HUMAN IMPERCEPTIBLE ATTACKS AND APPLICATIONS TO IMPROVE FAIRNESS

Anonymous authors
Paper under double-blind review

ABSTRACT

Modern neural networks are able to perform at least as well as humans in numerous tasks involving object classification and image generation. However, small perturbations which are imperceptible to humans may significantly degrade the performance of well-trained deep neural networks. We provide a Distributionally Robust Optimization (DRO) framework which integrates human-based image quality assessment methods to design optimal attacks that are imperceptible to humans but significantly damaging to deep neural networks. Our attack algorithm can generate better-quality (less perceptible to humans) attacks than other state-of-the-art human imperceptible attack methods. We provide an algorithmic implementation of independent interest which can speed up DRO training significantly. Finally, we demonstrate how DRO training using our optimally designed human imperceptible attacks can improve group fairness in image classification while maintaining a similar accuracy level.

1 INTRODUCTION

Deep learning models are making strides into our daily life with tremendous successes in diverse areas of applications, such as self-driving cars and face recognition. However, we still lack fundamental understanding in how deep neural networks (DNNs) perceive and process information. One behavior of DNNs that we do not fully understand is how they are impacted by adversarial attacks. The potential implication of these attacks involve threats in, for instance, safety and fairness. A formal definition of adversarial attacks on image classification (Moosavi-Dezfooli et al., 2016) is the following. Given the classifier f, an image $\boldsymbol{x}$, and a cost function c on the image space, an optimal adversarial attack solves $\boldsymbol{\delta}$ that can change the model's classification results with the smallest budget:

$$\min_{\boldsymbol{\delta}} c(\boldsymbol{x}, \boldsymbol{x}+\boldsymbol{\delta}), \quad \text{with} \quad f(\boldsymbol{x}+\boldsymbol{\delta}) \neq f(\boldsymbol{x}). \tag{1}$$

Our goal in this paper is in the systematic study of adversarial attack which are imperceptible to humans. So, we constrain the attacked image to be close to the original image in a chosen cost function which models human perception. This is one of the key features of our contribution, which is distinct relative to the literature on adversarial attacks. Traditional cost functions involve L_p distances, see (Goodfellow et al., 2014; Madry et al., 2017; Moosavi-Dezfooli et al., 2016; Tramèr et al., 2018). However, as reported in recent literature (Sharif et al., 2018; Wang et al., 2004), L_p distances do not accurately measure differences in human perception. In this paper, we study two better choices of cost function and demonstrate that we generate adversarial attacks of better quality (less perceptible to humans). We also combine Distributionally Robust Optimization (DRO) training with our attack method and test that our training algorithm improves fairness in classification.

Neural network design, over the years, has been inspired by the ways in which the human brain responds to visual stimuli (Xu & Vaziri-Pashkam, 2021; Voulodimos et al., 2018). On the other hand, in traditional computer vision, scientists handcraft features, for example SIFT (Lowe, 1999) and HOG (Dalal & Triggs, 2005), that they believe are important in classification. Modern DNNs, instead of relying on handcrafted features, focus on learning them for the task on hand (LeCun et al., 2015). Although adversarial attacks are targeted toward DNNs, they may cause differences to human vision systems as well (Zhou & Firestone, 2019; Elsayed et al., 2018). In our work, we study adversarial attacks that only have effects to machine vision and demonstrate that by making algorithms more resilient to these attacks, we are able to improve the performance of these algorithms in tasks that are important from a human vision standpoint.

To design adversarial attacks that humans cannot perceive, the choice of cost function c in equation 1 is important. As mentioned before, L_p cost functions may not constrain adversarial attacks to be imperceptible to humans. In our

work, we use human perceptual distances SSIM (Wang et al., 2004) and PieAPP (Prashnani et al., 2018) as our cost functions. Moreover, we aim to align machine vision or machine perception to human perception by DRO training with our attacks. The relationship between our human-imperceptible attacks, other adversarial attacks, machine perception, and human perception is illustrated in Figure 1. The two images from left to right also demonstrate how DRO training with our attack method aligns machine perception with human perception.

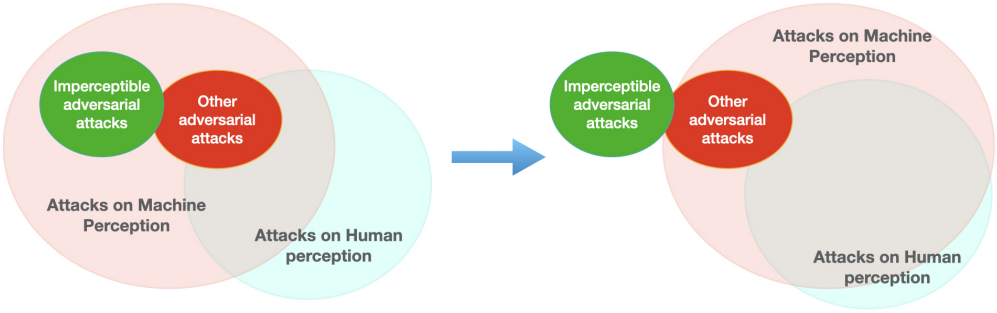


Figure 1: **Left**: The two big circles represent the attacks or small perturbations that can cause changes visually in machines perception and human perception respectively. Commonly used adversarial attacks (the red circle), for example L_p based adversarial attacks, are visually perceived by both machines and humans. We design our attacks (the green circle) to be imperceptible to humans and only affect machine perception. Then we start adversarial training with our attack method and move the relationship as displayed in the left image to the right image. **Right**: Adversarial training with our attacks discourages the model perceiving the attacks in the green circle, so it pushes the machine perception circle to the direction of human perception. In the end of the adversarial training, the perturbations that machines perceive and humans perceive overlap more, so that the two perception systems align more closely.

DRO framework is studied extensively in machine learning, because it can compute the best model under distributional uncertainty (Blanchet et al., 2018; Rahimian & Mehrotra, 2019). DRO-trained models are able to achieve uniform performance across all groups of data, even on out of sample data (Blanchet & Kang, 2021). DRO framework has been studied with data-driven distances (Blanchet et al., 2019b). In our work, our adversarial attack method solves exactly the inner sup problem with the cost function informed by human-based IQA methods (mathematical formulations are in Section 3). Solving the whole DRO mini-max problem trains the model to advance in the direction of human perception. We show that DRO training with our adversarial attacks improves group fairness compared to DRO training using PGD attack method.

Recently, fairness in machine learning has become a crucial topic. As we apply machine learning models in daily applications, we need models to be fair across the whole distribution, especially the data from underrepresented groups. Unfortunately, current datasets do not have a uniform distribution on images from all demographics. In both of the two popular open-source data sets: ImageNet and Open Images, approximately half of the images are collected from 2 countries: the United States and Great Britain (Shankar et al., 2017). Moreover, DNNs are suspected to learn spurious features to help classification and the spurious features are learned from the majority groups (de Vries et al., 2019). Both Shankar et al. (2017) and de Vries et al. (2019) define image groups as the country where images are collected, so we follow the definition and collect our Imagenet geo-location dataset. In the class mailbox in our collected dataset, we count over half of the mailboxes are red, so a DNN is likely to learn the spurious feature that mailboxes are all red and the probability of correctly classifying the mailbox from Cambodia (in Figure 2) decreases. However, as humans understand the meaning of word "mailbox", we are unlikely to use the color to classify mailboxes. There are other features that DNNs can perceive and possibly learn, for example, noises or textures in the images. By our proposed adversarial attack method and DRO training algorithms, we train the models to perceive information as humans perceive and perform more equally on the groups.

Figure 2: Mailbox from United Kingdom (**Left**) and from Cambodia (**Right**).

Our work's contributions are as follows:

1. We connect a human perception distance PieAPP with the DRO framework to generate adversarial attacks that are imperceptible to humans and successfully attack the classification models. We use two methods

introduced in the human vision learning area to show that our attacks are less perceptible to humans than other state-of-the-art (SOTA) imperceptible attacks. We incorporate confidence in our algorithm, so our method with high confidence attacks successfully against 2 defense methods.

2. We provide an algorithmic implementation of independent interest which can speed up DRO training significantly. We add a few speed-up techniques to make generating attacks and DRO training more practical.
3. We collect a dataset from ImageNet (Russakovsky et al., 2015) with country information[1], and design two hypothesis tests to test that DRO training with our attacks improves fairness in classification. The hypothesis tests can serve as a general methodology to test group fairness.

This paper unfolds as follows. In Section 2, we list related publications. In Section 3, we introduce our adversarial attack method that computes optimal imperceptible attacks. We also show numerical comparison results and image comparison examples. In Section 4, we demonstrate two algorithms and design two hypothesis tests to compare fairness on our method and PGD method. All the dataset, code and figures are available in the supplementary file.

2 Related Work

Adversarial attacks. Since the seminal work (Goodfellow et al., 2014), there is a surge of papers studying adversarial attacks: Carlini & Wagner (2017); Madry et al. (2017); Moosavi-Dezfooli et al. (2016); Kurakin et al. (2016); Dong et al. (2018); Chakraborty et al. (2018). There are papers that use Wasserstein distance (Wong et al., 2019), human perception distance (Zhao et al., 2020; Laidlaw et al., 2021), attacks in feature space (Xu et al., 2020). Other non-conventional adversarial attacks are: sparse adversarial attacks (Andriushchenko et al., 2020; Zhu et al., 2021), spatial perturbations (Engstrom et al., 2019; Zeng et al., 2019), and one-pixel attack (Su et al., 2019). Other than white-box attack methods, there are many successful black-box attack methods (Guo et al., 2019; Ilyas et al., 2018). We cannot include all the adversarial attack papers here, so refer to the review papers (Akhtar & Mian, 2018; Chakraborty et al., 2018) for more references.

