Informative Dropout for Robust Representation Learning: A Shape-bias Perspective

Baifeng Shi [*1] **Dinghuai Zhang** [*2] **Qi Dai** [3] **Zhanxing Zhu** [2 4 5] **Yadong Mu** [6] **Jingdong Wang** [3]

Abstract

Convolutional Neural Networks (CNNs) are known to rely more on local texture rather than global shape when making decisions. Recent work also indicates a close relationship between CNN's texture-bias and its robustness against distribution shift, adversarial perturbation, random corruption, *etc*. In this work, we attempt at improving various kinds of robustness universally by alleviating CNN's texture bias. With inspiration from the human visual system, we propose a light-weight model-agnostic method, namely Informative Dropout (InfoDrop), to improve interpretability and reduce texture bias. Specifically, we discriminate texture from shape based on local self-information in an image, and adopt a Dropout-like algorithm to decorrelate the model output from the local texture. Through extensive experiments, we observe enhanced robustness under various scenarios (domain generalization, few-shot classification, image corruption, and adversarial perturbation). To the best of our knowledge, this work is one of the earliest attempts to improve different kinds of robustness in a unified model, shedding new light on the relationship between shape-bias and robustness, also on new approaches to trustworthy machine learning algorithms. Code is available at https://github.com/bfshi/InfoDrop.

1. Introduction

Despite the impressive performance in a broad range of visual tasks, Convolutional Neural Network (CNN) is surprisingly vulnerable compared with the human visual system. For example, features learned by CNN have trouble in generalizing across shifted distributions between training and test data (Chen et al., 2019; Wang et al., 2019a). Random image corruptions can also considerably degrade its performance (Hendrycks & Dietterich, 2019). CNN is extremely defenseless under well-designed image perturbation as well (Szegedy et al., 2013). This is opposite to the human visual system, which is robust to domain gap, noisy input, *etc*. (Biederman, 1987; Bisanz et al., 2012; Geirhos et al., 2017).

(a) Original image (b) Edge detection (c) Information-guided

Figure 1. Comparison of different shape-biased methods. (a) Original image of cat and dog. (b) Simple edge detection is susceptible to complex patterns (*e.g*. stripes of the cat) and can severely damage image contents. (c) In this work, we reduce texture-bias under guidance of self-information, which aligns well with human vision. The definition and computation of self-information are in Sec. 3.

Another intriguing property of CNN is its 'texture bias', namely its bias towards texture instead of shape. Despite the earlier belief that CNN extracts more abstract shapes and structures layer by layer as human does (Kriegeskorte, 2015; LeCun et al., 2015), recent works reveal its reliance on the local texture when making decisions (Jo & Bengio, 2017; Geirhos et al., 2019; Brendel & Bethge, 2019). For instance, given an image with a cat's shape filled with an elephant's skin texture, CNN tends to classify it as an elephant instead of a cat (Geirhos et al., 2019).

Supported by some recent works, there seems to be a surprisingly close relationship between CNN's robustness and texture-bias. For example, Zhang & Zhu (2019) find that adversarially trained CNNs are innately less texture-biased. There are also a few attempts to tackle a specific task by training a less texture-biased model. Carlucci et al. (2019) propose to improve robustness against domain gap by training on jigsaw puzzles, which relies more on global structure information. Geirhos et al. (2019) find that shape-biased

[*]Equal contribution [1]School of EECS, Peking University, China [2]School of Mathematical Sciences, Peking University, China [3]Microsoft Research Asia [4]Center for Data Science, Peking University [5]Beijing Institute of Big Data Research [6]Wangxuan Institute of Computer Technology, Peking University. Correspondence to: Baifeng Shi <bfshi@pku.edu.cn>.

Proceedings of the 37^{th} International Conference on Machine Learning, Vienna, Austria, PMLR 119, 2020.

CNNs trained on stylized images are more robust to random image distortions. Up to this point, one may naturally wonder:

Is texture-bias a common reason for CNN's different kinds of non-robustness against distribution shift, adversarial perturbation, image corruption, etc.?

To explore the answer, this work aims at improving various kinds of robustness universally by alleviating CNN's texture bias and enhancing shape-bias. Some approaches to train shape-biased CNNs have been proposed recently. However, they either are susceptible to complex patterns (see Fig. 1(b)) (Radenovic et al., 2018), or have high computational complexity and auxiliary tasks (Geirhos et al., 2019; Wang et al., 2019a; Carlucci et al., 2019; Wang et al., 2019b). In this work, we propose a light-weight model-agnostic method, namely Informative Dropout (**InfoDrop**). The inspiration comes from earlier works on saliency detection and human eye movements: humans tend to look at regions with high self-information $-\log \mathbb{P}(\text{current region} \mid \text{surrounding regions})$, *i.e.*, regions whose being observed based on surrounding regions contains more 'surprise' (Bruce & Tsotsos, 2006; 2009). In other words, people tend to pay more attention to regions that look different from neighboring regions. In our case, patterns like flat regions or high-frequency textures tend to repeat themselves in the neighboring region, thus being less informative. On the other hand, visual primitives (*e.g.* edges, corners) are more unique and thus more informative among its neighborhood. Fig. 1(c) provides a visualization of the information distribution in natural images. Note that both shape and important characteristics (*e.g.* eyes, stripes) are accentuated, while texture (*e.g.* hair) is relatively repressed.

To this end, InfoDrop is proposed to reduce texture-bias by decorrelating each layer's output with less informative input regions. Specifically, we adopt a Dropout-like algorithm (Srivastava et al., 2014): for input regions with less information, we zero out the corresponding output neurons with higher probability. In this way, reliance on textures can be reduced and the model is trained to be more biased towards shapes. By eliminating InfoDrop after training, the model is further demonstrated to be internally shape-biased without InfoDrop during inference. The shape-bias property is exhibited through different experiments, both qualitatively and quantitatively.

To evaluate the robustness of InfoDrop, we conduct extensive experiments in four different tasks: domain generalization, few-shot classification, robustness against random corruption and adversarial robustness. Results show a consistent improvement in different kinds of robustness over various baselines, demonstrating the effectiveness and versatility of our method. We also demonstrate that InfoDrop can be combined with other algorithms (*e.g.* adversarial training) to further enhance the robustness.

