such semantic concepts, we adopt a few-shot image segmentation approach (Fig. 2B supervised) based on [64]. This method leverages the intermediate activation output of the generator G_s to learn an image segmentation model from few labeled examples only. The few-shot nature and generalizability across classes (see Supplement) of this approach reduce the workload required to produce annotations, thus allowing for more detailed labels. To reduce the computational cost we here use only every second layer of G_s and additionally downsample the output to 128×128 pixels (instead of 256×256). We label five images per class for a subset of classes and train a segmentation model with three convolutional layers for 100 epochs. For any generated image, we can then compute different metrics to quantify the segmented parts, each of which represents a semantic concept (e.g. eyes, tongue). First, we extract the area, luminance, and entropy (smoothness) for each identified segment. Finally, to study its shape and orientation, we fit an ellipse to each segmented component and compute the eccentricity and angle. In total, we characterize each image by 45 measures (5 metrics, 9 semantic labels). Moreover, to test whether a perturbation in r affects only one or many semantic

concepts, we define the label sparsity s according to [27]

$$s(x) = \frac{\sqrt{k} - \frac{\|x\|_1}{\|x\|_2}}{\sqrt{k} - 1}, \tag{2}$$

where $x \in \mathbb{R}^k$ is the change in label vector induced by a perturbation in r and k represents the label vector's dimension (here, $k = 9$ considering a metric separately (e.g., area), or $k = 45$ for the complete set of measures). If s is close to one, the label vector is sparse, whereas values approaching zero describe distributed representations.

3.3 Applications

We propose several applications to systematically analyze the learned features in the classifier's representation space using our linking network and automatic quantification pipeline. First, we analyze the representations learned by single units. To this end, we map an image I_o into the classifier and extract the activation $r \in R$. We then alter r of a unit of interest, map the perturbed activation r_p into W, and generate an image I_p. We can continuously visualize a whole trajectory of the representations encoded in a single unit by linearly altering that unit's activation (between the empirical minimum and maximum) and generating the corresponding sequence of images. We repeat this process for all units in R and quantify the encoded representations using the changes along the image sequence to compute summary statistics across several thousand units and to identify units with certain properties.

Second, we analyze distributed representations that form the classifier's decision boundary. Specifically, we find counterfactual explanations by changing r of an image of class c_{orig} such that the prediction logits of a target class $o_{c_{target}}(r)$ are maximized with minimal changes only. We therefore minimize:

$$L(r, \Delta r) = -o_{c_{\text{target}}}(r + \Delta r) + \lambda_1 o_{c_{\text{orig}}}(r + \Delta r) - \lambda_2 L_{\text{ID}} \tag{3}$$

where λ_1 and λ_2 are weighting coefficients, empirically set to $\lambda_1 = 0.6$, $\lambda_2 = 10$. L_{ID} is an extra penalization term to preserve the identity of the object in the W-space:

$$L_{\text{ID}}(r, \Delta r) = \frac{f(r) f(r + \Delta r)}{\|f(r)\|_2 \|f(r + \Delta r)\|_2} \tag{4}$$

The shift Δr is optimized using gradient descent until the predicted class for image $G_s(f(r + \Delta r))$ is c_{target} but at most for 2,000 steps.

4 Experiments & Results

In the following sections, we demonstrate the feasibility (Sec. 4.1) and performance (Sec. 4.2) of our proposed linking network. We then demonstrate how our method can aid our understanding of the learned representations in the hidden layers of a classifier (Sec. 4.3 - 4.5). In the main text, we report results with the ResNet-50 classifier (see Supplement for other classifier architectures).

4.1 Similarities between W and the representation space R

In this work, we introduce methods to interpret learned representations in CNNs trained on object classification by linking the representation space R of the classifier to the W-space in StyleGAN-XL. Finding a simple (or even linear) mapping between R to W may be possible if the representations within the two spaces are sufficiently similar, which we study in the following.

First, since the penultimate layer of the classifier contains class-specific representations, we test if the W-space also separates classes. We test this using k-means clustering due to its simplicity, but other clustering or classification methods (supervised or unsupervised) could be used instead. Specifically, we repeatedly fit a k-Means clustering ($k = 5$) to five randomly selected ImageNet classes and evaluate the performance using the Adjusted Rand Index (ARI) between the predicted clusters and the real class labels. We observe clustering by classes in both, the representation space and the W-space (Fig. 3A left), indicating that R and W separate classes to a similar degree.

Next, we use representation similarity analysis (RSA) to compare R and W on the representational level [38]. We first compute similarity matrices (based on pairwise correlations) and dissimilarity matrices (based on the Euclidean distance) in W and R separately across sets of 5 randomly chosen classes (as above) and then compute the correlations between these (flattened) matrices in W and R revealing high similarities between the two spaces (Fig. 3A right).

The robust clustering performance signifies a substantial degree of class demarcation within both spaces. The high representation similarity suggests a congruence in the representation of abstract concepts between W and R. Together, these findings suggest that simple (linear) models may be adequate for establishing a functional linkage between these two spaces.

4.2 Linking the representation space to W

The core of our method is a linear regression model that we train to link the representation space R in the classifier to the W-space in the StyleGAN-XL (Fig. 2 red arrow, see Supplement for more complex (non-linear) linking networks). First, we generate 5,000 images per class for several classes, all of which are correctly classified. We then encode these images into the classifier and extract the activations in the representation layer R. We then train the linear regression on the pair of activations $r \in R$ and the corresponding $w \in W$, using 5,000 examples per class.