Adversarial attack and human vision. There is literature studying the influences of adversarial attacks on human vision (Zhou & Firestone, 2019; Elsayed et al., 2018). Zhao et al. (2017) generate adversarial attacks that are semantically meaningful, which the contents of images are changed in human eyes as well. Madry et al. (2017) also report that L_2 based attacks can be large enough to cause misclassification by humans.

Human perceptual distance. In order to truly restrict the influences of adversarial attacks in human eyes, we need distance functions to measure differences in human vision. Image quality assessment (IQA) is a line of work to measure human perceptual distances. Traditional IQA methods include SSIM (Wang et al., 2004), MS-SSIM (Wang et al., 2003), and FSIM (Zhang et al., 2011). Deep neural network based IQA methods include DISTS (Ding et al., 2020), PieAPP (Prashnani et al., 2018), LPIPS (Zhang et al., 2018), and SWD (Gu et al., 2020).

DRO. As people care more about models' robustness in the extreme circumstances, DRO framework emerged to gain a lot of interests. There have been a number of theorectical work on DRO and Optimal Transport, see Blanchet et al. (2018); Blanchet & Murthy (2019); Duchi & Namkoong (2021); Rahimian & Mehrotra (2019); Kuhn et al. (2019); Staib & Jegelka (2019); Van Parys et al. (2021). In particular, Esfahani & Kuhn (2018); Shafieezadeh-Abadeh et al. (2015); Gao & Kleywegt (2016); Gao et al. (2017); Blanchet et al. (2019a) study the theory and applications of DRO problems using Wasserstein distance to parameterize the constaint set. Volpi et al. (2018) generalizes models to unseen domains by training the models with DRO. Sinha et al. (2018) first introduces combining DRO framework and adversarial attacks. Dong et al. (2020) introduces adversarial distributional training (ADT) to generalize the usual adversarial training as a special case, and solve the usual AT deficiencies introduced in Tramèr et al. (2018); Zhang & Wang (2019).

Fairness. Many recent papers discover unfairness in image classification and object detection models (de Vries et al., 2019; Wilson et al., 2019; Buolamwini & Gebru, 2018). Specifically, these papers point out that neural network models discriminate against underrepresented groups. One possible explanatory factor of unfairness is that the open-source datasets are unbalanced (Shankar et al., 2017). Yang et al. (2020); Gong et al. (2012) starts to fix the datasets by collecting data that are representative among all demographics. In natural language processing community, recent work discovers that word embedding models learn the biases from data (Bolukbasi et al., 2016; Caliskan et al., 2017). Mehrabi et al. (2021) is a recent review paper on fairness in machine learning community.

[1]The dataset is public and the link is in README file in the supplementary file.

3 METHOD

To adversarially train the model, we consider the following DRO problem, which calculates the model that could perform well in the worst case scenario:

$$\min_{\theta} \sup_{P:D(P,P_0)<\delta} \mathbb{E}_P[\ell(\theta; X, Y)], \tag{2}$$

where θ is the model parameter, ℓ is the loss function, P_0 is the empirical distribution of training data (X, Y), and D is the distance metric to characterize the set of distributions we want to generalize to.

Similar to Sinha et al. (2018), we choose Wasserstein distance as our metric D. Specifically, let $c((\boldsymbol{x}, y), (\boldsymbol{x}', y'))$ denote the cost function to measure the distances between two training samples $(\boldsymbol{x}, y)$ and $(\boldsymbol{x}', y')$ and $\Gamma(P, P_0)$ denote the set of all joint distributions of P and P_0, then our distance metric is given by

$$D(P, P_0) = \inf_{\gamma \in \Gamma(P,P_0)} \mathbb{E}_\gamma[c((\boldsymbol{x}, y), (\boldsymbol{x}', y'))],$$

where $c((\boldsymbol{x}, y), (\boldsymbol{x}', y')) = c_0(\boldsymbol{x}, \boldsymbol{x}') + \infty \cdot \mathbb{1}\{y \neq y'\} = c_0(\boldsymbol{x}, \boldsymbol{x}')$, as we are only interested in adversarial attacks and adversarial attacks that do not change the label.

To obtain a computationally feasible solution for equation 2, we consider its Lagrangian relaxation with penalty parameter λ

$$\min_{\theta} \sup_{P} \mathbb{E}_P[\ell(\theta; X, Y) - \lambda D(P, P_0)].$$

By Lemma 1 of Volpi et al. (2018), the inner optimization over P can be explicitly solved by

$$\min_{\theta} \sup_{P} \mathbb{E}_P[\ell(\theta; X, Y) - \lambda D(P, P_0)] = \min_{\theta} \mathbb{E}_{P_0}[\phi_\lambda(\theta; X, Y)], \tag{3}$$

where the robust surrogate loss ϕ_λ is defined by

$$\phi_\lambda(\theta; \boldsymbol{x}_0, y_0) = \sup_{\boldsymbol{x}} \left(\ell(\theta; \boldsymbol{x}, y_0) - \lambda c_0(\boldsymbol{x}, \boldsymbol{x}_0)\right). \tag{4}$$

Note that $\{\boldsymbol{x}_0, y_0\}$ is one data sample, where $\boldsymbol{x}_0 \in \mathbb{R}^n$ is a high-dimensional vector representation of the image and y_0 is its label.

As discussed in Section 1, instead of using L_p distances, we will define c_0 by two distances that better represent human perceptual distances, $c_0 = 1 -$ SSIM and $c_0 =$ PieAPP, which are discussed below in more detail:

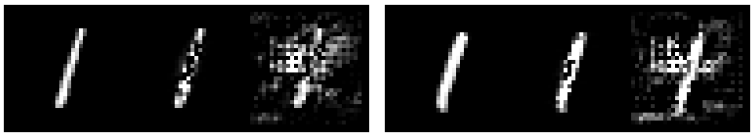
1. $c_0 = 1 -$ SSIM. Structural similarity index measure (SSIM) (Wang et al., 2004) is a reward function on two grayscale images that captures structural similarity in the two images. Wang & Bovik (2009) shows a table of the same image distorted by different methods. The table of images demonstrate that even with the same MSE error, two images have drastically different quality in human eyes, but two images with similar SSIM values are of similar quality in human eyes. SSIM's formula is stated in Section A. SSIM ≤ 1, so we use $1 -$ SSIM as a cost function. $1 -$ SSIM satisfy three properties: symmetry, boundedness, and unique minimum, which makes it very close to a distance function.
2. $c_0 =$ PieAPP. PieAPP (Prashnani et al., 2018) uses a DNN to measure two images' visual differences in human judgement. PieAPP can be applied on RGB images of size greater than 64×64. PieAPP measures a novel pairwise preference probability, for instance, the probability that humans prefer image A over image B with respect to a reference image R, which is more robust because humans may have clear preferences between all pairs of images but do not have a clear ranking over all images. Another advantage is that PieAPP does not depend on any existed architectures or pretrained models, as opposed to LPIPS and DISTS.

3.1 SSIM BASED ATTACK

In this section, we focus on $c_0 = 1 -$ SSIM. Let $d_{\boldsymbol{x}} = c_0(\boldsymbol{x}, \boldsymbol{x} + \boldsymbol{\Delta})$ and $\boldsymbol{H}_{\boldsymbol{x}}(\boldsymbol{\Delta})$ denote its Hessian matrix. We solve a one-step type of attack by solving equation 4:

$$\boldsymbol{x}_{adv} = \boldsymbol{x} + \epsilon \frac{\boldsymbol{\Delta}^*}{\|\boldsymbol{\Delta}^*\|_2}, \qquad \text{with} \quad \boldsymbol{\Delta}^* = \frac{1}{\lambda} \boldsymbol{H}_{\boldsymbol{x}}(\boldsymbol{0})^{-1} \nabla_{\boldsymbol{x}} \ell(\theta; \boldsymbol{x}, y), \tag{5}$$

the derivation of equation 5 can be found in Section A. Due to the size and computation cost of the Hessian matrix, the above method is only practical with small images, for example on MNIST images of size $1 \times 28 \times 28$. We use this simple example to show that using $1 -$ SSIM as our cost function does discourage any structural changes that may change the true meaning of the images.



The classified labels are: 1, 8, 4 The classified labels are: 1, 4, 4

Figure 3: **Left**: Original, **Middle**: Our one-step method, **Right**: PGD (L_2). We compare our one-step method and one-step PGD attack using L_2 cost function, both with $\epsilon = 10$. Here are the first two 1's in MNIST test split that are successfully attacked by both methods. Our attack does not change the structure nor the true class of the numbers, but L_2 attacks make the digits unrecognizable and more similar to the mis-classified label. More examples can be found in Section B.

3.2 HUMAN PERCEPTION BASED ATTACK

In this section, our choice of human perceptual distance c_0 is PieAPP. On ImageNet like data sets, we attack $3 \times 299 \times 299$ size images and use gradient descent method to solve equation 4, as described in Algorithm 1. Specifically, we choose a pretrained ResNet-50 on ImageNet (He et al., 2016) as our model θ, cross-entropy loss as our loss function ℓ, and $N = 100$, $\epsilon = 0.1$ and confidence $a = \{0, 1, 5\}$. We incorporate confidence level in our early-stop step (line 4 in Algorithm 1) to enhance the strength of our attacks, so they successfully attack defended images in Section 3.3.