1.1. Our Contribution

- With inspiration from the human visual system, we propose InfoDrop, an effective albeit simple plug-in method to reduce the general texture bias of any CNN-based model.
- As shown by extensive experiments, InfoDrop achieves consistently non-trivial improvement over multiple baselines in a wide variety of robustness settings. Furthermore, InfoDrop can be incorporated together with other algorithms to obtain higher robustness.
- To the best of our knowledge, this work is one of the earliest attempts to improve different kinds of robustness in a unified model. This sheds new light on the relationship between CNN's texture-bias and non-robustness, also on new approaches to building trustworthy machine learning algorithms.

2. Related Work

2.1. Vulnerability of CNNs

An important feature of intelligence is its ability to generalize knowledge across tasks, domains and categories (Csurka, 2017). However, CNNs still struggle when different kinds of distribution shifts exist between training and test data. For instance, in few-shot classification, where large class gap is the main challenge, complex algorithms make little improvement upon simple baselines (Chen et al., 2019; Huang et al., 2020; Dhillon et al., 2020). CNNs also have trouble with transferring knowledge across different domains, especially when data is unavailable in the target domain as in the task of domain generalization (Khosla et al., 2012; Li et al., 2017; 2018; Carlucci et al., 2019). In this work, we evaluate our method's robustness against distribution shift on tasks of few-shot classification and domain generalization.

CNNs are also sensitive to small perturbations and corruptions in images, which can be easily dealt with by humans (Azulay & Weiss, 2019). Hendrycks & Dietterich (2019) benchmark CNN's robustness against 18 types of random corruption, demonstrating its vulnerability. It is also shown that well-designed perturbation, namely adversarial perturbation, can severely degrade the performance of CNNs (Szegedy et al., 2013). We evaluate the robustness of our approach against both random corruption and adversarial perturbation, with other methods towards model robustness as baseline, *e.g.*, adversarial training (Madry et al., 2018; Zhang et al., 2019).

2.2. Texture-bias of CNNs

Despite the recent impressive performance of CNNs in various vision tasks, the visual processing mechanism behind

remains controversial. One widely accepted hypothesis is that CNNs extract low-level primitives (*e.g.* edges, corners) in lower layers and try to combine them into complex shapes in higher layers (Kriegeskorte, 2015; LeCun et al., 2015). This hypothesis is supported by numbers of empirical findings, both from computational (Zeiler & Fergus, 2014) and psychological (Ritter et al., 2017) perspectives. However, recent work argues that local texture is sufficient for CNNs to perform correct classification (Brendel & Bethge, 2019). Shape or contour information, on the other hand, seems hard for CNNs to understand (Ballester & Araujo, 2016). CNNs also fail at transferring between images with similar shapes yet distinct textures (Geirhos et al., 2019). These findings indicate an alternative explanation for the success of CNNs: local texture is what CNNs base on when making decisions.

2.3. Relation between Non-robustness and Texture-bias

More and more work indicates a close relationship between CNNs' non-robustness and texture-bias. Zhang & Zhu (2019) find that adversarially trained networks are less texture-biased. Geirhos et al. (2019) show that shape-biased models trained with stylized images are more robust against image distortion. Carlucci et al. (2019) propose to boost domain generalization by training to solve jigsaw puzzles, which relies more on global structure. Wang et al. (2019a) propose to penalize CNN's local predictive power to reduce the domain gap induced by image background. With the same objective, Wang et al. (2019b) propose to project out superficial statistics in feature space. However, none of the work has discussed the relationship between texture-bias and different types of non-robustness in a unified model.

2.4. Bias in Human Vision

It is known that human eyes tend to fixate on specific regions (saliency) rather than scan the whole image they see (Yarbus, 2013). The mechanism behind this kind of bias has attracted lots of interest. Itti et al. (1998) reveal the importance of center-surround contrast of units in the human visual system. Hou & Zhang (2007) detect saliency using residual contrast in the spectral domain. Other works propose to use Shannon entropy to measure saliency and predict fixation (Fritz et al., 2004; Renninger et al., 2005). In Bruce & Tsotsos (2006), self-information is proposed to better model saliency.

In addition, shape-bias is also found critical in the human visual system. A large amount of evidence shows shape is the most important single clue for human vision learning and processing (Landau et al., 1988). For example, young children tend to extend object names based on its shape, rather than size, color or material (Diesendruck & Bloom, 2003). The shape bias of human vision, together with its bias towards self-information, further motivates our proposed method.

3. Methodology

Let $\mathbf{x} \in \mathbb{R}^{c_0 \times h_0 \times w_0}$ denotes an image with c_0 channels and spatial shape of $h_0 \times w_0$. For a CNN, we denote the input of l-th convolutional layer by $\mathbf{z}^{\ell-1} \in \mathbb{R}^{c_{\ell-1} \times h_{\ell-1} \times w_{\ell-1}}$ and output by $\mathbf{z}^{\ell} \in \mathbb{R}^{c_{\ell} \times h_{\ell} \times w_{\ell}}$. Note that $\mathbf{z}^0$ equals to the input image $\mathbf{x}$. Assume the l-th layer has a convolutional kernel $\mathbf{k}^{\ell} \in \mathbb{R}^{c_{\ell} \times c_{\ell-1} \times k \times k}$ and bias $\mathbf{b}^{\ell} \in \mathbb{R}^{c_{\ell}}$, where k is the kernel size. Then for c-th channel's j-th element $z^{\ell}_{c,j}$ in output $\mathbf{z}^{\ell}$ ($j \in \{1, 2, ..., h_{\ell} w_{\ell}\}$), we have $z^{\ell}_{c,j} = \sigma(\mathbf{k}^{\ell}_c \cdot \mathbf{p}^{\ell-1}_j + b^{\ell}_c)$, where $\mathbf{p}^{\ell-1}_j \in \mathbb{R}^{c_{\ell-1} \times k \times k}$ is the j-th patch in $\mathbf{z}^{\ell-1}$, $\mathbf{k}^{\ell}_c$ and b^{ℓ}_c are the kernel and bias for c-th output channel, $\cdot$ indicates inner product and $\sigma(\cdot)$ is an entry-wise activation function (*e.g.* ReLU). All through this paper $\|\cdot\|$ denotes Euclidean norm.