We assess the performance of this linking network in W as well as in the image domain using a newly generated test set. We observe that after mapping an image for a full cycle, i.e., $w \rightarrow I \rightarrow r \rightarrow \tilde{w} \rightarrow \tilde{I}$, the mapped image is highly similar to the original image (Fig. 3B). We quantify the loss in W as the mean squared error (MSE) between the original w and the cycled prediction $\tilde{w}$ and observe a high performance (loss values close to 0) that is significantly better than that obtained for randomly selected images (Fig. 3C left). Moreover, we obtain consistent results for a comparison within the image domain using the

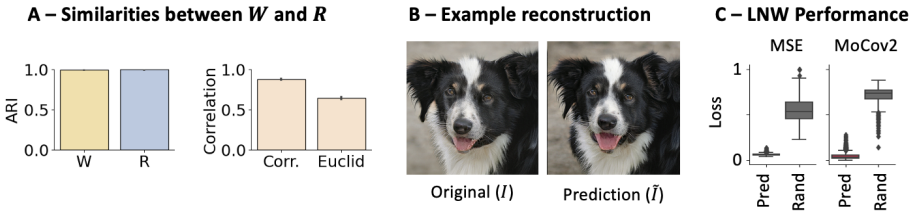

Fig. 3: Feasibility and performance of linking network. A) High similarity between StyleGAN-XL's W-space and representation space R in ResNet-50. Across 100 repetitions, 100 examples for five different ImageNet classes are sampled. **Left:** We fit a k-Means clustering ($k = 5$, 20 initializations) on the selected examples and compute the Adjusted Rand Index (ARI) between the predicted clusters and the real class labels. **Right**: Learned representations in W and R are highly similar, shown here by high average correlations between flattened similarity (correlation) and dissimilarity (Euclidean distance) matrices of the selected examples computed for the two spaces. **B)** The trained linking network achieves high similarities between generated images (I) and images cycled through the linking network and the GAN ($\tilde{I}$). **C)** We quantify the performance of the linking network by the MSE between w and $\tilde{w}$ and by the perceptual image distance MoCov2 [10] between I and $\tilde{I}$.

perceptual distance measure MoCov2 [10] (Fig. 3C right, see Supplementary Fig. S1 for other image similarity metrics).

4.3 Conceptualizing single-unit representations

In this section, we demonstrate how our method can be used to effectively visualize abstract concepts encoded in individual units in the classifier, and to quantify systematically such representations across large numbers of units. For a given input image and unit, we incrementally alter its activation, a process that yields a sequence of images, enabling a visual inspection of variations in salient features encoded in that unit. Our unsupervised method additionally visualizes these features in a vector field (Fig. 4). As a classifier's hidden layers contain large numbers of units (here 2,048), such that visual inspection is impractical, we propose an automatic analysis pipeline allowing us to examine *all* units in R. Specifically, we use semantic image segmentation to evaluate changes in area, luminance, entropy, eccentricity, and rotation angle of each segmentation label. We find individual units that represent specific features such as gender or color (see Fig. 4A). These single-unit representations can be robust across classes (Fig. 4B) and differ between units (Fig. 4C). Note that our method also reveals relevant features when the classifier and the GAN were trained on similar yet different datasets (see Supplementary Fig. S4).

Next, to test whether R contains disentangled feature representations, we quantify each unit's label sparsity (Eq. 2). We now focus on different dog classes, as dogs share many features but also show a high variability such as different fur colors, or ear shapes. We perturb each unit separately and compute the label

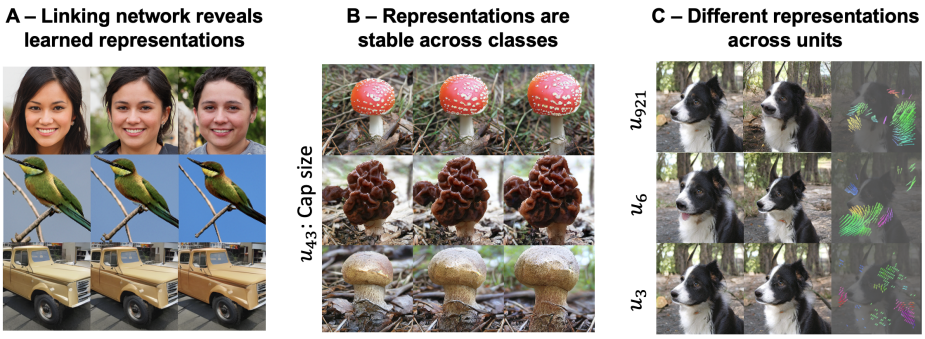

Fig. 4: Automatically revealed abstract concepts encoded in individual units. We tune the activation of individual units in R and visualize the results. We observe abstract concepts to be encoded in single units, such as gender or color (**A**), and to be stable across different classes, shown here for the cap sizes of different fungi (**B**). **C**) Different units encode different concepts that can be visualized with our unsupervised tracking method (vector fields).

sparsity from the median change in each label (e.g., legs, body, etc.; Eq. 2) across 100 example images. We find long tail distributions of label sparsity, with few, highly sparse units, that vary between classes (Fig. 5A left), showing highly disentangled representation for features such as legs or ears (Fig. 5A right). Next, asking whether R exhibits a semantic order, we visualize single-unit representations using tSNE (Fig. 5B). We find some labels to occupy specific regions in the representational space, with interdependence (overlap) between several labels (e.g. snout, ear, and head). Clustering the semantic concepts reveals disentangled (Fig. 5C, cluster 9) as well as entangled representations (Fig. 5C, clusters 18 and 65) suggesting superposition of several labels.