Algorithm 1 Human_Perc_OT: attack an image

Input: image $\boldsymbol{x}$, label y, loss function ℓ, cost function c_0, number of iterations N, model θ, step size ϵ, confidence a
Output: adversarial image $\boldsymbol{x}_{adv}$

1: **Initialize:** $\boldsymbol{x}_{adv} \leftarrow \boldsymbol{x}$
2: **for** $k = 1, 2, \ldots, N$ **do**
3: $\quad$ logits $= \theta(\boldsymbol{x}_{adv})$ $\quad\triangleright$ the logits before the softmax layer
4: $\quad$ **If** $\max \text{logits}_{i \neq y} - \text{logits}_y > a$ **then Output** $\boldsymbol{x}_{adv}$ **end if** $\quad\triangleright$ logits_i means the logit for class i
5: $\quad$ $\boldsymbol{\Delta} = \frac{\partial \ell(\theta; \boldsymbol{x}_{adv}, y)}{\partial \boldsymbol{x}_{adv}} - \lambda \frac{\partial c_0(\boldsymbol{x}_{adv}, \boldsymbol{x})}{\partial \boldsymbol{x}_{adv}}$ $\quad\triangleright$ We compute $\frac{\partial c_0(\boldsymbol{x}_{adv}, \boldsymbol{x})}{\partial \boldsymbol{x}_{adv}}$ every 5 steps for performance
6: $\quad$ $\boldsymbol{x}_{adv} \leftarrow \boldsymbol{x}_{adv} + \epsilon * \boldsymbol{\Delta}$
7: $\quad$ validate $\boldsymbol{x}_{adv}$ $\quad\triangleright$ Force $\boldsymbol{x}_{adv}$ to be a valid RGB image
8: **end for**

We compare our method with PGD (L_2), and with SOTA methods NPTM (Laidlaw et al., 2021) and PerC (Zhao et al., 2020) in terms of total time, success rate and human perceptibility. Specifically, we compare with NPTM (PPGD) and NPTM (LPA) (two methods proposed in the NPTM paper), and PerC_AL (the faster and less perceptible method in the paper). Different from PerC and NPTM, our attack method directly solves the inner optimization problem (equation 4 of the DRO problem). PerC_AL alternates between the two goals of attacking the image successfully and minimizing the perceptual distance, while our method combines the two goals in a single step (line 7 in Algorithm 1). NPTM (PPGD) and NPTM (LPA) requires an extra projection step, while our method does not.

Our attack method is evaluated on the development set of the ImageNet-Compatible dataset (same as Zhao et al. (2020)). Since the dataset has 1000 images and we plan to compare against four other methods, involving humans to judge every pair of images is expensive. Other than displaying images to qualitatively judge the attacks' imperceptibility, we apply two human perceptual distances and a salient object detection network to measure the quality of attacks.

We apply two Image Quality Assessment (IQAs) methods, LPIPS (Zhang et al., 2018) and DISTS (Ding et al., 2020), to quantify perceptual distance between two images in human vision. The numerical results are given in Table 1. Figure 4 provides two examples to visually compare the quality of attacks.

The last comparison method is applying EGNet (Zhao et al., 2019) to images. EGNet is a model to predict human saliency map, which means the object in an image that draws attention the most. We compute the human saliency maps of original images and the attacked images, and compute multiple distances between the original and attacked human saliency maps. Figure 6 illustrates two examples to qualitatively compare human saliency maps and Table 2 includes all the numerical comparison results.

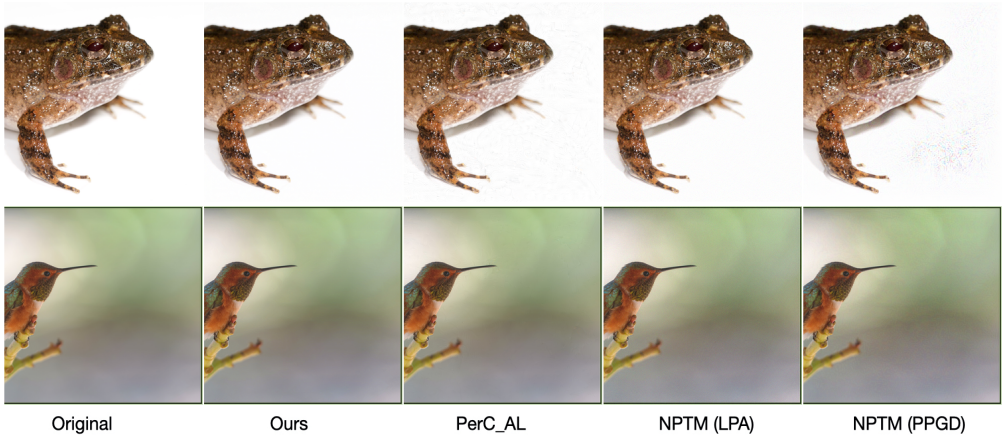


Figure 4: The comparison between the original image and adversarial attacks. Our method and PGD method generate images of similar visual quality, so we do not put PGD images here. PerC_AL has marble effects (in the background of the first image and around the beak in the second image). The image quality of LPA degrades, as there are noticeable sand effects in the second image, compared with other images. PPGD has an area of noises in the first image. The images in full resolution are available in the supplementary file. More results can be found in Figure 10.

Table 1: Each row shows one attack method. The first column is each method's success rate, which is defined as, the number of attacked images that labels change from being correct to incorrect divided by the number of correctly classified images. Remaining columns represent the distance functions to measure the difference between attacked images and original images. We embolden the smallest values in each column. Our method with $a = 0$ has the smallest human perceptual distances, despite larger L_p distances than PGD.

	Success Rate	L_1	L_2	L_∞	**LPIPS (x1000)**	**DISTS (x1000)**
PerC_AL	100%	633.116	2.216	0.085	33.960	33.819
PGD (L_2)	100%	**592.737**	**1.561**	**0.005**	7.823	8.770
NPTM (PPGD)	95.75%	2544.206	6.604	0.115	81.569	51.083
NPTM (LPA)	99.78%	2157.771	5.309	0.049	51.644	35.920
Human_Perc_OT ($a = 0$) (ours)	100%	783.855	1.905	0.006	**7.303**	**8.165**

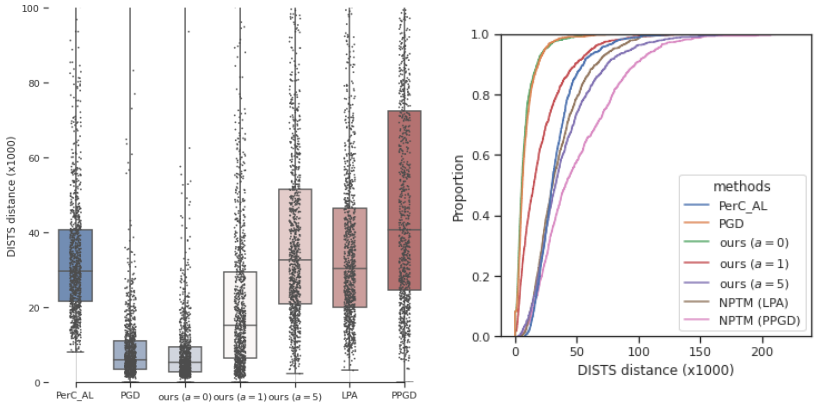


Figure 5: DISTS distances comparison between the attacks and original images. **Left**: boxplots of empirical DISTS distance distribution of all methods. The box's 3 bars each represents the distribution's 3rd quantile, median, and 1st quantile respectively. Our method with $a = 0$'s distribution has the smallest quantiles. **Right**: the cumulative distribution function (CDF) of empirical DISTS distance distribution of all attack methods. At any DIST distance d, our method with $a = 0$ has the largest $\mathbb{P}(\boldsymbol{x}_{adv} \leq d)$. Our method with $a = 1$ has smaller perceptual distances than PerC_AL, NPTM (LPA), and NPTM (PPGD). The same plots for LPIPS distance can be found in Figure 12.

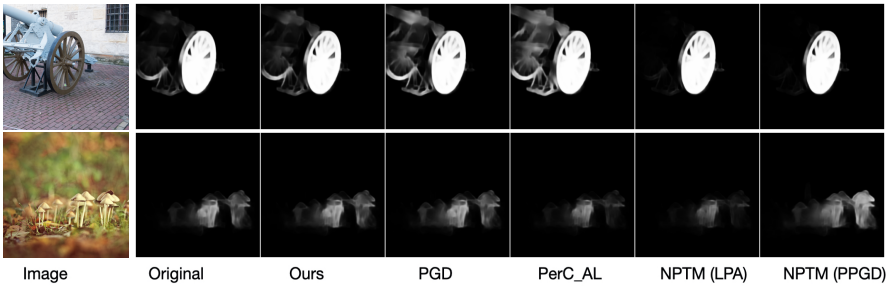


Figure 6: Comparison between the human saliency maps of the original image and attacked images. Compared with the original saliency map, our method generates the map with the least distinction. In the second image, NPTM (PPGD) shifts the attention from the mushrooms in the middle to the right. More results can be found in Figure 11.

Table 2: Comparing differences in human saliency maps for all methods. The first column is the attack's success rate (same values as Table 1) and rest columns are the distances between attacked images' saliency maps and original images' saliency maps. We embolden the smallest distance value. Our method with $a = 0$ generates attack images with the smallest distances in human saliency map, which means they induce the smallest changes in human saliency.

	Success Rate	L_1	L_2	L_∞	**SSIM**
PerC_AL	100%	1129.920	10.803	0.319	0.093
PGD (L_2)	100%	385.700	3.880	0.132	0.020
NPTM (PPGD)	95.75%	946.390	9.068	0.276	0.086
NPTM (LPA)	99.78%	522.030	5.337	0.188	0.036
Human_Perc_OT ($a = 0$) (ours)	100%	**325.729**	**3.339**	**0.118**	**0.016**

3.3 Comparison with other attacks on defenses

Without knowing which adversarial attack is applied on the image, there are generic defense methods against attacks. We test our method on two such defense methods: **jpeg compression** (Das et al., 2018; Dong et al., 2019; Dziugaite et al., 2016; Guo et al., 2017) and **bit depth reduction** (Guo et al., 2017; He et al., 2017; Xu et al., 2017). The comparison result is shown in Figure 7.