3.1. Informative Dropout

Now we develop our information-based Dropout method for alleviating texture-bias. As discussed in Section 1, regions of textures tend to contain low self-information. To this end, we propose to reduce texture-bias by decorrelating each layer's output with low-information regions in input. Specifically, we adopt a Dropout-like approach for the purpose. In traditional Dropout (Srivastava et al., 2014), a multiplicative Bernoulli noise is introduced to help prevent overfitting, where each neuron is zeroed out with equal probability. In order to suppress texture-bias, we propose to zero out an output neuron with higher probability if the input patch contains less information, and vice versa. Specifically, we model the drop coefficient r of the j-th neuron in output's c-th channel with a Boltzmann distribution:

$$r(z^{\ell}_{c,j}) \propto e^{-\mathcal{I}(\mathbf{p}^{\ell-1}_j)/T}, \tag{1}$$

where $\mathbf{p}^{\ell-1}_j$ is the patch in the input related to the computation of $z^{\ell}_{c,j}$, $\mathcal{I}$ denotes self-information and T is temperature. When value of $\mathcal{I}$ is low, the corresponding neuron is likely to be dropped, and the network tends to rely less on $\mathbf{p}^{\ell-1}_j$. Here the temperature T serves as a 'soft threshold' of information. When T is small, the threshold lowers down, and only patches with least information (*e.g.* a patch in a solid-colored region) will be dropped. When T goes to infinity, all neuron will be dropped with equal probability, and the whole algorithm becomes regular Dropout.

First we discuss how to estimate $\mathcal{I}$. The definition of information could date back to Shannon's work (Shannon, 1948), from where we borrow the concept of self-information $\mathcal{I}$ to describe the information of a patch:

$$\mathcal{I}(\mathbf{p}^{\ell-1}_j) = -\log q^{\ell-1}_j(\mathbf{p}^{\ell-1}_j), \tag{2}$$

where $q^{\ell-1}_j$ is the distribution which $\mathbf{p}^{\ell-1}_j$ is sampled from,

if we see $\mathbf{p}_j^{\ell-1}$ as a realization of a random variable. As a simple case, we can assume that all patches in the neighborhood of $\mathbf{p}_j^{\ell-1}$ are different realizations of the same random variable, *i.e.*, they are all sampled from the same distribution $q_j^{\ell-1}$. In this case, if $\mathbf{p}_j^{\ell-1}$ contains more "texture" than "shape", its pattern shall repeat itself within a local region, resulting in a high likelihood $q_j^{\ell-1}(\mathbf{p}_j^{\ell-1})$ and hence low self-information and should be zeroed out with high probability.

To approximate $q_j^{\ell-1}(\cdot)$, we assume that $\mathbf{p}_j^{\ell-1}$ and other patches in its neighbourhood $\mathcal{N}_j^{\ell-1}$ come from the same distribution $\mathbf{p} \sim q_j^{\ell-1}(\mathbf{p})$. Here the neighbourhood means a local region centered at $\mathbf{p}_j^{\ell-1}$, with Manhattan radius R, *i.e.*, the neighborhood contains $(2R+1)^2$ patches. Then, with neighboring patches as samples, we approximate $q_j^{\ell-1}(\cdot)$ with its kernel density estimator $\hat{q}_j^{\ell-1}$, *i.e.*

$$\hat{q}_j^{\ell-1}(\mathbf{p}) = \frac{1}{(2R+1)^2} \sum_{\mathbf{p}' \in \mathcal{N}_j^{\ell-1}} K(\mathbf{p}, \mathbf{p}'), \tag{3}$$

where $K(\cdot,\cdot)$ is kernel function. Here we use Gaussian kernel, *i.e.*, $K(\mathbf{p},\mathbf{p}') = \frac{1}{\sqrt{2\pi}h} \exp(-||\mathbf{p}-\mathbf{p}'||^2/2h^2)$, where h is the bandwidth. Then information of $\mathbf{p}_j^{\ell-1}$ is estimated by

$$\hat{\mathcal{I}}(\mathbf{p}_j^{\ell-1}) = -\log\{ \sum_{\mathbf{p}' \in \mathcal{N}_j^{\ell-1}} e^{-||\mathbf{p}_j^{\ell-1}-\mathbf{p}'||^2/2h^2} \} + \text{const.} \tag{4}$$

As one can observe, the more different $\mathbf{p}_j^{\ell-1}$ is from neighbouring patches, the more information it contains. For regions of solid color or high-frequency texture, similar patterns tend to repeat in the neighborhood, and thus little information is contained. Local shapes, on the other hand, are more unique in their surroundings and thus more informative.

Then we discuss how the dropout process works. A direct way is to sample neurons in the output $\mathbf{z}^\ell$ with probabilities given by Eq. 1, and set them to zero. During training, for the c-th channel of ℓ-th layer's output $\mathbf{z}_c^\ell \in \mathbb{R}^{h_\ell \times w_\ell}$, we randomly choose neurons to drop by running weighted multinomial sampling with replacement for $r_0 \cdot h_\ell \cdot w_\ell$ times,[1] where r_0 is a hyper-parameter controlling the amount of dropped neurons. The algorithm is shown in Alg. 1.

Note that when training with InfoDrop on, we are *intentionally* filtering out texture to make the model learn to recognize shape. However, during inference, we expect to see a genuinely shape-biased model which can filter out texture by itself *without* InfoDrop's help. To check if our model has obtained this "internal" shape-bias, one way is to directly remove the InfoDrop blocks during inference. However, there may be statistical mismatch (*e.g.* in batch normalization) between clean images and images processed by InfoDrop. To this end, we take the inspiration from (Geirhos et al., 2019) and propose to **finetune the network on clean images with InfoDrop removed**, as an extra step after training with InfoDrop. In this way, we can safely remove InfoDrop during testing, and examine if our network has truly learned shape-bias.

[1] Here we choose sampling with replacement over without replacement because the former runs faster in practice. Hence here r_0 can be any positive real number due to collision of samples, and the actual dropout rate (expected ratio of sampled neurons) will be lower than r_0.

Algorithm 1 Informative Dropout (InfoDrop)

Input: input activation map $\mathbf{z}^{\ell-1}$
Parameters: convolutional kernel $\mathbf{k}^\ell$, bias $\mathbf{b}^\ell$, radius R, temperature T, bandwidth h, "dropout rate" r_0
Output: output activation map $\mathbf{z}^\ell$

for each element $z_{c,j}^\ell$ in output **do**
  $z_{c,j}^\ell \leftarrow \sigma(\mathbf{k}_c^\ell \cdot \mathbf{p}_j^{\ell-1} + b_c^\ell)$
end for
for $c = 1$ **to** c_ℓ **do**
  for $i = 1$ **to** $\lfloor r_0 \cdot h_\ell \cdot w_\ell \rfloor$ **do**
    sample j from $[1, h_\ell \cdot w_\ell]$ with probability $r(z_{c,j}^\ell)$ given by Eq. 1
    $z_{c,j}^\ell \leftarrow 0$
  end for
end for