4.4 Class relevance of single units

Next, to systematically reveal the influence of single units on the classifier's prediction, we perturb each unit individually, visualize the new image, and extract the new prediction probability from the classifier given this image. We find that several units have a discernible effect on the prediction (Fig. 6A). Our systematic quantification of the entire representation space allows us to identify class-relevant units (average change in probability greater than 0.15) that are unique to single classes, as well as a few units that are relevant to several classes (Fig. 6B). Further, correlating the changes in prediction probability associated with each unit across classes reveals clusters of classes suggesting similar representational manifolds (e.g. between Chihuahua, Weimaraner, and Pug; Fig. 6C). Additionally, visualizing the representations encoded in the class-relevant unit shared between most example classes (Fig. 6B) reveals that such units can encode human-interpretable semantic concepts such as color or the length of the snout and can even cause a change in the predicted class (Fig. 6D, see the

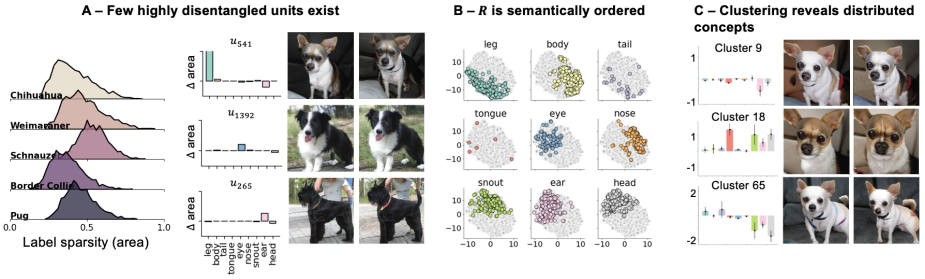

Fig. 5: Overview of features represented by individual units for the 2,048 units in the hidden layer of the ResNet-50 classifier. A) Left: We compute the label sparsity of each quantification metric (here area, see Supplement for other metrics) across all units for 100 test seeds. Different classes exhibit different levels of sparsity. Right: Highly sparse units reveal disentangled representations of concepts such as long legs, larger eyes, or longer ears. **B**) R is semantically ordered. We encode all changes in area induced by single-unit perturbations into a low-dimensional space using tSNE and color units by the label with the strongest change. Regional overlap between labels indicates interdependent representations of these concepts; as observed for snout, ear, and head, but not for legs and body. **C**) Hierarchical clustering of the label vectors reveals clusters representing disentangled concepts (cluster 9) as well as combinations of previously observed overlapping concepts (clusters 18 and 65).

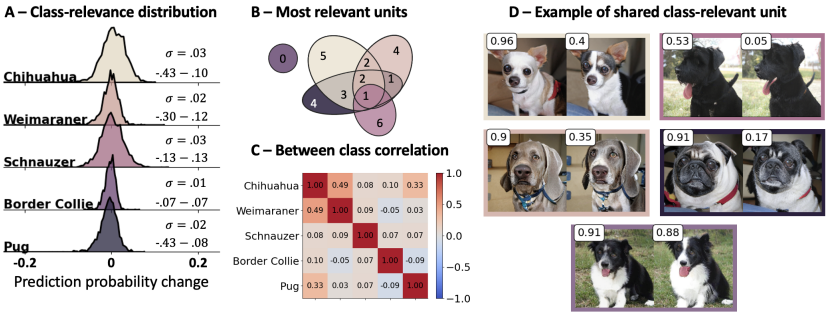

Fig. 6: Single units encode class-relevant representations. We analyze the effect of single-unit perturbations on the classifier's prediction probabilities. **A**) We depict the distributions of the change in prediction probability (softmax) for five example classes averaged across 100 test seeds per class. **B**) Class-relevant units can be shared between classes. The Venn diagram shows the number of highly class-relevant units (average change in probability higher than 0.15) in each class. Further, our method can uncover robust classes that are not affected by single-unit perturbations (here e.g., Border Collie). **C**) Correlating the changes in the classifier's prediction over all units reveals similarities in class encodings. Whereas Chihuahua, Weimaraner, and Pug rely on similar representations (high correlations), Schnauzer and Border Collie appear to be encoded differently (low correlations). **D**) Varying the activation of the most relevant unit in B–which strongly affects prediction probabilities (upper left number) of all classes but the Border Collie (bottom row)–reveals human-interpretable features.

change in the classifier's probability of predicting the original class indicated in the upper left corner).

4.5 Discovering the classifier's decision boundaries

In Sec. 4.3 we show that our method can reveal learned representations in single units. However, our method can also be extended to visualize and quantify representations encoded in *distributed* activations in R. Specifically, we analyze the representations that change across a classifier's decision boundary. For an image of a given class, we linearly manipulate the activation $r \in R$ to shift the prediction probability towards a target class (see Eq. 3). We refer to the point in R at which the predicted class changes as the decision boundary. Our proposed method allows to generate human interpretable visualizations of the representations along such counterfactual directions, showing that images at the decision boundary are visually indistinguishable, despite rapid changes in the prediction probability (Fig. 7A, see probability insets). Using our quantification method, we can zoom in on specific concepts (ear, legs, etc.) and pinpoint how their representation changes across the decision boundary (Fig. 7B right). Such level of specificity is not reached in previous models, including GradCAM [58], which may emphasize less interpretable properties, such as features of the background (Fig. 7B left). Along the counterfactual trajectory, common image similarity metrics (MSE, LPIPS) show smooth trajectories across the decision boundary (Fig. 7C left). In contrast, our quantification pipeline draws a more comprehensive picture of the decision-relevant features learned by the classifier (Fig. 7C right). For example, between Pug and Chihuahua, we observe a continuous increase in the area of the ears while the luminance saturates after the decision boundary (Fig. 7C right, top row, pink lines). Between the Border Collie and Chihuahua, the luminance of the head, eyes, and body increases around the decision boundary (Fig. 7C right, bottom row, gray and orange lines). Our method hence offers the opportunity to identify features (ir-)relevant to a classifier's decision.