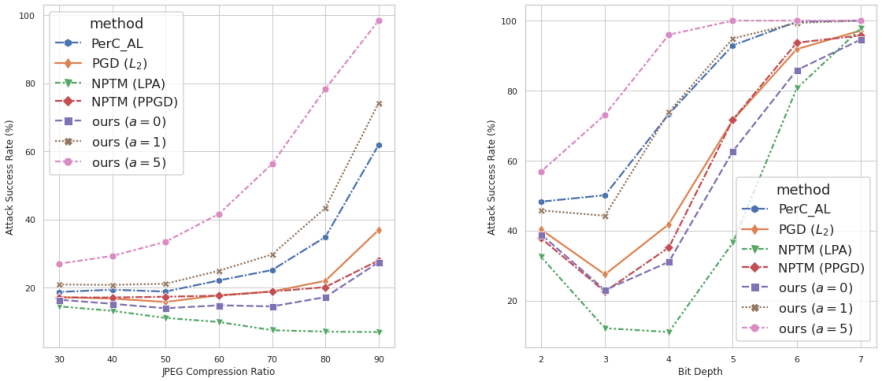


Figure 7: After the attacked being processed with jpeg compression defense and bit depth compression defense, our method with confidence 5 has the highest attack success rate in both defense methods and the images are still less perceptible than NPTM (PPGD)'s images (Figure 5). Our method with confidence 1 generates less perceptible attack images than PerC_AL and has a better attack success rate in jpeg compression defense.

4 Fairness

Recent works reveal that current open-source large datasets are severely unbalanced in the geographical location of images, and public object recognition systems do not perform uniformly on images from different locations (Shankar

et al., 2017; de Vries et al., 2019). Our collected ImageNet geo-location dataset is also amerocentric and eurocentric as discovered in Shankar et al. (2017), see Figure 13. The detailed procedure of collecting the dataset is in Section E.

In this section, we introduce a method to measure the level of fairness. We first fix a dataset and the empirical distribution is a uniform distribution. With the dataset, we can train a model with stochastic gradient descent and the model is a random vector based on the dataset, where randomness comes from the training process. Then we group our dataset into sets of images by their country information: $\{S_1, S_2, ...\}$. For each country, we know its income level $\mathbf{g}(S_i)$. Now we sample countries independently and denote accuracy of images in S_i as $\mathbf{p}(\theta; S_i)$. Ideally, for a country S_i and random vector θ, classification accuracy should be *independent* of the income level of the country:

$$\mathbb{P}\left(\mathbf{p}(\theta; S_i) \leq r | \mathbf{g}(S_i)\right) = \mathbb{P}(\mathbf{p}(\theta; S_i) \leq r), \quad \forall r \in [0, 1]. \tag{6}$$

After we sample a θ_0 from the model space, we can test independence based on model by testing whether there exists relationship between $\mathbf{p}(\theta_0; S_i)$ and $\mathbf{g}(S_i)$. If there exists a relationship for a significant probability of all the models, we know equation 6 cannot hold. For example, we know $\mathbf{p}(\theta; S_i) = \mathbf{g}(S_i)$ for 80% of the models and all models have an accuracy between $[0.6, 0.7]$ over the whole dataset, then $\mathbb{P}(\mathbf{p}(\theta; S_i) \leq 0.1 | \mathbf{g}(S_i) = 0.1) > 0.8$, but $\mathbb{P}(\mathbf{p}(\theta; S_i) \leq 0.1) = 0$.

We design Algorithm 2 to approximate the solution to equation 3 and generate an augmented dataset $\mathbf{D}$. Algorithm 3 samples a number of θ based on $\mathbf{D}$. More implementation details can be found in Section D.

Algorithm 2 Generate adversarial dataset $\mathbf{D}$

Input: Initial model θ_0, learning rate α, dataset $\mathbf{D} = \{x_i, y_i\}_{i=1,...N}$, number of steps T_1 **Output:** dataset $\mathbf{D} = \{x_i, y_i, P_i\}_{i=1,...M}$, optimal model θ

Initialize: $\theta = \theta_0, \mathbf{D} = \{x_i, y_i, P_i\}_{i=1,...N}$ with $P_i = 1$
for $k = 1, 2, \ldots, T_1$ **do**
  Sample $\{x_i, y_i, P_i\}_{i=1,...N}$ proportionally to the weights P_i with replacement from dataset $\mathbf{D}$
  for $i = 1, 2, \ldots, N$ **do**
    $\theta \leftarrow \theta - \alpha P_i \nabla_\theta \ell(\theta; x_i, y_i)$
    Input θ, x_i, y_i to Algorithm 1 to generate attack $\{x_i', y_i\}$
    append $\{x_i', y_i, P_i\}$ to dataset $\mathbf{D}$ with weight $P_i = (k-1)N + i$
  end for
end for

Algorithm 3 DRO training with a given adversarial dataset

Input: Initial model θ_0, learning rate α, adversarial dataset $\mathbf{D} = \{x_i, y_i, P_i\}_{i=1,...M}$, number of steps T_2
Output: DRO trained model: θ

Initialize: $\theta = \theta_0$
for $k = 1, 2, \ldots, T_2$ **do**
  for $i = 1, 2, \ldots, M$ **do**
    Sample $\{x_i, y_i\}$ proportionally to the weights P_i with replacement from dataset $\mathbf{D}$
    $\theta \leftarrow \theta - \alpha P_i \nabla_\theta \ell(\theta; x_i, y_i)$
  end for
end for

The intuition behind Algorithm 2 is that the outer loop chooses batches of size N and the batches are sampled biased towards recent iterations. In turn, adversarial examples are added in the inner loop corresponding to the current optimization model parameters, which are updated according to standard stochastic gradient descent. The overall result is similar to a two-time-scale stochastic approximation algorithm, (Borkar, 1997), which will be analyzed in future work. We use Algorithm 2 to generate a dataset $\mathbf{D}$ of optimal adversarial perturbation corresponding to the optimal solution of equation 3. Then, we run Algorithm 3 with dataset $\mathbf{D}$ 50 times to sample 50 models in the model space. We use a significance level of 0.05 in the following tests.

Given all the images and one pretrained model θ_0, we test θ_0 on all images and compute accuracies by groups. We denote the groups as $\{\mathbf{g}_i, \mathbf{p}_i\}$, where $\mathbf{g}_i$ is the per capita GDP of ith country in log scale and $\mathbf{p}_i$ is the accuracy of classifying the images in ith country. We assume that there exists a postive correlation between $\mathbf{g}$ and $\mathbf{p}$. We also assume that the error in accuracy of each country is negatively related to the number of images in the country, so we

write a covariance matrix as $\mathbf{\Sigma}_{ii} = 1/\sqrt{n_i}$, where n_i is the number of images in ith country. Then we run generalized least square (GLS) with $\mathbf{\Sigma}$ on data $\{\mathbf{g}_i, \mathbf{p}_i\}$ and obtain a linear estimator $\mathbf{p} = \beta\mathbf{g} + \epsilon$.

Given the pretrained model θ_0 and computed linear regression model $\mathbf{p} = \beta_0\mathbf{g} + \epsilon$ (see Figure 8a), we hypothesis test whether there exists exists significant linear relationship:

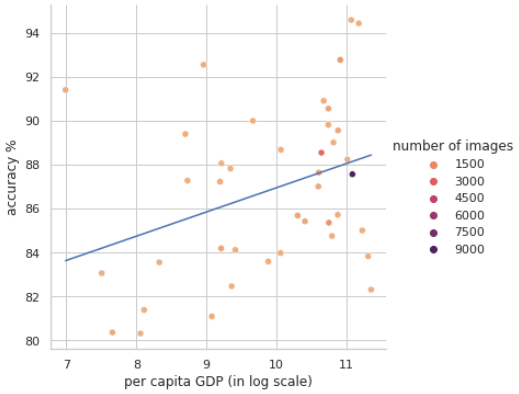
H_0: $\beta_0 = 0$. versus H_1: $\beta_0 \neq 0$.

An F-test on linear regression models test whether $\mathbf{g}$ depends on $\mathbf{p}$ (Hahs-Vaughn & Lomax, 2020). We compute $F_0 = 5.392$ with degrees of freedom $\nu_1 = 1, \nu_2 = 39$ and $\mathbb{P}(f > F_0) = 0.02554$. Thus, we can reject the null hypothesis and conclude that there exists significant linear relationship between $\mathbf{g}$ and $\mathbf{p}$. For this model, we know it is not group-fair.

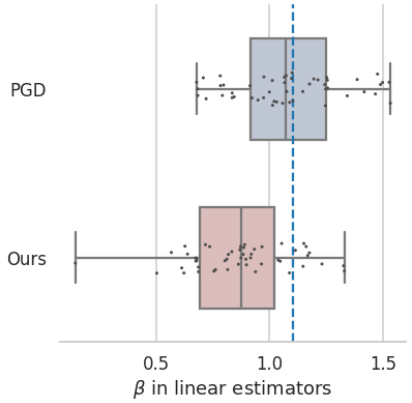
After we train 50 models and compute 50 linear estimators with our method and PGD method respectively, we use a standard t-test to compare the two means of betas (Sheynin, 1995) to compare the magnitude of unfairness. We denote β_m as our model's mean and β'_m as PGD model's mean. This corresponds to hypothesis testing

H_0: $\beta_m \geq \beta'_m$ versus H_1: $\beta_m < \beta'_m$.

We compute t-value is 3.852 with a probability 0.00017, so we can reject the null hypothesis and conclude that $\beta_m < \beta'_m$.



The line shows a linear regression model with independent variable $\mathbf{g}$ and dependent variable $\mathbf{p}$. Each dot represents one country and the color of dots denote the number of images in this country.



The two boxes each shows the distribution of β obtained from PGD method and our method respectively. The blue dashed line represents β_0 of the pretrained model. Both methods reduce the correlation between $\mathbf{g}$ and $\mathbf{p}$. Training with our attack method improves fairness more than PGD method.