3.2. Computational Complexity

There are two parts of computational cost in InfoDrop: (i) calculation of self-information of input patches, and (ii) manipulation of each output element. For self-information calculation, there are $\mathcal{O}(h_{\ell-1} \cdot w_{\ell-1})$ input patches, each with size $\mathcal{O}(c_{\ell-1})$. Note that kernel size and scale of neighborhood are constants. This means a time complexity of $\mathcal{O}(c_{\ell-1} \cdot h_{\ell-1} \cdot w_{\ell-1})$ for part (i). As for part (ii), both sampling and element-wise product needs $\mathcal{O}(c_\ell \cdot h_\ell \cdot w_\ell)$. Note that spatial shape often stays unchanged through convolution. Therefore, time complexity of InfoDrop is $\mathcal{O}((c_{\ell-1} + c_\ell) \cdot h_\ell \cdot w_\ell)$, which is little overhead compared with $\mathcal{O}(c_{\ell-1} \cdot c_\ell \cdot h_\ell \cdot w_\ell)$ in convolutional operation.

4. Experiments

We conduct extensive experiments for further understanding properties of InfoDrop and its benefits over standard CNN-based models. First we discuss the shape-bias property of InfoDrop in Sec. 4.1. Then in Sec. 4.2 we evaluate robustness of InfoDrop through four different tasks, *viz.* domain generalization, few-shot classification, robustness against random corruption and adversarial robustness, and also com-

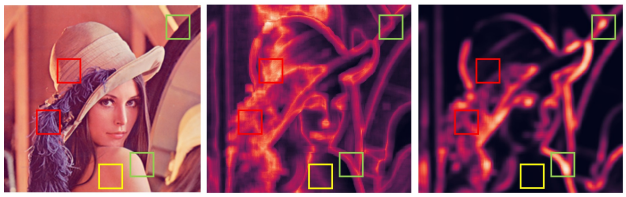

Figure 2. Picture of Lenna, its frequency map and self-information map. Lighter regions indicate higher values.

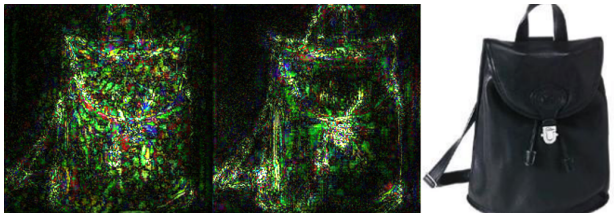

Figure 3. Gradient-based saliency maps of regular CNN (left) and CNN with InfoDrop (middle). Input image is shown on the right.

pare with other shape-biased approaches. In Sec. 4.3, we conduct ablations for further analysis. The balance between shape and texture is discussed in Sec. 4.4. Please refer to Appendix for specific experimental settings.

4.1. Shape-bias of InfoDrop

We conduct several experiments, both qualitatively and quantitatively, to analyze the shape-bias property of InfoDrop. Due to limited space, we refer readers to Appendix for more visualization and detailed experimental settings.

A Frequency Perspective We first analyze the shape-bias property of self-information by visualizing how it responds to local regions with different spatial frequency. To obtain the average frequency of a local region, we apply Discrete Cosine Transform (DCT) (Ahmed et al., 1974) to the local 8×8 patch to get the power spectrum, which is further used as weights of each frequency level to get the average frequency. We repeat the process for each position and get the frequency map (Fig. 2(b)). We also calculate each position's self-information (Fig. 2(c)). As one can observe, for visual primitives including edges and corners (**green boxes**), they present medium frequency, but are most highlighted by self-information. High-frequency textures (**red boxes**), as highlighted in frequency map, however, contain relatively low information due to its high-frequency self-repeating. Flat regions (**yellow boxes**) are filtered by both frequency and information map. This is also consistent with our previous discussions.

Saliency Map of CNN To verify the shape-bias InfoDrop brings to CNNs, we visualize gradients of model output w.r.t. input pixels, which serve as a "saliency map" of the network. Specifically we use SmoothGrad (Smilkov et al., 2017) to calculate saliency map $S(x)$,

$$S(x) = \frac{1}{n} \sum_{i=1}^{n} \frac{\partial f(x_i)}{\partial x_i}, \tag{5}$$

where $x_i = x + \delta_i$ is original image x with i.i.d. Gaussian noise δ_i, and $f(\cdot)$ is the network. An example is shown in Fig. 3. We can see that InfoDrop is more human-aligned, sensitive to shapes of objects, while the saliency map of regular CNN is more noisy and less shape-biased, lacking interpretability.

Table 1. Degradation of classification accuracy on patch-shuffled images. Each image is divided into $m \times m$ patches. Here we use $m = 1$ as baseline, referring to accuracy on original images.

m	1	2	3	4
REGULAR CNN	99.88	99.16	97.60	92.99
W/ INFODROP	99.80	95.37	89.03	79.90

Patch Shuffling We also evaluate the shape-bias of InfoDrop through recognizing images whose shape information is ruined but texture is retained. Following (Zhang & Zhu, 2019), we achieve this goal by dividing images into $m \times m$ patches and randomly shuffling them. Through patch shuffling, global structure is ruined while local texture in each patch is left untouched. We train our model on clean images and test on patch-shuffled test set. We set different values of m and results are listed in Table 1. Note that $m = 1$ means no shuffling is used. As m goes up, global structures are severely ruined, causing a rapid declination in InfoDrop's performance. However, regular CNN is barely influenced since most texture information is preserved. This also indicates that CNN with InfoDrop is more biased towards shape information.

Style Transfer To understand the features extracted by InfoDrop, we conduct ablations in the task of style transfer. Recently, Huang & Belongie (2017) proposed AdaIN algorithm to render a content image with the style of another image (style image). Specifically, features of both content and style images are first extracted by encoder, and then the mean and variance of the content feature is aligned with those of the style feature. Transferred image is then decoded from the aligned content feature. In our experiment, we apply InfoDrop in the encoder and observe changes in the rendered image. By doing so, we expect to see that only the edging style of the content image is rendered by that of the style image, and the texture style is preserved. This is verified by the results in Fig. 4. Take the first row as example, we can see that baseline method mainly change the tone of the whole image. In contrast, InfoDrop inherits the style of red edging and sketching, and applies it on the

Figure 4. Results of style transfer. From left to right: content image, style image, baseline result, result of InfoDrop. For instance, in baseline result of the last row, both shape (*e.g.* edging) and texture (*e.g.* coloring) style are inherited from style image. However, InfoDrop mainly renders edges in content image, while texture (*e.g.* sky) or color tone is less affected.

shape of content image, indicating that InfoDrop is more shape-biased in both content and style images.