5 Discussion and Limitations

We introduce a method that leverages the semantic structure in a pre-trained GAN, and thus does not require intensive (re-)training as often suggested by previous work (e.g., [6,9,39]). Our approach offers a computationally inexpensive method for analyzing several different classifiers (trained on similar datasets as the GAN), as training the linking network is fast. Our method even generalizes to models trained on similar but non-overlapping datasets (e.g., different face datasets). Currently, our method does not generalize to ViTs, where regional representations are encoded, as the W-space is not regionally disentangled.

Our analysis pipeline uses a few-shot image segmentation model, allowing us to identify fine-grained features, while minimizing time-intensive image labeling. Note that, we use few-shot segmentation as currently, no large-scale segmentation dataset exists that contains as much feature detail. A network trained with

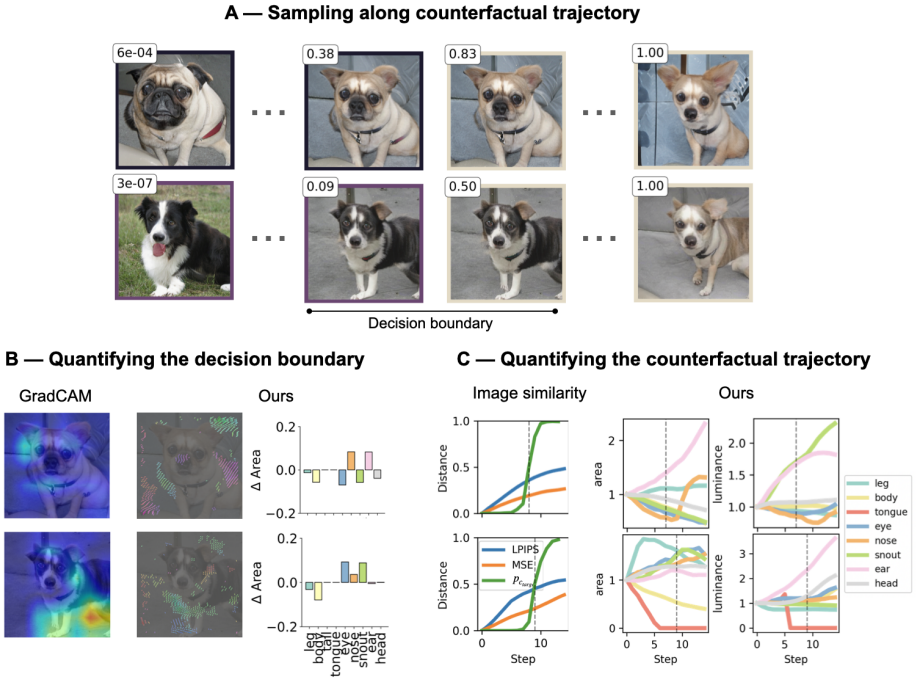

Fig. 7: Counterfactual trajectory between classes reveals the classifier's decision boundary. Given an input image (upper row: Pug, bottom row: Border Collie, in A-C) we manipulate its activation $r \in R$ to produce a counterfactual example of a given target class (upper and bottom row: Chihuahua, in A-C). Based on our linking network, we visualize and quantify the representations along the counterfactual trajectory. **A)** Abrupt transitions in prediction probability at the decision boundary, with changes hardly visible to the human eye. For the two counterfactual examples, we show the original input image (left), the two images around the decision boundary (center), and the final image of the target class (right). The probability of the target class is indicated in the upper left corner. The predicted class for each image is coded by the frame-color. **B)** Quantifying the changes at the decision boundary reveals human-interpretable concept changes with our method (right), which previous methods, such as GradCAM [58] could resolve (left). **C)** Left: Common similarity metrics (MSE, LPIPS) fail to capture the abrupt transition in the classifier's prediction probability ($p_{c_{\text{target}}}$, green). Right: Our method reveals comprehensive trajectories to interpret a classifier's decision boundary across several labels and metrics. The decision boundary is indicated by the vertical dashed line. All metrics along the trajectory are computed using the original image as a reference and normalized.

more data could further improve the segmentation masks. However, averaging across many generated images, as done here, diminishes errors due to imprecise segmentation masks. Note that, despite being trained only on a limited set of classes (e.g, some dog breads), the image segmentation model generalizes to other, similar classes (Supplementary Fig. S6), which drastically reduces the amount of required labeled data.

So far, our analyses use only generated images, as current generative models still cannot fully capture the diversity of real data. However, we assume that many of the features that a GAN can generate are also relevant features encoded in CNN classifiers. Note that despite this limitation, our approach can resolve fine details (e.g., eyes). We expect the development of more accurate generative models in the future will make applications of our approach to real images more robust. Despite not yet being readily applicable to real images, we believe our work introduces a novel type of approach to enable an unprecedented view of the learned features represented in hidden layers of the CNNs.

Our analyses focus on highly similar classes that share general large-scale features (e.g., general body shape) but differ along fine-grained features. It is possible to train the linking network with more (diverse) classes. However, this would shift the emphasis of the revealed features to a more macro-scale level.