5 Conclusion

We combine cost functions introduced from image quality assessment work with the DRO framework, and design two DRO training algorithms to efficiently sample a number of models. We show that our adversarial attack method can generate successful and the least human perceptible attacks, comparing with other SOTA methods. For specific ImageNet datasets, we test the existence of inherent unfairness, such as geo-location biases. After testing two collections of models that are respectively trained by DRO algorithm with our attack method and with PGD attack method, our method improves fairness more significantly than PGD method. Our hypothesis tests provide a general framework to test group fairness on the space of datasets and models based on the datasets. The limitation of our method is that we do not have enough computational resources or data to sample datasets, and we can only condition on a given dataset and randomize the models and attacks. With generating a variation of adversarial attacks, our training process mitigates the biases in the given dataset. We hope future work will incorporate the randomness in datasets and conduct the complete test in fairness. Given our results of human imperceptible adversarial attacks, we believe future work can design methods to detect imperceptible attacks, so humans can recognize situations when imperceptible attacks are applied. We also hope our work can help understand the gap between machine perception and human perception, and bridge the two areas of adversarial attacks and fairness in machine learning.

REFERENCES

Naveed Akhtar and Ajmal Mian. Threat of adversarial attacks on deep learning in computer vision: A survey. *Ieee Access*, 6:14410–14430, 2018.

Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square attack: a query-efficient black-box adversarial attack via random search. In *European Conference on Computer Vision*, pp. 484–501. Springer, 2020.

Amir Beck. *Introduction to nonlinear optimization: Theory, algorithms, and applications with MATLAB*. SIAM, 2014.

Jose Blanchet and Yang Kang. Sample out-of-sample inference based on wasserstein distance. *Operations Research*, 2021.

Jose Blanchet and Karthyek Murthy. Quantifying distributional model risk via optimal transport. *Mathematics of Operations Research*, 44(2):565–600, 2019.

Jose Blanchet, Karthyek Murthy, and Fan Zhang. Optimal transport based distributionally robust optimization: Structural properties and iterative schemes. *arXiv preprint arXiv:1810.02403*, 2018.

Jose Blanchet, Yang Kang, and Karthyek Murthy. Robust wasserstein profile inference and applications to machine learning. *Journal of Applied Probability*, 56(3):830–857, 2019a.

Jose Blanchet, Yang Kang, Karthyek Murthy, and Fan Zhang. Data-driven optimal transport cost selection for distributionally robust optimization. In *2019 Winter Simulation Conference (WSC)*, pp. 3740–3751, 2019b. doi: 10.1109/WSC40007.2019.9004785.

Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *Advances in neural information processing systems*, 29:4349–4357, 2016.

Vivek S Borkar. Stochastic approximation with two time scales. *Systems & Control Letters*, 29(5):291–294, 1997.

Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, pp. 77–91. PMLR, 2018.

Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186, 2017.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 39–57. IEEE, 2017.

Anirban Chakraborty, Manaar Alam, Vishal Dey, Anupam Chattopadhyay, and Debdeep Mukhopadhyay. Adversarial attacks and defences: A survey. *arXiv preprint arXiv:1810.00069*, 2018.

N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pp. 886–893 vol. 1, 2005. doi: 10.1109/CVPR.2005.177.

Nilaksh Das, Madhuri Shanbhogue, Shang-Tse Chen, Fred Hohman, Siwei Li, Li Chen, Michael E Kounavis, and Duen Horng Chau. Shield: Fast, practical defense and vaccination for deep learning using jpeg compression. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 196–204, 2018.

Terrance de Vries, Ishan Misra, Changhan Wang, and Laurens van der Maaten. Does object recognition work for everyone? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 52–59, 2019.

Keyan Ding, Kede Ma, Shiqi Wang, and Eero P. Simoncelli. Image quality assessment: Unifying structure and texture similarity. *CoRR*, abs/2004.07728, 2020. URL `https://arxiv.org/abs/2004.07728`.

Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9185–9193, 2018.

Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4312–4321, 2019.

Yinpeng Dong, Zhijie Deng, Tianyu Pang, Jun Zhu, and Hang Su. Adversarial distributional training for robust deep learning. In *NeurIPS*, 2020.

John C Duchi and Hongseok Namkoong. Learning models with uniform performance via distributionally robust optimization. *The Annals of Statistics*, 49(3):1378–1406, 2021.

Gintare Karolina Dziugaite, Zoubin Ghahramani, and Daniel M Roy. A study of the effect of jpg compression on adversarial images. *arXiv preprint arXiv:1608.00853*, 2016.

Gamaleldin F Elsayed, Shreya Shankar, Brian Cheung, Nicolas Papernot, Alex Kurakin, Ian Goodfellow, and Jascha Sohl-Dickstein. Adversarial examples that fool both computer vision and time-limited humans. *arXiv preprint arXiv:1802.08195*, 2018.

Logan Engstrom, Brandon Tran, Dimitris Tsipras, Ludwig Schmidt, and Aleksander Madry. Exploring the landscape of spatial robustness. In *International Conference on Machine Learning*, pp. 1802–1811. PMLR, 2019.

Peyman Mohajerin Esfahani and Daniel Kuhn. Data-driven distributionally robust optimization using the wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming*, 171(1):115–166, 2018.

William Falcon et al. Pytorch lightning. *GitHub. Note: https://github. com/PyTorchLightning/pytorch-lightning*, 3:6, 2019.

Rui Gao and Anton J Kleywegt. Distributionally robust stochastic optimization with wasserstein distance. *arXiv preprint arXiv:1604.02199*, 2016.

Rui Gao, Xi Chen, and Anton J Kleywegt. Wasserstein distributional robustness and regularization in statistical learning. *arXiv e-prints*, pp. arXiv–1712, 2017.

Boqing Gong, Fei Sha, and Kristen Grauman. Overcoming dataset bias: An unsupervised domain adaptation approach. In *NIPS Workshop on Large Scale Visual Recognition and Retrieval*, volume 3. Citeseer, 2012.

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

Jinjin Gu, Haoming Cai, Haoyu Chen, Xiaoxing Ye, Jimmy Ren, and Chao Dong. Image quality assessment for perceptual image restoration: A new dataset, benchmark and metric. *arXiv preprint arXiv:2011.15002*, 2020.

Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens Van Der Maaten. Countering adversarial images using input transformations. *arXiv preprint arXiv:1711.00117*, 2017.

Chuan Guo, Jacob Gardner, Yurong You, Andrew Gordon Wilson, and Kilian Weinberger. Simple black-box adversarial attacks. In *International Conference on Machine Learning*, pp. 2484–2493. PMLR, 2019.

Debbie L Hahs-Vaughn and Richard G Lomax. *An introduction to statistical concepts*. Routledge, 2020.

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

Warren He, James Wei, Xinyun Chen, Nicholas Carlini, and Dawn Song. Adversarial example defense: Ensembles of weak defenses are not strong. In *11th {USENIX} workshop on offensive technologies ({WOOT} 17)*, 2017.

Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. In *International Conference on Machine Learning*, pp. 2137–2146. PMLR, 2018.

Daniel Kuhn, Peyman Mohajerin Esfahani, Viet Anh Nguyen, and Soroosh Shafieezadeh-Abadeh. Wasserstein distributionally robust optimization: Theory and applications in machine learning. In *Operations Research & Management Science in the Age of Analytics*, pp. 130–166. INFORMS, 2019.

Alexey Kurakin, Ian Goodfellow, Samy Bengio, et al. Adversarial examples in the physical world, 2016.

Cassidy Laidlaw, Sahil Singla, and Soheil Feizi. Perceptual adversarial robustness: Defense against unseen threat models. In *ICLR*, 2021.

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

David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pp. 1150–1157. Ieee, 1999.

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

Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6):1–35, 2021.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: A simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32:8026–8037, 2019.

Ekta Prashnani, Hong Cai, Yasamin Mostofi, and Pradeep Sen. Pieapp: Perceptual image-error assessment through pairwise preference. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1808–1817, 2018.

Hamed Rahimian and Sanjay Mehrotra. Distributionally robust optimization: A review. *arXiv preprint arXiv:1908.05659*, 2019.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

Soroosh Shafieezadeh-Abadeh, Peyman Mohajerin Esfahani, and Daniel Kuhn. Distributionally robust logistic regression. *arXiv preprint arXiv:1509.09259*, 2015.

Shreya Shankar, Yoni Halpern, Eric Breck, James Atwood, Jimbo Wilson, and D Sculley. No classification without representation: Assessing geodiversity issues in open data sets for the developing world. *arXiv preprint arXiv:1711.08536*, 2017.

Mahmood Sharif, Lujo Bauer, and Michael K Reiter. On the suitability of lp-norms for creating and preventing adversarial examples. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1605–1613, 2018.

Oscar Sheynin. Helmert's work in the theory of errors. *Archive for History of Exact Sciences*, 49(1):73–104, 1995. ISSN 00039519, 14320657. URL `http://www.jstor.org/stable/41133999`.

Aman Sinha, Hongseok Namkoong, and John Duchi. Certifiable distributional robustness with principled adversarial training. In *International Conference on Learning Representations*, 2018. URL `https://openreview.net/forum?id=Hk6kPgZA-`.

Matthew Staib and Stefanie Jegelka. Distributionally robust optimization and generalization in kernel methods. *Advances in Neural Information Processing Systems*, 32:9134–9144, 2019.

Jiawei Su, Danilo Vasconcellos Vargas, and Kouichi Sakurai. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 23(5):828–841, 2019.

Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. In *International Conference on Learning Representations*, 2018.

Bart PG Van Parys, Peyman Mohajerin Esfahani, and Daniel Kuhn. From data to decisions: Distributionally robust optimization is optimal. *Management Science*, 67(6):3387–3402, 2021.

Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. *arXiv preprint arXiv:1805.12018*, 2018.

Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, and Eftychios Protopapadakis. Deep learning for computer vision: A brief review. *Computational intelligence and neuroscience*, 2018, 2018.

Z. Wang, E.P. Simoncelli, and A.C. Bovik. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems Computers, 2003*, volume 2, pp. 1398–1402 Vol.2, 2003. doi: 10.1109/ACSSC.2003.1292216.

Zhou Wang and Alan C Bovik. Mean squared error: Love it or leave it? a new look at signal fidelity measures. *IEEE signal processing magazine*, 26(1):98–117, 2009.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Benjamin Wilson, Judy Hoffman, and Jamie Morgenstern. Predictive inequity in object detection. *arXiv preprint arXiv:1902.11097*, 2019.

Eric Wong, Frank Schmidt, and Zico Kolter. Wasserstein adversarial examples via projected sinkhorn iterations. In *International Conference on Machine Learning*, pp. 6808–6817. PMLR, 2019.

Qiuling Xu, Guanhong Tao, Siyuan Cheng, and Xiangyu Zhang. Towards feature space adversarial attack. *arXiv preprint arXiv:2004.12385*, 2020.

Weilin Xu, David Evans, and Yanjun Qi. Feature squeezing: Detecting adversarial examples in deep neural networks. *arXiv preprint arXiv:1704.01155*, 2017.

Yaoda Xu and Maryam Vaziri-Pashkam. Limits to visual representational correspondence between convolutional neural networks and the human brain. *Nature communications*, 12(1):1–16, 2021.

Kaiyu Yang, Klint Qinami, Li Fei-Fei, Jia Deng, and Olga Russakovsky. Towards fairer datasets: Filtering and balancing the distribution of the people subtree in the imagenet hierarchy. In *Conference on Fairness, Accountability, and Transparency*, 2020. doi: 10.1145/3351095.3375709.

Xiaohui Zeng, Chenxi Liu, Yu-Siang Wang, Weichao Qiu, Lingxi Xie, Yu-Wing Tai, Chi-Keung Tang, and Alan L. Yuille. Adversarial attacks beyond the image space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Haichao Zhang and Jianyu Wang. Defense against adversarial attacks using feature scattering-based adversarial training. *Advances in Neural Information Processing Systems*, 32:1831–1841, 2019.

Lin Zhang, Lei Zhang, Xuanqin Mou, and David Zhang. Fsim: A feature similarity index for image quality assessment. *IEEE transactions on Image Processing*, 20(8):2378–2386, 2011.

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

Jia-Xing Zhao, Jiang-Jiang Liu, Deng-Ping Fan, Yang Cao, Jufeng Yang, and Ming-Ming Cheng. Egnet:edge guidance network for salient object detection. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2019.

Zhengli Zhao, Dheeru Dua, and Sameer Singh. Generating natural adversarial examples. *arXiv preprint arXiv:1710.11342*, 2017.

Zhengyu Zhao, Zhuoran Liu, and Martha Larson. Towards large yet imperceptible adversarial image perturbations with perceptual color distance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1039–1048, 2020.

Zhenglong Zhou and Chaz Firestone. Humans can decipher adversarial images. *Nature communications*, 10(1):1–9, 2019.

Mingkang Zhu, Tianlong Chen, and Zhangyang Wang. Sparse and imperceptible adversarial attack via a homotopy algorithm. *arXiv preprint arXiv:2106.06027*, 2021.

A METHOD

To measure the differences between two images, Wang et al. (2004) defines the SSIM function as a product of three components: the luminance l, the contrast c, and structure s. More precisely, we have

$$\text{SSIM}(\boldsymbol{x}, \boldsymbol{x}') = l(\boldsymbol{x}, \boldsymbol{x}')c(\boldsymbol{x}, \boldsymbol{x}')s(\boldsymbol{x}, \boldsymbol{x}'), \tag{7}$$

where the formulae for the three components are:

$$\begin{aligned} l(\boldsymbol{x}, \boldsymbol{x}') &= \frac{2\mu_x\mu_{x'} + C_1}{\mu_x^2 + \mu_{x'}^2 + C_1}, \\ c(\boldsymbol{x}, \boldsymbol{x}') &= \frac{2\sigma_x\sigma_{x'} + C_2}{\sigma_x^2 + \sigma_{x'}^2 + C_2}, \\ s(\boldsymbol{x}, \boldsymbol{x}') &= \frac{\sigma_{x'} + C_3}{\sigma_x\sigma_{x'} + C_3}. \end{aligned}$$

When implementing SSIM, Wang et al. (2004) shifts a 11×11 window with Gaussian weights $\boldsymbol{w} = \{w_i | i = 1, ..., N\}$ over two images $\boldsymbol{x}, \boldsymbol{x}'$ simultaneously and compute the local statistics $\mu_x, \sigma_x, \sigma_{xx'}$ as:

$$\begin{aligned} \mu_x &= \sum_{i=1}^N w_i x_i, \mu_{x'} = \sum_{i=1}^N w_i x_i' \\ \sigma_x &= \left(\sum_{i=1}^N w_i (x_i - \mu_x)^2\right)^{\frac{1}{2}}, \sigma_{x'} = \left(\sum_{i=1}^N w_i (x_i' - \mu_{x'})^2\right)^{\frac{1}{2}} \\ \sigma_{xx'} &= \sum_{i=1}^N w_i (x_i - \mu_x)(x_i' - \mu_{x'}) \end{aligned}$$

After shifting over the whole image with M windows, we can compute the SSIM score as a mean of all the local SSIM scores:

$$\text{MSSIM}(\boldsymbol{x}, \boldsymbol{x}') = \frac{1}{M}\sum_{j=1}^M \text{SSIM}(\boldsymbol{x}_j, \boldsymbol{x}_j')$$

For a fixed image $\boldsymbol{x}$, we now consider the difference $d_{\boldsymbol{x}}$ brought by a small perturbation $\boldsymbol{\Delta}$, which is given by

$$d_{\boldsymbol{x}}(\boldsymbol{\Delta}) = c_0(\boldsymbol{x}, \boldsymbol{x} + \boldsymbol{\Delta}) = 1 - \text{SSIM}(\boldsymbol{x}, \boldsymbol{x} + \boldsymbol{\Delta}).$$

Since SSIM has a unique maximum at 1, we may observe that $d_{\boldsymbol{x}}(\boldsymbol{\Delta}) \geqslant 0 = d_{\boldsymbol{x}}(\mathbf{0})$, hence $\mathbf{0}$ is a local minimum and $\nabla d_{\boldsymbol{x}}(\mathbf{0}) = 0$. Let $\boldsymbol{H}_{\boldsymbol{x}}(\boldsymbol{\Delta})$ denote the Hessian matrix of $d_{\boldsymbol{x}}(\boldsymbol{\Delta})$, from Beck (2014, Thm 2.26), we know $\boldsymbol{H}_{\boldsymbol{x}}(\mathbf{0})$ is a positive semi-definite matrix. Although theoretically $\boldsymbol{H}_{\boldsymbol{x}}$ is not positive definite, we have not encountered non-invertible Hessian matrices when generating perturbations for MNIST images.

Since $d_{\boldsymbol{x}}$ is twice continuously differentiable at $\boldsymbol{\Delta} = \mathbf{0}$, we may consider its Taylor expansion at $\mathbf{0}$ and locally approximate $d_{\boldsymbol{x}}(\mathbf{0})$ by a quadratic form of $\boldsymbol{\Delta}$:

$$\begin{aligned} d_{\boldsymbol{x}}(\boldsymbol{\Delta}) &= d_{\boldsymbol{x}}(\mathbf{0}) + \nabla d_{\boldsymbol{x}}(\mathbf{0})^T\boldsymbol{\Delta} + \frac{1}{2}\boldsymbol{\Delta}^T\boldsymbol{H}_{\boldsymbol{x}}(\mathbf{0})\boldsymbol{\Delta} + O(||\boldsymbol{\Delta}||^3) \\ &= \frac{1}{2}\boldsymbol{\Delta}^T\boldsymbol{H}_{\boldsymbol{x}}(\mathbf{0})\boldsymbol{\Delta} + O(\|\boldsymbol{\Delta}\|^3). \end{aligned}$$

Assume our loss function $\ell(\theta; \boldsymbol{x}, y_0)$ is continuously differentiable at $\boldsymbol{x}$, then we may also write

$$\ell(\theta; \boldsymbol{x} + \boldsymbol{\Delta}, y) = \ell(\theta; \boldsymbol{x}, y) + \nabla_{\boldsymbol{x}}\ell(\theta; \boldsymbol{x}, y)^T\boldsymbol{\Delta} + O(\|\boldsymbol{\Delta}\|^2).$$

Then the robust surrogate loss ϕ_λ in equation 4 can be solved by

$$\begin{aligned}
\phi_\lambda(\theta; \boldsymbol{x}, y) &= \sup_{\boldsymbol{x}'} \left(\ell(\theta; \boldsymbol{x}', y) - \lambda c((\boldsymbol{x}, y), (\boldsymbol{x}', y))\right) \\
&= \sup_{\boldsymbol{\Delta}} \left(\ell(\theta; \boldsymbol{x} + \boldsymbol{\Delta}, y) - \lambda c_0(\boldsymbol{x}, \boldsymbol{x} + \boldsymbol{\Delta})\right) \\
&= \sup_{\boldsymbol{\Delta}} \left(\ell(\theta; \boldsymbol{x}, y) + \nabla_{\boldsymbol{x}} \ell(\theta; \boldsymbol{x}, y)^T \boldsymbol{\Delta} - \frac{\lambda}{2} \boldsymbol{\Delta}^T \boldsymbol{H}_{\boldsymbol{x}}(\mathbf{0}) \boldsymbol{\Delta} + O(||\boldsymbol{\Delta}||^2))\right) \\
&\approx \ell(\theta; \boldsymbol{x}, y) + \sup_{\boldsymbol{\Delta}} \left(\nabla_{\boldsymbol{x}} \ell(\theta; \boldsymbol{x}, y)^T \boldsymbol{\Delta} - \frac{\lambda}{2} \boldsymbol{\Delta}^T \boldsymbol{H}_{\boldsymbol{x}}(\mathbf{0}) \boldsymbol{\Delta}\right).
\end{aligned}$$