4.2. Robustness of InfoDrop

In this section, we first evaluate various kinds of robustness (against distribution shift, image corruption and adversarial perturbation) of InfoDrop through four different tasks (Sec. 4.2.1 $\sim$ Sec. 4.2.4). Since InfoDrop can be applied to any CNN-based models, and extensive exploration of more complicated base models is beyond the main scope of our studies in this section, we only use simple architecture (*e.g.* ResNet (He et al., 2016)) and baseline algorithms, and observe incremental results when InfoDrop is applied. Then we compare InfoDrop with other approaches towards shape-bias (Sec. 4.2.5). Due to limited space, detailed experimental configuration and additional results are deferred to Appendix.

4.2.1. Domain Generalization

Due to the natural data variance induced by time, location, weather, *etc*., it's a significant feature for visual models to generalize across different domains. To this end, the task of *domain adaptation* is proposed, where labeled data from source domain and unlabeled data from target domain are provided (Shimodaira, 2000). Prior arts mainly focus on diminishing the distribution shift in feature space between source and target domain (Gretton et al., 2007; 2009; Long et al., 2015). A more challenging task, namely *domain generalization*, is later proposed, where data from target domain is unavailable during training. Previous solutions include learning invariant features (Muandet et al., 2013), or utilizing auxiliary tasks (Carlucci et al., 2019).

Table 2. Incremental results of single-source domain generalization. + (-) indicates performance gain (decline) from InfoDrop.

Source \ Target	Photo	Art	Cartoon	Sketch
Photo	*-0.06*	**+2.49**	**+6.52**	**+14.76**
Art	**+0.12**	*+0.20*	**+1.57**	**+0.81**
Cartoon	-0.84	-0.44	*+0.04*	**+4.81**
Sketch	**+11.91**	**+4.23**	**+6.19**	*+0.15*

Table 3. Results on multi-source domain generalization. Performance of JiGen (Carlucci et al., 2019) and D-SAM (D'Innocente & Caputo, 2018) are listed for comparison.

Methods \ Target	Photo	Art	Cartoon	Sketch
D-SAM	95.30	77.33	72.43	**77.83**
JiGen	96.03	79.42	75.25	71.35
Baseline	95.98	77.87	74.86	70.17
+ InfoDrop	**96.11**	**80.27**	**76.54**	76.38

In our experiment, we use the naive algorithm as baseline: training a classification model on source domain, and testing on target domain. Following the literature (Carlucci et al., 2019), we use PACS (Li et al., 2017) as dataset, which consists of four domains, *viz*. photo, art, cartoon and sketch.

Results on single-source domain generalization are shown in Table 2. Here we report the relative improvement of InfoDrop over baseline. For the absolute accuracies, please refer to Appendix. Compared with baseline, InfoDrop boosts performances in multiple settings, especially with sketch as the source or target domain. This also reflects the shape-bias of InfoDrop, considering that sketches mainly consist of shape information. It is also worth noticing that our model can keep the performance on the *source* domain after InfoDrop is applied.

We also obtain results on multi-source domain generalization. Table 3 shows results on each domain after trained on other three domains. When trained with InfoDrop, the model is more robust to the distribution shift between different domains, and obtains consistent improvements over all target domains. Moreover, the vanilla baseline with InfoDrop is already better than or comparable with other state-of-the-art methods on each target domain.

4.2.2. Few-shot Classification

Current CNNs rely on huge amount of labeled data to learn powerful representations for downstream tasks. However, the learned representations may generalize poorly to unseen objects and scenes. This is in contrast to the human visual

Table 4. Few-shot classification results under different settings with ProtoNet as baseline. All experiments are 5-way. Usage of data augmentation is denoted by 'w/', and vice versa.

	CUB				*mini*-IMAGENET				*mini*-IMAGENET→CUB			
	5-SHOT		1-SHOT		5-SHOT		1-SHOT		5-SHOT		1-SHOT	
	W/O	W/	W/O	W/	W/O	W/	W/O	W/	W/O	W/	W/O	W/
PROTONET	67.13	77.64	51.62	58.83	63.84	66.85	47.96	47.17	52.71	54.62	**39.36**	35.24
+ INFODROP	**70.94**	**78.18**	**52.40**	**59.06**	**66.85**	**67.25**	**49.61**	**50.09**	**55.06**	**55.09**	37.11	**37.50**

Table 5. Few-shot classification results with different baseline methods. All results are from 5-way classification on CUB without data augmentation.

	5-SHOT	1-SHOT
MATCHINGNET	71.18 ± 0.70	57.81 ± 0.88
+ INFODROP	$\mathbf{72.32 \pm 0.69}$	$\mathbf{57.88 \pm 0.91}$
PROTONET	67.13 ± 0.74	51.62 ± 0.90
+ INFODROP	$\mathbf{70.94 \pm 0.72}$	$\mathbf{52.40 \pm 0.90}$
RELATIONNET	69.85 ± 0.75	56.71 ± 1.01
+ INFODROP	$\mathbf{73.72 \pm 0.71}$	$\mathbf{59.21 \pm 0.98}$

system, which is able to quickly grasp the feature of an unseen object given only a few examples. To this end, the task of *few-shot classification* is proposed, where a model needs to recognize classes unseen during training with limited examples. The main challenge here is the huge class-wise distribution shift. Following the literature, we use 'm-way n-shot classification' to refer to the setting where test data come from m novel classes each with n examples provided.

Following the setting in Chen et al. (2019), we evaluate InfoDrop on two popular datasets: CUB (Wah et al., 2011) and *mini*-ImageNet (Ravi & Larochelle, 2017), meanwhile also test our model in the cross-domain scenario (Chen et al., 2019), where *mini*-ImageNet is used for training and CUB for testing. We denote this setting by *mini*-ImageNet→CUB. For a full comparison, we test models trained both with and without data augmentation. For baseline algorithms, we follow Chen et al. (2019) and adopt three common approaches, *viz*. ProtoNet (Snell et al., 2017), MatchingNet (Vinyals et al., 2016) and RelationNet (Sung et al., 2018).