In contrast to previous studies, we here analyze all individual dimensions of the representation space and compute comprehensive summary statistics. To the best of our knowledge, no benchmark has been proposed that can quantify abstract representations encoded in single units of CNNs. We believe that our method can open several avenues for studying how the representational geometry (e.g. feature (dis-)entanglement or sparsity) affects a model's performance and its robustness to adversarial attacks. In the future, insights about the complete representational space may inspire model architectures or training strategies that constrain the representational geometry. Additionally, extending our approach to other convolutional layers could reveal insights into features represented at different hierarchies in the classifier. Also, using our method to find difficult-to-classify images, as in [51], appears to be an interesting future direction.

6 Conclusion

We introduce a simple, yet effective tool to visualize and systematically analyze the representations learned in a classifier. Our approach allows us to study abstract concepts encoded in individual units, as well as representations distributed over many units. The proposed automatic quantification pipeline provides a tool to interpret thousands of units simultaneously, demonstrating that even single units can encode meaningful, partially disentangled features carrying relevant class information, and revealing concepts that change across the decision boundary. We believe that our methods can provide novel insights into the representation of abstract concepts in the hidden layers of classifiers and thereby aid the introduction of such models in various real-world applications.

Acknowledgements

The research leading to these results has received funding from the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 414985841 GRK 2566 "iMOL" (MW, POV, and MK), from the German Research Foundation - DFG Research Unit FOR 5368 ARENA (MK), and from SPP 2041 "Computational Connectomics" (POV and MK).

References

1. Alicioglu, G., Sun, B.: A survey of visual analytics for Explainable Artificial Intelligence methods. Computers & Graphics **102**, 502–520 (Feb 2022). https://doi.org/10.1016/j.cag.2021.09.002
2. Amjad, R.A., Liu, K., Geiger, B.C.: Understanding Neural Networks and Individual Neuron Importance via Information-Ordered Cumulative Ablation. IEEE Transactions on Neural Networks and Learning Systems **33**(12), 7842–7852 (Dec 2022). https://doi.org/10.1109/TNNLS.2021.3088685, http://arxiv.org/abs/1804.06679, arXiv:1804.06679 [cs, math, stat]
3. Baek, S., Song, M., Jang, J., Kim, G., Paik, S.B.: Face detection in untrained deep neural networks. Nature Communications **12**(1), 7328 (Dec 2021). https://doi.org/10.1038/s41467-021-27606-9, https://www.nature.com/articles/s41467-021-27606-9, number: 1 Publisher: Nature Publishing Group
4. Bau, A., Belinkov, Y., Sajjad, H., Durrani, N., Dalvi, F., Glass, J.: Identifying and Controlling Important Neurons in Neural Machine Translation (Nov 2018). https://doi.org/10.48550/arXiv.1811.01157, http://arxiv.org/abs/1811.01157, arXiv:1811.01157 [cs]
5. Bau, D., Zhu, J.Y., Strobelt, H., Lapedriza, A., Zhou, B., Torralba, A.: Understanding the role of individual units in a deep neural network. Proceedings of the National Academy of Sciences **117**(48), 30071–30078 (Dec 2020). https://doi.org/10.1073/pnas.1907375117, https://www.pnas.org/doi/abs/10.1073/pnas.1907375117, publisher: Proceedings of the National Academy of Sciences
6. Bordes, F., Balestriero, R., Vincent, P.: High Fidelity Visualization of What Your Self-Supervised Representation Knows About (Aug 2022), http://arxiv.org/abs/2112.09164, arXiv:2112.09164 [cs]
7. Brock, A., Donahue, J., Simonyan, K.: Large Scale GAN Training for High Fidelity Natural Image Synthesis (Feb 2019). https://doi.org/10.48550/arXiv.1809.11096
8. Buckner, C.: Understanding adversarial examples requires a theory of artefacts for deep learning. Nature Machine Intelligence **2**(12), 731–736 (Dec 2020). https://doi.org/10.1038/s42256-020-00266-y, https://www.nature.com/articles/s42256-020-00266-y, number: 12 Publisher: Nature Publishing Group
9. Casanova, A., Careil, M., Verbeek, J., Drozdzal, M., Romero-Soriano, A.: Instance-Conditioned GAN (Nov 2021). https://doi.org/10.48550/arXiv.2109.05070, http://arxiv.org/abs/2109.05070, arXiv:2109.05070 [cs]
10. Chen, X., Fan, H., Girshick, R., He, K.: Improved Baselines with Momentum Contrastive Learning (Mar 2020). https://doi.org/10.48550/arXiv.2003.04297
11. Dalvi, F., Nortonsmith, A., Bau, A., Belinkov, Y., Sajjad, H., Durrani, N., Glass, J.: NeuroX: A Toolkit for Analyzing Individual Neurons in Neural Networks. Proceedings of the AAAI Conference on Artificial Intelligence **33**(01), 9851–9852 (Jul 2019). https://doi.org/10.1609/aaai.v33i01.33019851, https://ojs.aaai.org/index.php/AAAI/article/view/5063, number: 01
12. Dhamdhere, K., Sundararajan, M., Yan, Q.: How Important is a Neuron (Sep 2018), https://openreview.net/forum?id=SylKoo0cKm
13. Donnelly, J., Roegiest, A.: On Interpretability and Feature Representations: An Analysis of the Sentiment Neuron. In: Azzopardi, L., Stein, B., Fuhr, N., Mayr, P., Hauff, C., Hiemstra, D. (eds.) Advances in Information Retrieval. pp. 795–802. Lecture Notes in Computer Science, Springer International Publishing, Cham (2019). https://doi.org/10.1007/978-3-030-15712-8_55