Setting the gradient to zero, we get $\nabla_{\boldsymbol{x}} \ell(\theta; \boldsymbol{x}, y)^T = \lambda \boldsymbol{H}_{\boldsymbol{x}}(\mathbf{0}) \boldsymbol{\Delta}$, which then gives us

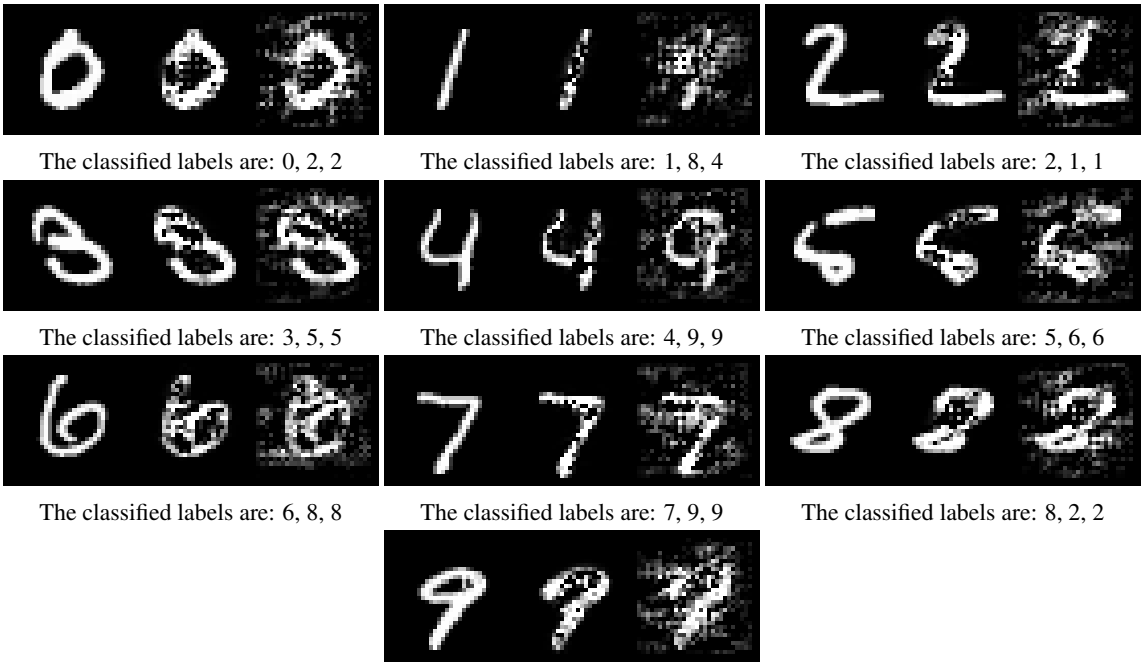
$$\boldsymbol{\Delta}^* = \frac{1}{\lambda} \boldsymbol{H}_{\boldsymbol{x}}(\mathbf{0})^{-1} \nabla_{\boldsymbol{x}} \ell(\theta; \boldsymbol{x}, y).$$

Note that $\boldsymbol{x} + \boldsymbol{\Delta}^*$ is an approximation solution to ϕ_λ since we omit the $O(||\boldsymbol{\Delta}||^2)$ term, in the final implementation the adversarial image is actually generated by

$$\boldsymbol{x}_{adv} = \boldsymbol{x} + \epsilon \frac{\boldsymbol{\Delta}^*}{\|\boldsymbol{\Delta}^*\|_2}.$$

B MORE RESULTS

In this section, we show more images that do not fit in the main text.



The classified labels are: 0, 2, 2 | The classified labels are: 1, 8, 4 | The classified labels are: 2, 1, 1

The classified labels are: 3, 5, 5 | The classified labels are: 4, 9, 9 | The classified labels are: 5, 6, 6

The classified labels are: 6, 8, 8 | The classified labels are: 7, 9, 9 | The classified labels are: 8, 2, 2

The classified labels are: 9, 4, 4

Figure 9: **Left**: original, **Middle**: one-step our method, **Right**: one-step PGD (L_2). Here are additional results of Figure 3. Each image is the first example of each digit that is successfully attacked by our method and PGD method in the MNIST test split. We show that using $1 - \text{SSIM}$ as the cost function successfully penalize any stuctural changes. On the contrary, attacks with L_2 cost function change the true meaning of the images, for example, the digit 4 attacked by PGD method on the right image looks like a 9.

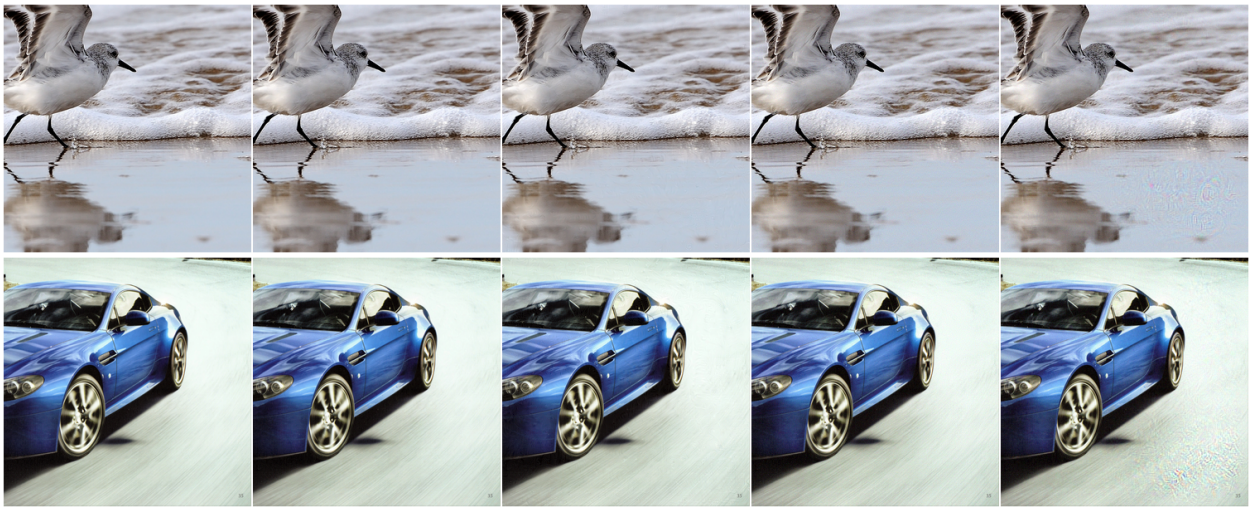


Figure 10: More qualitative results on image comparison. PerC_AL has marble effects in the water in the first image and above the car in the second image. The image quality of LPA degrades, as there are noticeable sand effects (in the water in the first image). PPGD has an area of noticeable noises in the first image.

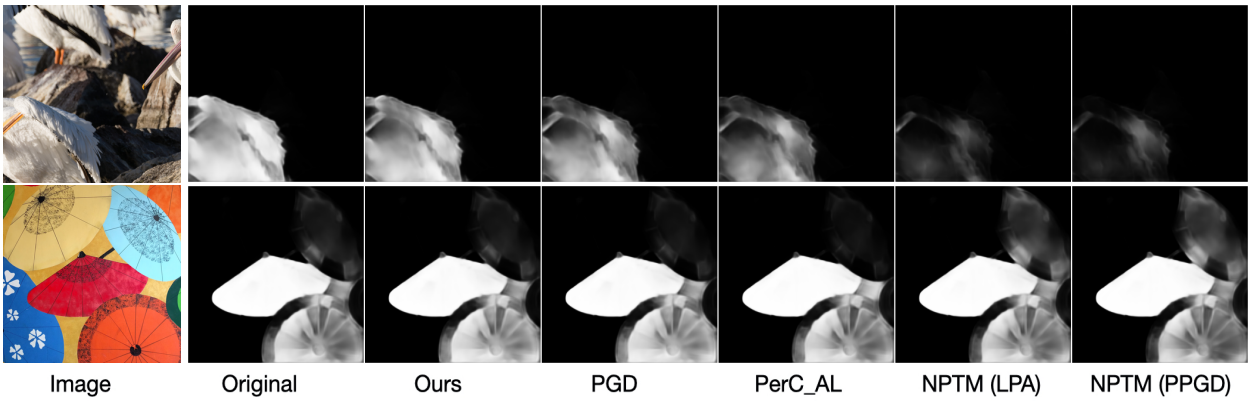


Figure 11: The comparison between the human saliency maps of the original image and adversarial attacked images. Our method's saliency maps are the most similar images to the original saliency maps in both examples.

Table 3: Comparing results with different λ. Our method with $\lambda = 1$ balances the two goals of attacking the images and minimizing the perceptual distances the best, because it has the smallest computation cost. Our method with $\lambda = 2$ minimizes DISTS distance better than with $\lambda = 1$.

	Success Rate	Time	L_1	L_2	L_∞	LPIPS (x1000)	DISTS (x1000)
Human_Perc_OT ($\lambda = 1$) (ours)	100%	2491.13s	783.855	1.905	0.006	7.303	8.165
Human_Perc_OT ($\lambda = 0.5$) (ours)	100%	2686.08s	706.974	1.755	0.006	7.443	8.465
Human_Perc_OT ($\lambda = 2.0$) (ours)	100%	2757.73s	919.670	2.194	0.006	7.356	8.138

Table 4: Comparing results with different step size. Our method with $\epsilon = 0.05$ has the smallest distance values in all distance measures but requires the most time to attack all the images.

