

Can perceptual similarity metrics be used to compare adversarial attacks?

Grigor Bezirganyan*, Aix-Marseille University

December 7th, 2023

Adversarial attacks in computer vision are often bound in ℓ_p distance. This precludes comparing these techniques with Wasserstein-based attacks. After introducing some classical perceptual metrics, we investigate empirically whether LPIPS can be used for this purpose and find that it cannot.

In Computer Vision an *adversarial sample* is defined as a perturbed image $\mathbf{x}' \in \mathbb{R}^{n \times n}$ constructed from the original image $\mathbf{x} \in \mathbb{R}^{n \times n}$, such that the perturbed image is classified differently from the original image by a model $F: \mathbb{R}^{n \times n} \rightarrow \mathbb{N}$, but it is still classified correctly by a human. The problem of crafting adversarial samples can thus be formulated as finding a perturbation $\mathbf{x} \mapsto \eta(\mathbf{x})$ such that $F(\mathbf{x}) \neq F(\mathbf{x} + \eta(\mathbf{x}))$, but which is imperceptible by humans, defined as being smaller than some $\epsilon > 0$ in a given norm.

The most common norm taken is ℓ_p [GSS, KGBa], i.e. $\|\eta\|_p \leq \epsilon$. However, as [SBR] show, ℓ_p norms are not suitable for crafting and comparing adversarial attacks, as it is possible to craft samples that have a small ℓ_p distance from the source image, but are perceptually very different, and vice versa. To remedy this, subsequent works suggest other norms. [WSK] propose the Wasserstein distance, which restricts the amount of pixel mass that can be moved to get one image from the other. Later, [WWY] and [HSSY] independently proposed improvements to the original idea, to make it stronger and faster.¹ Note however, that while we can compare adversarial attacks using ℓ_p distances with each other, or different attacks based on Wasserstein distance, it is not possible to directly compare ℓ_p attacks with non- ℓ_p ones in terms of the amount of perturbation.

Of course one can use misclassification, but since the goal of the adversarial attacks is to stay unnoticeable for humans, it would be very convenient to have a similarity metric that aligns with human judgment. Here, we discuss usage of so-called **perceptual similarity metrics** for comparing adversarial attacks, and show that they are not suitable because they can be fooled by adversarial samples.

1 Perceptual Similarity Metrics

As motivation for the metric we use, we quickly recall first how three classical similarity metrics are built. The first, SSIM aggregates rough pixel information globally. The second, MSSIM, does so in patches to allow for some locality. The third, FSIM, looks at engineered low-level features. After these, a natural step presents itself: to use *learned features*, as does LPIPS, or, going a bit further, an ensemble of them, in E-LPIPS.

1.1 (Mean) Structural Similarity Index (SSIM)

One of the most popular perceptual similarity metrics is the **Structural Similarity Index** (SSIM) [WBSS]. Because the human visual system (HVS) attends

Contents

1 Perceptual Similarity Metrics	1
1.1 (Mean) Structural Similarity Index (SSIM)	1
1.2 Features Similarity Index Matrix (FSIM)	2
1.3 Learned Perceptual Image Patch Similarity (LPIPS)	2
2 Problem Setting	3
3 Experimental setup	4
4 Results	5
4.1 Measuring effectiveness of attack transfers	5
4.2 Can we use LPIPS to compare Wasserstein and ℓ_p attacks?	6
5 Conclusions	8
Appendix A Comparing FW+Dual LMO and IIW attacks	8
Appendix B Bi-modality in the distance distributions	9
Bibliography	10

¹ To the best of our knowledge there is no comparison between these two methods, and at the end of the article we will provide one.

*. Work done while interning at the appliedAI Initiative GmbH during Fall 2022.

to the structural information in a scene, SSIM separates the task of similarity measurement into three comparisons: *luminance*, *contrast*, and *structure*.

The means of pixel values μ_x and μ_y are used to assess the similarity in luminance between the two images, while the standard deviations σ_x and σ_y measure contrast, and the covariance σ_{xy} represents structural differences. The comparison function is a function of these quantities, which under some simplifying assumptions becomes:

$$\text{SSIM}(\mathbf{x}, \mathbf{y}) := \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}, \quad (1)$$

² These are included to avoid a division by zero or by a very small denominator, and are usually defined in terms of the dynamic range of the images and small tunable constants.

where $C_1, C_2 > 0$ are small stabilizing terms.² The authors observe that this kind of *global* similarity measurement may not work well, and they suggest applying the function locally in different regions of the image and then compute the mean. This is known as the **Mean Structural Similarity Index** or MSSIM.

$$\text{MSSIM}(\mathbf{X}, \mathbf{Y}) := \frac{1}{M} \sum_{j=1}^M \text{SSIM}(\mathbf{x}_j, \mathbf{y}_j). \quad (2)$$

1.2 Features Similarity Index Matrix (FSIM)

[ZZMZ] argue that SSIM has the deficiency that all pixels have the same importance, but the HVS attributes different importance to different regions of an image. The authors argue that the HVS mostly pays attention to low-level features, such as edges or other zero crossings, and suggest comparing two sets of features:

³ According to the authors: *Based on the physiological and psychophysical evidences, the PC theory provides a simple but biologically plausible model of how mammalian visual systems detect and identify features in an image. PC can be considered as a dimensionless measure for the significance of a local structure.* [ZZMZ]

Phase Congruency (PC) PC is a measure of the alignment of phase across different scales of an image and is a feature that is considered to be invariant to changes in brightness or contrast. It is believed to be a good approximation to how the HVS detects features in an image because the latter is more sensitive to structure than to amount of light.³

The computation of phase congruency involves several steps (for details, see [ZZMZ]). First a decomposition using wavelets or filter banks, then pixel-wise extraction of phase information, followed by a comparison of the phases.