First we use ProtoNet as baseline and evaluate our method under different settings (Table 4). Under almost all the settings, InfoDrop brings a non-trivial improvement in performance. One may notice that improvements on *mini*-ImageNet are larger than CUB, which is reasonable due to the larger distribution shift to overcome in *mini*-ImageNet (Chen et al., 2019). As another observation, the improvements on 5-shot classification is larger than 1-shot. This implies that despite the robustness of shape features, they may not be as discriminative as texture features, hence requiring more examples for recognition. As a consequence, we may still need *some* texture to learn a discriminative and robust model (Sec. 4.4). Also, note that for baseline method, sometimes data augmentation may damage performance, which is possibly because augmentation leads to overfitting in the base classes. However, similar behavior is not observed on InfoDrop.

Then we check whether InfoDrop can bring a consistent improvement on different baselines. As shown in Table 5, on three baseline methods, InfoDrop improves the robustness universally. Note that InfoDrop most benefits RelationNet, possibly because its relation head learns a better similarity metrics between complex shapes.

4.2.3. ROBUSTNESS AGAINST IMAGE CORRUPTION

It is essential for visual models to give stable predictions under various kinds of corruptions (*e.g*. weather, blur, noise), especially in safety-critical situations. However, current CNNs are vulnerable to random corruptions and hardly generalize to different kinds of corruptions when trained on a specific one (Dodge & Karam, 2017). Recently, Geirhos et al. (2019) find that a *consistently* improved robustness against different corruptions can be achieved by training a shape-biased model. In Hendrycks & Dietterich (2019), benchmarks of model robustness are provided on 18 common types of corruption. In our experiments, we apply the same corruption functions on Caltech-256 dataset (Griffin et al., 2007) to test the robustness of InfoDrop. For comparison, we also test robustness of adversarially trained networks with and without InfoDrop. Adversarial training is known to improve robustness to noise and blur corruptions, while degrade performance on some others (*e.g*. fog, contrast) (Gilmer et al., 2019). Results are shown in Table 6. Due to limited space, we only show 12 types of corruptions here. Full comparisons can be found in Appendix. Clearly, InfoDrop improves baseline's robustness against most corruptions (*e.g*. noise, weather, digital) universally, although no noisy data is used for training. This also implies the potential of InfoDrop to generalize to other untested types of corruptions. Nonetheless, the performance may further degrade under blurring nonetheless, which is reasonable because blurring brings more distortion of shapes while others mainly corrupts texture information. It is also noticeable that InfoDrop can be incorporated with adversarial training

Table 6. Classification accuracy on clean and randomly corrupted images. 'A' and 'I' means usage of adversarial training and InfoDrop, respectively. All corruptions are generated under severity of level 1 (Hendrycks & Dietterich, 2019).

A	I	CLEAN	NOISE			BLUR			WEATHER			DIGITAL		
			GAUSSIAN	SHOT	IMPULSE	DEFOCUS	MOTION	GAUSSIAN	SNOW	FROST	FOG	ELASTIC	JPEG	SATURATE
✗	✗	82.98	66.38	62.85	49.97	**65.97**	**74.79**	**78.75**	53.10	67.09	72.42	**76.58**	79.77	77.15
✗	✓	**83.14**	69.58	66.83	53.00	62.52	71.76	77.03	56.44	**69.80**	**72.75**	74.54	**80.49**	**77.77**
✓	✗	79.69	75.30	73.80	70.71	61.53	71.68	73.77	61.11	69.06	54.52	71.69	79.31	72.62
✓	✓	78.59	**76.17**	**74.90**	**72.26**	62.32	71.32	74.04	**61.69**	**69.83**	55.00	70.26	78.10	71.26

Table 7. Adversarial robustness under different perturbation norm on CIFAR-10. 'A' and 'I' refer to the usage of adversarial training and InfoDrop, respectively.

A	I	$\ell_\infty = 0$	$\ell_\infty = \frac{1}{255}$	$\ell_\infty = \frac{2}{255}$	$\ell_\infty = \frac{8}{255}$
✗	✗	94.57	55.26	7.99	0.01
✗	✓	94.08	59.35	12.41	0.03
✓	✗	86.62	82.03	77.44	42.05
✓	✓	86.50	82.06	77.41	43.19

and obtain even better robustness with little overhead.

4.2.4. Adversarial Robustness

Except for random corruptions, CNNs are also vulnerable to carefully-designed imperceptible perturbations, namely adversarial perturbations (Szegedy et al., 2013). This leads to another crucial challenge for current CNN-based models. Most work on adversarial robustness is based on adversarial training (Madry et al., 2018). To evaluate adversarial robustness of InfoDrop, we conduct ablations on both baseline and adversarial trained models. Following the literature, we use CIFAR-10 (Krizhevsky et al., 2009), a widely-reported benchmark. For attacking, we use 20 runs of PGD (Madry et al., 2018) with constrained ℓ_∞ norm in both adversarial training and testing. As shown in Table 7, InfoDrop can improve robustness of baseline models under low-norm attack, but it still fails when the perturbation is large. Moreover, InfoDrop can be combined with adversarial training and provide extra robustness. Under the norm $\ell_\infty = \frac{8}{255}$, InfoDrop brings an improvement of 1% accuracy.

4.2.5. Comparison with Other Shape-biased Methods

Some approaches have also been proposed recently to train a shape-biased model. For example, Geirhos et al. (2019) propose to train the network on extra images with various texture styles in order to learn the shared shape features. Wang et al. (2019b) propose to use Gray-level Co-occurrence Matrix (Lam, 1996) as an indicator of texture, and decompose the feature from it. Other attempts include using different auxiliary tasks (Wang et al., 2019a; Carlucci et al., 2019).

Here we compare InfoDrop with the approach in Geirhos et al. (2019), which pretrains the network on ImageNet (IN) as well as Stylized-ImageNet (SIN). For comparison with other shape-biased methods, please refer to Appendix. Specifically, we evaluate the performances on single-source domain generalization. We compare InfoDrop (pretrained only on IN) with a ResNet50 pretrained on both IN and SIN. Results are shown in Table 8. We can see that both methods can bring an improvement in the model robustness. Especially, pretraining on SIN can largely increase the accuracy on Sketch domain, which is probably because SIN already contains images with sketch style. Remarkably, InfoDrop can improve the robustness consistently without seeing any target domain examples beforehand.

Table 8. Performance of different shape-biased methods on single-source domain generalization. Here we use Photo as the source domain, and report the accuracies on the other three target domains. Baseline indicates a simple ResNet50 model. † means extra finetuning on ImageNet is required during pretraining.