14. Elhage, N., Hume, T., Olsson, C., Schiefer, N., Henighan, T., Kravec, S., Hatfield-Dodds, Z., Lasenby, R., Drain, D., Chen, C., Grosse, R., McCandlish, S., Kaplan, J., Amodei, D., Wattenberg, M., Olah, C.: Toy Models of Superposition
15. Engstrom, L., Ilyas, A., Santurkar, S., Tsipras, D., Tran, B., Madry, A.: Adversarial Robustness as a Prior for Learned Representations (Sep 2019). https://doi.org/10.48550/arXiv.1906.00945, http://arxiv.org/abs/1906.00945, arXiv:1906.00945 [cs, stat]
16. Ghorbani, A., Zou, J.Y.: Neuron Shapley: Discovering the Responsible Neurons. In: Advances in Neural Information Processing Systems. vol. 33, pp. 5922–5932. Curran Associates, Inc. (2020), https://proceedings.neurips.cc/paper_files/paper/2020/hash/41c542dfe6e4fc3deb251d64cf6ed2e4-Abstract.html
17. Gilmer, J., Metz, L., Faghri, F., Schoenholz, S.S., Raghu, M., Wattenberg, M., Goodfellow, I.: Adversarial Spheres (Sep 2018), http://arxiv.org/abs/1801.02774, arXiv:1801.02774 [cs]
18. Goh, G., Cammarata, N., Voss, C., Carter, S., Petrov, M., Schubert, L., Radford, A., Olah, C.: Multimodal Neurons in Artificial Neural Networks, https://distill.pub/2021/multimodal-neurons/
19. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative Adversarial Networks (Jun 2014). https://doi.org/10.48550/arXiv.1406.2661
20. Goyal, A., Bengio, Y.: Inductive biases for deep learning of higher-level cognition. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences **478**(2266), 20210068 (Oct 2022). https://doi.org/10.1098/rspa.2021.0068, https://royalsocietypublishing.org/doi/10.1098/rspa.2021.0068, publisher: Royal Society
21. Goyal, Y., Wu, Z., Ernst, J., Batra, D., Parikh, D., Lee, S.: Counterfactual Visual Explanations. In: Proceedings of the 36th International Conference on Machine Learning. pp. 2376–2384. PMLR (May 2019), https://proceedings.mlr.press/v97/goyal19a.html, iSSN: 2640-3498
22. Greff, K., van Steenkiste, S., Schmidhuber, J.: On the Binding Problem in Artificial Neural Networks (Dec 2020), http://arxiv.org/abs/2012.05208, arXiv:2012.05208 [cs]
23. Guidotti, R.: Counterfactual explanations and how to find them: literature review and benchmarking. Data Mining and Knowledge Discovery (Apr 2022). https://doi.org/10.1007/s10618-022-00831-6, https://doi.org/10.1007/s10618-022-00831-6
24. Hassabis, D., Kumaran, D., Summerfield, C., Botvinick, M.: Neuroscience-Inspired Artificial Intelligence. Neuron **95**(2), 245–258 (Jul 2017). https://doi.org/10.1016/j.neuron.2017.06.011, https://www.cell.com/neuron/abstract/S0896-6273(17)30509-3, publisher: Elsevier
25. He, W., Li, B., Song, D.: DECISION BOUNDARY ANALYSIS OF ADVERSARIAL EXAMPLES (2018)
26. Hou, X., Zhang, X., Liang, H., Shen, L., Lai, Z., Wan, J.: GuidedStyle: Attribute knowledge guided style manipulation for semantic face editing. Neural Networks **145**, 209–220 (Jan 2022). https://doi.org/10.1016/j.neunet.2021.10.017
27. Hoyer, P.O., Hoyer, P.: Non-negative Matrix Factorization with Sparseness Constraints
28. Härkönen, E., Hertzmann, A., Lehtinen, J., Paris, S.: GANSpace: Discovering Interpretable GAN Controls (Dec 2020). https://doi.org/10.48550/arXiv.2004.02546