Gradient magnitude (GM) Image gradient computation is a cornerstone of image processing, e.g. for edge detection (high luminance gradients). The GM of image $f(\mathbf{x})$ is the Euclidean norm of the gradient: $G(\mathbf{x}) := \|\nabla f(\mathbf{x})\|_2$.⁴

⁴ To make notation more explicit, we consider images as maps from coordinates \mathbf{x} to channels $f(\mathbf{x}) = \mathbf{x}$.

For color images, PC and GM features are computed from their luminance channels. Given images f_i and corresponding features GM_i and PC_i , $i = 1, 2$, S_{PC} , the harmonic mean of the PCs, and S_{GM} , the harmonic mean of the GMs are combined into the **Features Similarity Index score**:

$$S_L(\mathbf{x}) := [S_{PC}(\mathbf{x})]^\alpha [S_{GM}(\mathbf{x})]^\beta, \quad (3)$$

where α, β are parameters to tune the effect of PC and GM features.

1.3 Learned Perceptual Image Patch Similarity (LPIPS)

[ZIE+] choose a different strategy for computing similarity. They argue that internal activations of neural networks trained for high-level classification tasks, even across network architectures and without further calibration, correspond to human perceptual judgments.

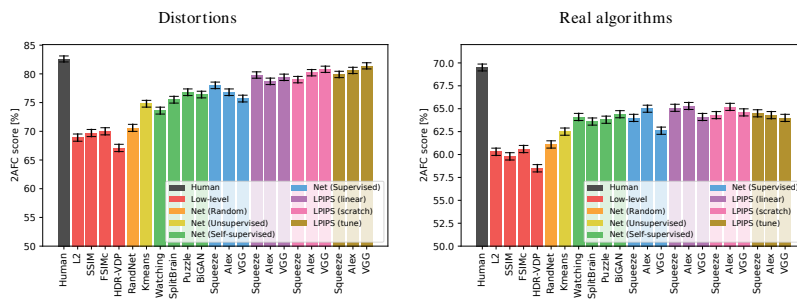
To get the similarity metric between two images \mathbf{x}, \mathbf{x}_0 with a neural network F , they extract features from L layers and unit-normalize in the channel dimension. For layer l those become \hat{y}^l, \hat{y}_0^l . Then, they scale the activations channel-wise with vectors $w^l \in \mathbb{R}^{C_l}$ and compute the ℓ_2 distance between the respective activations for each image. Finally, they compute the spatial average and sum channel-wise. The complete metric is:

$$d_{\text{LPIPS}}(\mathbf{x}, \mathbf{x}_0) := \sum_l \frac{1}{H_l W_l} \sum_{h,w} |w_l \odot (\hat{y}_{hw}^l - \hat{y}_{0hw}^l)|_2^2. \quad (4)$$

The weights w_l are optimized such that the metric best agrees with human judgment derived from two-alternative forced choice (2AFC) test results.⁵ In the 2AFC task an image is presented to a human subject together with two distorted versions of the source image. The goal is to choose which of the altered images is closer to the original.

After tuning LPIPS, the authors evaluate several similarity metrics (including SSIM and FSIM) on a separate dataset and compute agreement of the algorithm with all of the judgments. To aggregate information, if p fraction of the humans vote for image 1 and $1 - p$ fraction vote for image 2, the human will get a score of $p^2 + (1 - p)^2$.

⁵ A 2AFC test is a psychophysical method used to measure an individual's perception or to make decisions between two alternatives under forced conditions. It's commonly used in sensory testing, where a subject is presented with two stimuli and is required to choose one according to a specific criterion.



In Figure 1, we see that LPIPS has a higher 2AFC score than the SSIM or FSIM measures or the ℓ_2 distance.

However, a problem with the LPIPS distance is that since it uses activations of Deep Neural Networks, it is prone to adversarial attacks itself. Hence, [KHL] introduced the E-LPIPS or **Ensembled LPIPS**. They transform both input images using simple random transformations and define:

$$d_{\text{E-LPIPS}}(\mathbf{x}, \mathbf{y}) := \mathbb{E}[d_{\text{LPIPS}}(T(\mathbf{x}), T(\mathbf{y}))], \quad (5)$$

where the expectation is taken over a family of image perturbations. The authors claim that the E-LPIPS model is more robust but has the same prediction power.