	Art	Cartoon	Sketch
Baseline	73.68	34.34	36.73
IN + SIN	72.80	40.04	58.70
IN + SIN†	74.51	38.38	42.61
InfoDrop	74.07	41.40	54.31

4.3. Ablation Studies

In this section we mainly discuss the role of temperature T in Eq. 1. Further ablations on other hyper-parameters (*e.g.* 'dropout rate' r_0, number of InfoDrop applied) can be found in supplementary. Intuitively, lower temperature means more conservative filtering, *i.e.*, only local regions with an almost constant value are dropped, while most shape and texture are preserved. An infinite temperature, however, will wipe out differences between shape and texture and act in a purely random way as regular Dropout. Apparently, somewhere between is what we intend for, where it can distinguish shape and texture, and filter out the latter. As verification, we conduct ablations on 5-way 1-shot classification on *mini*-ImageNet→CUB. As shown in Table 9, it reaches the highest accuracy when $T = 1$. Higher or lower T will degrade the performance. This means to be more robust, the model needs to filter out textures whilst preserve

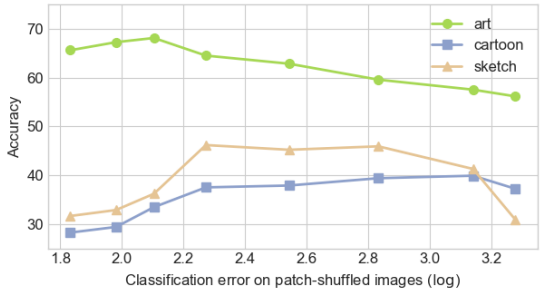

Figure 5. Domain generalization performance of models with different levels of shape-bias. The x-axis is the classification error on images with shuffled (3×3) patches, which is used as a indicator of the shape-bias level, *i.e.*, models with larger shape-bias tend to fail to recognize patch-shuffled images.

Table 9. Ablation study of temperature T in few-shot classification. Here we use ProtoNet as baseline (denoted by '-'). When $T = inf$, it degrades to regular Dropout.

T	-	0.1	0.5	1.0	3.0	INF
ACC	35.24	36.33	37.50	37.89	36.21	35.54

shape information, which is consistent with our analysis.

4.4. Is Shape Information All You Need?

In previous sections we have demonstrated how shape-bias can benefit CNN's robustness under different scenarios. This brings us another question: how biased should our model be? For example, does a visual model still work well if it only perceives shape information? The answer may be "no", considering that texture information plays a different but also important role in the human visual system (*e.g.* multi-modal perception (Sann & Streri, 2007)). It is also verified in experiments on deep models (Xiao et al., 2019) that shape itself does not suffice for high-quality visual recognition. Intuitively, there should exist an optimal "bias level" so that the model can be robust enough and meanwhile recognize objects with a proper precision, and this optimal level may vary from task to task.

To verify this, we conduct experiments on domain generalization. Specifically, we tune the temperature T to train models with different levels of shape-bias. To quantify the shape-bias, we use the classification error on patch-shuffled images as an indicator, considering that larger shape-bias generally leads to higher classification error on patch-shuffled images. We use photo as source domain, and test the performances on art, cartoon and sketch. As shown in Fig. 5, the performances on all target domains all go through an ascending at first, and then fall back when the shape-bias keeps being enhanced. Moreover, different target domains prefer different optimal bias levels. This implies that current CNNs are overly texture-biased, and we need to reach a "sweet spot" between shape and texture.[2]

5. Conclusion

In this work, we aim at universally improving various kinds of robustness of CNN by alleviating its texture-bias. To reduce texture-bias, we get our inspiration from the human visual system and propose Informative Dropout, an effective model-agnostic algorithm. We detect texture and shape by the local self-information in an image, and use a Dropout-like algorithm to decorrelate the model output from the local texture. Through extensive experiments we observe improved shape-bias as well as various kinds of robustness. Furthermore, we find our method can be incorporated with other algorithms (*e.g.* adversarial training) and achieve higher robustness. Through this work, we shed some light on the relationship between CNN's shape-bias and robustness, as well as new approaches to trustworthy machine learning algorithms.

Acknowledgement

Prof. Yadong Mu is partly supported by National Key R&D Program of China (2018AAA0100702) and Beijing Natural Science Foundation (Z190001). Dr. Zhanxing Zhu is supported by National Natural Science Foundation of China (No.61806009 and 61932001), PKU-Baidu Funding 2019BD005 and Beijing Academy of Artificial Intelligence (BAAI). Dinghuai Zhang is supported by the Elite Undergraduate Training Program of Applied Math of the School of Mathematical Sciences at Peking University. The authors are thankful to Tianyuan Zhang, Dejia Xu, Yiwen Guo and the anonymous reviewers for the insightful discussions and useful suggestions.

References

Ahmed, N., Natarajan, T., and Rao, K. R. Discrete cosine transform. *IEEE transactions on Computers*, 100(1):90–93, 1974.

Azulay, A. and Weiss, Y. Why do deep convolutional networks generalize so poorly to small image transformations? *Journal of Machine Learning Research*, 20(184):1–25, 2019. URL http://jmlr.org/papers/v20/19-519.html.

Ballester, P. and Araujo, R. M. On the performance of

[2]One may wonder what is the proper relationship between shape and texture? Should they act like two separate cues in a parallel way, or in a hierarchical way, where shape first provides a quick, coarse recognition, and then details are observed through texture? We leave this for further exploration.

googlenet and alexnet applied to sketches. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

Biederman, I. Recognition-by-components: a theory of human image understanding. *Psychological review*, 94 (2):115, 1987.

Bisanz, J., Bisanz, G. L., and Kail, R. *Learning in children: Progress in cognitive development research*. Springer Science & Business Media, 2012.

Brendel, W. and Bethge, M. Approximating CNNs with bag-of-local-features models works surprisingly well on imagenet. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=SkfMWhAqYQ.

Bruce, N. and Tsotsos, J. Saliency based on information maximization. In *Advances in Neural Information Processing Systems*, pp. 155–162, 2006.

Bruce, N. D. and Tsotsos, J. K. Saliency, attention, and visual search: An information theoretic approach. *Journal of vision*, 9(3):5–5, 2009.

Carlucci, F. M., D'Innocente, A., Bucci, S., Caputo, B., and Tommasi, T. Domain generalization by solving jigsaw puzzles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2229–2238, 2019.

Chen, W.-Y., Liu, Y.-C., Kira, Z., Wang, Y.-C. F., and Huang, J.-B. A closer look at few-shot classification. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=HkxLXnAcFQ.

Csurka, G. *Domain adaptation in computer vision applications*. Springer, 2017.