29. Jahanian, A., Chai, L., Isola, P.: On the "steerability" of generative adversarial networks (Feb 2020). https://doi.org/10.48550/arXiv.1907.07171
30. Joshi, A., Mukherjee, A., Sarkar, S., Hegde, C.: Semantic Adversarial Attacks: Parametric Transformations That Fool Deep Classifiers. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 4772–4782. IEEE, Seoul, Korea (South) (Oct 2019). https://doi.org/10.1109/ICCV.2019.00487, https://ieeexplore.ieee.org/document/9010394/
31. Kang, M., Zhu, J.Y., Zhang, R., Park, J., Shechtman, E., Paris, S., Park, T.: Scaling up GANs for Text-to-Image Synthesis (Jun 2023). https://doi.org/10.48550/arXiv.2303.05511, http://arxiv.org/abs/2303.05511, arXiv:2303.05511 [cs]
32. Karimi, H., Derr, T., Tang, J.: Characterizing the Decision Boundary of Deep Neural Networks (Jun 2020). https://doi.org/10.48550/arXiv.1912.11460, http://arxiv.org/abs/1912.11460, arXiv:1912.11460 [cs, stat]
33. Karimi, H., Tang, J.: Decision Boundary of Deep Neural Networks: Challenges and Opportunities. In: Proceedings of the 13th International Conference on Web Search and Data Mining. pp. 919–920. ACM, Houston TX USA (Jan 2020). https://doi.org/10.1145/3336191.3372186, https://dl.acm.org/doi/10.1145/3336191.3372186
34. Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., Aila, T.: Training Generative Adversarial Networks with Limited Data. In: Advances in Neural Information Processing Systems. vol. 33, pp. 12104–12114. Curran Associates, Inc. (2020)
35. Karras, T., Laine, S., Aila, T.: A Style-Based Generator Architecture for Generative Adversarial Networks (Mar 2019). https://doi.org/10.48550/arXiv.1812.04948
36. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T.: Analyzing and Improving the Image Quality of StyleGAN (Mar 2020). https://doi.org/10.48550/arXiv.1912.04958
37. Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., Sayres, R.: Interpretability Beyond Feature Attribution: Quantitative Testing with Concept Activation Vectors (TCAV). In: Proceedings of the 35th International Conference on Machine Learning. pp. 2668–2677. PMLR (Jul 2018), https://proceedings.mlr.press/v80/kim18d.html, iSSN: 2640-3498
38. Kriegeskorte, N., Mur, M., Bandettini, P.A.: Representational similarity analysis - connecting the branches of systems neuroscience. Frontiers in Systems Neuroscience **2** (Nov 2008). https://doi.org/10.3389/neuro.06.004.2008, https://www.frontiersin.org/journals/systems-neuroscience/articles/10.3389/neuro.06.004.2008/full, publisher: Frontiers
39. Lang, O., Gandelsman, Y., Yarom, M., Wald, Y., Elidan, G., Hassidim, A., Freeman, W.T., Isola, P., Globerson, A., Irani, M., Mosseri, I.: Explaining in Style: Training a GAN to explain a classifier in StyleSpace (Sep 2021). https://doi.org/10.48550/arXiv.2104.13369
40. Leavitt, M.L., Morcos, A.: Selectivity considered harmful: evaluating the causal impact of class selectivity in DNNs (Oct 2020). https://doi.org/10.48550/arXiv.2003.01262, http://arxiv.org/abs/2003.01262, arXiv:2003.01262 [cs, q-bio, stat]
41. Lundstrom, D.D., Huang, T., Razaviyayn, M.: A Rigorous Study of Integrated Gradients Method and Extensions to Internal Neuron Attributions. In: Proceedings of the 39th International Conference on Machine Learning. pp. 14485–14508.

PMLR (Jun 2022), https://proceedings.mlr.press/v162/lundstrom22a.html, iSSN: 2640-3498

42. Luo, J., Wang, Z., Wu, C.H., Huang, D., De La Torre, F.: Zero-Shot Model Diagnosis. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11631–11640. IEEE, Vancouver, BC, Canada (Jun 2023). https://doi.org/10.1109/CVPR52729.2023.01119, https://ieeexplore.ieee.org/document/10204233/
43. Mahendran, A., Vedaldi, A.: Understanding Deep Image Representations by Inverting Them (Nov 2014). https://doi.org/10.48550/arXiv.1412.0035, http://arxiv.org/abs/1412.0035, arXiv:1412.0035 [cs]
44. Manerikar, A., Kak, A.C.: Self-Supervised One-Shot Learning for Automatic Segmentation of StyleGAN Images (Oct 2023). https://doi.org/10.48550/arXiv.2303.05639, http://arxiv.org/abs/2303.05639, arXiv:2303.05639 [cs]
45. Morcos, A.S., Barrett, D.G.T., Rabinowitz, N.C., Botvinick, M.: On the importance of single directions for generalization (Feb 2018), https://openreview.net/forum?id=r1iuQjxCZ
46. Mothilal, R.K., Sharma, A., Tan, C.: Explaining machine learning classifiers through diverse counterfactual explanations. In: Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency. pp. 607–617. FAT* '20, Association for Computing Machinery, New York, NY, USA (Jan 2020). https://doi.org/10.1145/3351095.3372850, https://dl.acm.org/doi/10.1145/3351095.3372850
47. Mu, J., Andreas, J.: Compositional Explanations of Neurons. In: Advances in Neural Information Processing Systems. vol. 33, pp. 17153–17163. Curran Associates, Inc. (2020), https://proceedings.neurips.cc/paper/2020/hash/c74956ffb38ba48ed6ce977af6727275-Abstract.html
48. Oakden-Rayner, L., Dunnmon, J., Carneiro, G., Re, C.: Hidden stratification causes clinically meaningful failures in machine learning for medical imaging. In: Proceedings of the ACM Conference on Health, Inference, and Learning. pp. 151–159. CHIL '20, Association for Computing Machinery, New York, NY, USA (Apr 2020). https://doi.org/10.1145/3368555.3384468, https://dl.acm.org/doi/10.1145/3368555.3384468
49. Pakhomov, D., Hira, S., Wagle, N., Green, K.E., Navab, N.: Segmentation in Style: Unsupervised Semantic Image Segmentation with Stylegan and CLIP (Nov 2021). https://doi.org/10.48550/arXiv.2107.12518, http://arxiv.org/abs/2107.12518, arXiv:2107.12518 [cs]
50. Plumerault, A., Borgne, H.L., Hudelot, C.: Controlling generative models with continuous factors of variations (Jan 2020). https://doi.org/10.48550/arXiv.2001.10238
51. Prabhu, V., Yenamandra, S., Chattopadhyay, P., Hoffman, J.: LANCE: Stress-testing Visual Models by Generating Language-guided Counterfactual Images (Oct 2023). https://doi.org/10.48550/arXiv.2305.19164, http://arxiv.org/abs/2305.19164, arXiv:2305.19164 [cs]
52. Qiu, H., Xiao, C., Yang, L., Yan, X., Lee, H., Li, B.: SemanticAdv: Generating Adversarial Examples via Attribute-Conditioned Image Editing. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M. (eds.) Computer Vision – ECCV 2020, vol. 12359, pp. 19–37. Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-58568-6_2, https://link.springer.com/10.1007/978-3-030-58568-6_2, series Title: Lecture Notes in Computer Science