	Success Rate	Time	L_1	L_2	L_∞	LPIPS (x1000)	DISTS (x1000)
Human_Perc_OT ($\epsilon = 0.1$) (ours)	100%	2491.13s	783.855	1.905	0.006	7.303	8.165
Human_Perc_OT ($\epsilon = 0.05$) (ours)	100%	2755.56s	782.530	1.902	0.005	7.240	8.143
Human_Perc_OT ($\epsilon = 0.20$) (ours)	100%	2444.25s	784.145	1.906	0.010	7.301	8.220

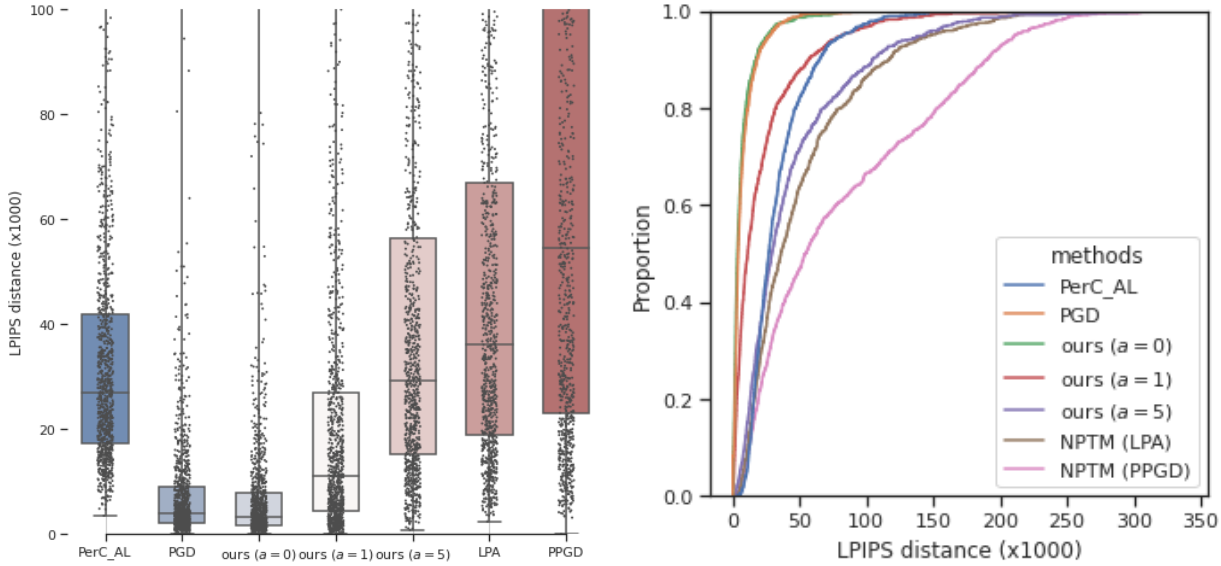


Figure 12: LPIPS distances comparison between all the attacks and the original images. Similar to Figure 5, our method with $a = 0$ has the lowest boxplot and the CDF curve enclosing the largest area. Both of the plots show that our method generates attack images with the smallest perceptual distances.

C Ablation Study

We will conduct ablation study on choices of parameters: λ, ϵ in Algorithm 1. The results are in Table 3 and Table 4. The default parameters are $\epsilon = 0.1, \lambda = 1$ and appear as the first row in both tables.

In equation 3, λ means how much penalty of cost function we add to the optimization goal. In DRO formulation, a larger λ corresponds to smaller δ and a smaller distributional neighborhood around P_0. As for Algorithm 1, with larger λ, we penalize the human perception distance more, but we may require more steps to successfully attack one image, because the direction of each step deviates from the gradient's direction more.

ϵ represents the step size of gradient descent method when attacking one image in Algorithm 1. Smaller ϵ means the final perturbation is likely to be smaller but the total time to find a successful attack is longer.

D IMPLEMENTATION DETAILS AND SPEEDUP TECHNIQUES

The algorithms are implemented with Pytorch (Paszke et al., 2019) and Pytorch Lightning (Falcon et al., 2019). Algorithm 2 is run on 4 NVIDIA V100 Tensor Core GPUs; Algorithm 1 and Algorithm 3 are run on 1 NVIDIA V100 Tensor Core GPU. See Table 5 for implementation details. When computing distances on images, we normalize the images to the $[0, 1]$ range, except that SSIM takes in images in $[0, 255]$ range.

Table 5: Implementation details in Algorithm 2 and Algorithm 3

	ImageNet
Learning rate α	0.1
Optimizer	SGD
Momentum	$5e^{-4}$
T_1, T_2	3
ϵ	0.1
ℓ	Cross entropy
λ	1

In Algorithm 1, in each step, our algorithm requires the gradient of PieAPP model with respect to images and computing the gradient is expensive. First, we sped up our algorithm by adding a early-stop mechanism, which means we will stop adding perturbation if the attack is successful. Second, we reuse the gradient of PieAPP model with respect to images $\partial c_0(\boldsymbol{x}_{adv}, \boldsymbol{x})/\partial \boldsymbol{x}_{adv}$ every 5 steps. Leveraging these two techniques, our method is comparably fast with PerC_AL, see Table 6. NPTM (PPGD) and NPTM (LPA) are both faster than our method, but they both do not reach 100% success rate and are much more perceptible in both distance measures we use.

During the DRO training procedure in Algorithm 2, we use k-means to group images of the same label into clusters. Approximately 5 images are in the same cluster. Before training starts, we run k-means clustering to establish the clusters and for each cluster, one image cloesest to the cluster's mean is called the cluster center. During training, on step 6 of Algorithm 2, we look up the cluster the image belongs to and load in the cluster center's attack. First we test whether the cluster center's attack is successful on the current image. If not, we will follow step 6. If one image is a cluster center, we update this image's attack every epoch. We anticipate this technique will degrade the quality of attacks severely.

We include the links to the methods that we use or compare with:

1. PieAPP (Prashnani et al., 2018)[2]
2. LPIPS (Zhang et al., 2018)[3]
3. DISTS (Ding et al., 2020)[4]
4. EGNet (Zhao et al., 2019)[5]

E IMAGENET WITH GEO-LOCATION DATASET

From the images of 1000 classes from ILSVRC2012 (Russakovsky et al., 2015), we select the subset of images that are from Flickr, and we use the Flickr's API[6] to obtain the geo-information of each corresponding image. Using this geo-information (latitude and longitude coordinates), we retrieve the country information of this image. The dataset has 103995 images in total, we split the datasets into 60% train split, 20% validation split, and 20% test split.

In the dataset, the images are collected from 207 countries. We put a pie chart Figure 13 to demonstrate the distribution of images' geo-locations. When we conduct the hypothesis testings in Section 4, we also filter out the countries with less than 50 images. We download all the countries' per capita GDP from the World Bank website[7].

[2]https://github.com/prashnani/PerceptualImageError
[3]https://github.com/richzhang/PerceptualSimilarity
[4]https://github.com/dingkeyan93/DISTS
[5]https://github.com/JXingZhao/EGNet
[6]https://www.flickr.com/services/api/
[7]`https://data.worldbank.org/indicator/NY.GDP.PCAP.CD`.

Table 6: Total time to attack 1000 images.

	Time (s)
PerC_AL	2386.77
PGD (L_2)	73.63
NPTM (PPGD)	281.86
NPTM (LPA)	551.87
Human_Perc_OT ($a = 0$) (ours)	2491.13

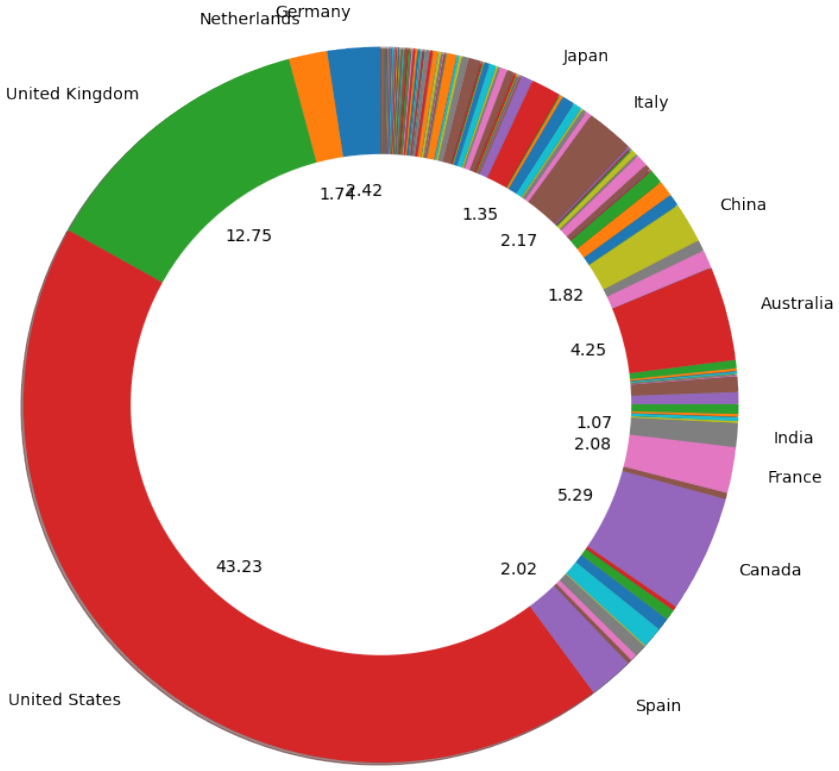


Figure 13: This pie chart shows the percentage of images coming from each country, which has a similar distribution as collected in Shankar et al. (2017). For a clearer visualization, we unfortunately cannot fit the country's names and percentage if it has < 1000 images.

Our Algorithm 2 with $T_1 = 3$ takes about 9 hours and Algorithm 3 with $T_2 = 3$ 50 times takes about 25 hours.