Dhillon, G. S., Chaudhari, P., Ravichandran, A., and Soatto, S. A baseline for few-shot image classification. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=rylXBkrYDS.

Diesendruck, G. and Bloom, P. How specific is the shape bias? *Child development*, 74(1):168–178, 2003.

Dodge, S. and Karam, L. A study and comparison of human and deep learning recognition performance under visual distortions. In *2017 26th international conference on computer communication and networks (ICCCN)*, pp. 1–7. IEEE, 2017.

D'Innocente, A. and Caputo, B. Domain generalization with domain-specific aggregation modules. In *German Conference on Pattern Recognition*, pp. 187–198. Springer, 2018.

Fritz, G., Seifert, C., Paletta, L., and Bischof, H. Attentive object detection using an information theoretic saliency measure. In *International workshop on attention and performance in computational vision*, pp. 29–41. Springer, 2004.

Geirhos, R., Janssen, D. H., Schütt, H. H., Rauber, J., Bethge, M., and Wichmann, F. A. Comparing deep neural networks against humans: object recognition when the signal gets weaker. *arXiv preprint arXiv:1706.06969*, 2017.

Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., and Brendel, W. Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bygh9j09KX.

Gilmer, J., Ford, N., Carlini, N., and Cubuk, E. Adversarial examples are a natural consequence of test error in noise. In *International Conference on Machine Learning*, pp. 2280–2289, 2019.

Gretton, A., Borgwardt, K., Rasch, M., Schölkopf, B., and Smola, A. J. A kernel method for the two-sample-problem. In *Advances in Neural Information Processing Systems*, pp. 513–520, 2007.

Gretton, A., Smola, A., Huang, J., Schmittfull, M., Borgwardt, K., and Schölkopf, B. Covariate shift by kernel mean matching. *Dataset shift in machine learning*, 3(4): 5, 2009.

Griffin, G., Holub, A., and Perona, P. Caltech-256 object category dataset. 2007.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Hendrycks, D. and Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=HJz6tiCqYm.

Hou, X. and Zhang, L. Saliency detection: A spectral residual approach. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8. Ieee, 2007.

Huang, G., Larochelle, H., and Lacoste-Julien, S. Are few-shot learning benchmarks too simple ?, 2020. URL https://openreview.net/forum?id=SygeY1SYvr.

Huang, X. and Belongie, S. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1501–1510, 2017.

Itti, L., Koch, C., and Niebur, E. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (11):1254–1259, 1998.

Jo, J. and Bengio, Y. Measuring the tendency of cnns to learn surface statistical regularities. *arXiv preprint arXiv:1711.11561*, 2017.

Khosla, A., Zhou, T., Malisiewicz, T., Efros, A. A., and Torralba, A. Undoing the damage of dataset bias. In *European Conference on Computer Vision*, pp. 158–171. Springer, 2012.

Kriegeskorte, N. Deep neural networks: a new framework for modeling biological vision and brain information processing. *Annual review of vision science*, 1:417–446, 2015.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Lam, S.-C. Texture feature extraction using gray level gradient based co-occurence matrices. In *1996 IEEE International Conference on Systems, Man and Cybernetics. Information Intelligence and Systems (Cat. No. 96CH35929)*, volume 1, pp. 267–271. IEEE, 1996.

Landau, B., Smith, L. B., and Jones, S. S. The importance of shape in early lexical learning. *Cognitive development*, 3(3):299–321, 1988.

LeCun, Y., Bengio, Y., and Hinton, G. Deep learning. *nature*, 521(7553):436, 2015.

Li, D., Yang, Y., Song, Y.-Z., and Hospedales, T. M. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 5542–5550, 2017.

Li, H., Jialin Pan, S., Wang, S., and Kot, A. C. Domain generalization with adversarial feature learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5400–5409, 2018.

Long, M., Cao, Y., Wang, J., and Jordan, M. Learning transferable features with deep adaptation networks. In *International Conference on Machine Learning*, pp. 97–105, 2015.

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

Muandet, K., Balduzzi, D., and Schölkopf, B. Domain generalization via invariant feature representation. In *International Conference on Machine Learning*, pp. 10–18, 2013.

Radenovic, F., Tolias, G., and Chum, O. Deep shape matching. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 751–767, 2018.

Ravi, S. and Larochelle, H. Optimization as a model for few-shot learning. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=rJY0-Kcll.

Renninger, L. W., Coughlan, J. M., Verghese, P., and Malik, J. An information maximization model of eye movements. In *Advances in Neural Information Processing Systems*, pp. 1121–1128, 2005.

Ritter, S., Barrett, D. G., Santoro, A., and Botvinick, M. M. Cognitive psychology for deep neural networks: A shape bias case study. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 2940–2949. JMLR. org, 2017.

Sann, C. and Streri, A. Perception of object shape and texture in human newborns: evidence from cross-modal transfer tasks. *Developmental Science*, 10(3):399–410, 2007.

Shannon, C. E. A mathematical theory of communication. *Bell system technical journal*, 27(3):379–423, 1948.

Shimodaira, H. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of statistical planning and inference*, 90(2):227–244, 2000.

Smilkov, D., Thorat, N., Kim, B., Viégas, F., and Wattenberg, M. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

Snell, J., Swersky, K., and Zemel, R. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, pp. 4077–4087, 2017.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P. H., and Hospedales, T. M. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1199–1208, 2018.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*, pp. 3630–3638, 2016.

Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. The caltech-ucsd birds-200-2011 dataset. 2011.

Wang, H., Ge, S., Lipton, Z., and Xing, E. P. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pp. 10506–10518, 2019a.

Wang, H., He, Z., and Xing, E. P. Learning robust representations by projecting superficial statistics out. In *International Conference on Learning Representations*, 2019b. URL https://openreview.net/forum?id=rJEjjoR9K7.

Xiao, C., Sun, M., Qiu, H., Liu, H., Liu, M., and Li, B. Shape features improve general model robustness. 2019. URL https://openreview.net/forum?id=SJlPZlStwS.

Yarbus, A. L. *Eye movements and vision*. Springer, 2013.

Zeiler, M. D. and Fergus, R. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pp. 818–833. Springer, 2014.

Zhang, D., Zhang, T., Lu, Y., Zhu, Z., and Dong, B. You only propagate once: Accelerating adversarial training via maximal principle. In *Advances in Neural Information Processing Systems*, pp. 227–238, 2019.

Zhang, T. and Zhu, Z. Interpreting adversarially trained convolutional neural networks. In *International Conference on Machine Learning*, pp. 7502–7511, 2019.