53. Radford, A., Jozefowicz, R., Sutskever, I.: Learning to Generate Reviews and Discovering Sentiment (Apr 2017). https://doi.org/10.48550/arXiv.1704.01444, http://arxiv.org/abs/1704.01444, arXiv:1704.01444 [cs]
54. Revaud, J., Leroy, V., Weinzaepfel, P., Chidlovskii, B.: PUMP: Pyramidal and Uniqueness Matching Priors for Unsupervised Learning of Local Descriptors. pp. 3926–3936 (2022)
55. Ribeiro, M.T., Singh, S., Guestrin, C.: "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 1135–1144. KDD '16, Association for Computing Machinery, New York, NY, USA (Aug 2016). https://doi.org/10.1145/2939672.2939778
56. Räuker, T., Ho, A., Casper, S., Hadfield-Menell, D.: Toward Transparent AI: A Survey on Interpreting the Inner Structures of Deep Neural Networks (Aug 2023), http://arxiv.org/abs/2207.13243, arXiv:2207.13243 [cs]
57. Sauer, A., Schwarz, K., Geiger, A.: StyleGAN-XL: Scaling StyleGAN to Large Diverse Datasets (May 2022)
58. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization. International Journal of Computer Vision **128**(2), 336–359 (Feb 2020). https://doi.org/10.1007/s11263-019-01228-7
59. Shen, Y., Gu, J., Tang, X., Zhou, B.: Interpreting the Latent Space of GANs for Semantic Face Editing. pp. 9243–9252 (2020)
60. Shen, Y., Zhou, B.: Closed-Form Factorization of Latent Semantics in GANs. pp. 1532–1540 (2021)
61. Shrikumar, A., Greenside, P., Kundaje, A.: Learning Important Features Through Propagating Activation Differences (Oct 2019). https://doi.org/10.48550/arXiv.1704.02685
62. Singla, S., Eslami, M., Pollack, B., Wallace, S., Batmanghelich, K.: Explaining the Black-box Smoothly- A Counterfactual Approach (Nov 2022)
63. Somepalli, G., Fowl, L., Bansal, A., Yeh-Chiang, P., Dar, Y., Baraniuk, R., Goldblum, M., Goldstein, T.: Can Neural Nets Learn the Same Model Twice? Investigating Reproducibility and Double Descent From the Decision Boundary Perspective. pp. 13699–13708 (2022), https://openaccess.thecvf.com/content/CVPR2022/html/Somepalli_Can_Neural_Nets_Learn_the_Same_Model_Twice_Investigating_Reproducibility_CVPR_2022_paper.html
64. Tritrong, N., Rewatbowornwong, P., Suwajanakorn, S.: Repurposing GANs for One-shot Semantic Part Segmentation (Jul 2021). https://doi.org/10.48550/arXiv.2103.04379, http://arxiv.org/abs/2103.04379, arXiv:2103.04379 [cs]
65. Varoquaux, G., Cheplygina, V.: Machine learning for medical imaging: methodological failures and recommendations for the future. npj Digital Medicine **5**(1), 1–8 (Apr 2022). https://doi.org/10.1038/s41746-022-00592-y, https://www.nature.com/articles/s41746-022-00592-y, number: 1 Publisher: Nature Publishing Group
66. Voynov, A., Babenko, A.: Unsupervised Discovery of Interpretable Directions in the GAN Latent Space (Jun 2020). https://doi.org/10.48550/arXiv.2002.03754
67. Xu, J., Zhang, Z., Hu, X.: Extracting Semantic Knowledge from GANs with Unsupervised Learning (Nov 2022), http://arxiv.org/abs/2211.16710, arXiv:2211.16710 [cs]
68. Yuan, X., He, P., Zhu, Q., Li, X.: Adversarial Examples: Attacks and Defenses for Deep Learning. IEEE Transactions on Neural Networks and Learning Systems **30**(9), 2805–2824 (Sep 2019). https :

//doi.org/10.1109/TNNLS.2018.2886017, https://ieeexplore.ieee.org/abstract/document/8611298?casa_token=XxG23qXGxMIAAAAA:f9C_FUajUaMucwgNgXAcO90ZuRn9sbUR3f5JZurAxz1PH16tpf4B4cAS5vK-UhFBCj2rUuGUHiQ, conference Name: IEEE Transactions on Neural Networks and Learning Systems

69. Zeiler, M.D., Fergus, R.: Visualizing and Understanding Convolutional Networks. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) Computer Vision – ECCV 2014. pp. 818–833. Lecture Notes in Computer Science, Springer International Publishing, Cham (2014). https://doi.org/10.1007/978-3-319-10590-1_53
70. Zhang, J., Li, C.: Adversarial Examples: Opportunities and Challenges. IEEE Transactions on Neural Networks and Learning Systems **31**(7), 2578–2593 (Jul 2020). https://doi.org/10.1109/TNNLS.2019.2933524, https://ieeexplore.ieee.org/abstract/document/8842604?casa_token=xXCxlCV3SHcAAAAA:cRxHQtRQaAzxdPVn8Q2EzdOM8NhOLZ7QqMEcipBHNOfT4jdgh2pdSA_HMJKiqN4lzIQ-uPrA5Mc, conference Name: IEEE Transactions on Neural Networks and Learning Systems
71. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Object Detectors Emerge in Deep Scene CNNs (Apr 2015). https://doi.org/10.48550/arXiv.1412.6856, http://arxiv.org/abs/1412.6856, arXiv:1412.6856 [cs]