2 Problem Setting

As previously discussed, our main goal is to determine whether perceptual metrics can be used to compare adversarial attacks in image classification. We focus on LPIPS for its popularity and because it more closely aligns with human ratings than other ones (Figure 1).

[KHL] showed that LPIPS is sensitive to adversarial samples, but used attacks specifically targeting it. If generic adversarial attacks for image classification transferred poorly to an LPIPS network, then LPIPS could be considered for perceptual evaluation of adversarial attacks.

Consider a target network F_T , and an attack $\mathbf{x} \mapsto \mathbf{x}' := \mathbf{x} + \eta(\mathbf{x})$ perturbing \mathbf{x} by some η s.t. $\|\mathbf{x} - \mathbf{x}'\| < \varepsilon$ in some norm. Note that induced by the distribution of images $\mathbf{x} \sim \mathbf{X}$, one has a distribution of perturbations $\boldsymbol{\eta} \sim \eta(\mathbf{X})$. We can define an **attack transfer** from classification to LPIPS as successful if the quantity

$$\mathbb{E}_{\mathbf{x}, \boldsymbol{\eta}}[|d_{\text{LPIPS}}(\mathbf{x}, \mathbf{x} + \eta(\mathbf{x})) - d_{\text{LPIPS}}(\mathbf{x}, \mathbf{x} + \boldsymbol{\eta})|]$$

is large enough. To compute this, we take $\mathbf{x} \in D_{\text{train}}$ and approximate $\boldsymbol{\eta}$ by independently sampling from the set $\{\eta(\mathbf{x}) : \mathbf{x} \in D_{\text{train}}\}$. Using only the first moment is however less informative than comparing the full distributions of distances, e.g. with the Wasserstein metric.⁶

That is, the classification attack transfers to LPIPS successfully when LPIPS believes the adversarial samples $\mathbf{x} + \eta(\mathbf{x})$ to be generally much closer to, or farther apart from the source images \mathbf{x} , than from randomly, but not adversarially perturbed images $\mathbf{x} + \boldsymbol{\eta}$ (although visually very similar to the adversarial samples these are not specifically designed to fool the network on a particular sample).⁷ Our hypothesis is that since LPIPS works on top of convolutional networks designed for image classification (i.e. VGG, RESNET, ALEXNET), then adversarial samples crafted for those networks will fool LPIPS as well, and hence it cannot be used to compare adversarial attacks.

3 Experimental setup

To test out the hypothesis we design the experiment following the description above. We construct adversarial samples \mathbf{x}' from source images $\mathbf{x} \in D_{\text{train}}$, and samples $\mathbf{x}'' := \mathbf{x} + \boldsymbol{\eta}$ which are visually similar to the adversarial samples but remain classified with the correct label (*fake adversaries*). Then, we will compare the distributions of the scores and if there is a significant difference in distributions, our hypothesis will be verified.⁸

We work with the IMAGENETTE dataset [AI], which is a subsample of 10 classes from the IMAGENET dataset [DDS+]. For the attacks we consider a pre-trained VGG-11 architecture [SZ] and craft adversarial samples with Projected Gradient Descent (PGD) [KGBb], Wasserstein Attack using Frank Wolfe method with Dual Linear Minimization Oracle (FW + Dual LMO) [WWY], and Improved Image Wasserstein attack (IIW) [HSSY]. We use multiple tolerance radiuses $\varepsilon \in [0, 1]$ for each method.

To create the fake adversaries we take perturbations at random from all those used for crafting the adversarial samples and add them to the source images:

$$f_i := C_{[0,1]}[\mathbf{x}_i + \eta_j], \quad (6)$$

where \mathbf{x}_i is a source image, η_j is a perturbation sampled at random from those applied to generate adversarial samples, and $C(\cdot)$ is a clipping function. We construct 10 fake adversaries for each source image, by adding 10 different perturbations from adversarial samples. A batch size of 64 is used for all the attacks. We use the LPIPS [ZIE+] metric for computing the distances with three different pre-trained backends: VGG-16, ALEXNET [KSH] and SQUEEZE NET [IHM+] (the **distance network**).⁹ The **target network** for all the experiments discussed here is only VGG-16. We use the implementations from the official PYTORCH repository [PGM+].

⁶ A simple approach here would be to use a non-parametric test like Kolmogorov-Smirnov. However, in our experiments, visual inspection and Wasserstein provided ample evidence.

⁷ Here we are glossing over the fact that $\mathbf{x} + \boldsymbol{\eta}$, where $\boldsymbol{\eta}$ is sampled independently from \mathbf{x} might not be as imperceptible a modification as $\mathbf{x} + \eta(\mathbf{x})$.

⁸ There are then three sets of images: original, adversaries, and fake adversaries. We compare the distances between first and second, to the distances between first and third set.

⁹ We chose these distance networks for consistency with the official LPIPS implementation.

The source-code¹⁰ is open-sourced and a demo is available for interactively exploring the results.¹¹

¹⁰ github.com/beziryganyan/adversarial_arena
¹¹ beziryganyan-aai-adversarial-arena-demo-main-58mbz2.streamlitapp.com

4 Results

4.1 Measuring effectiveness of attack transfers

As a baseline, we explore if samples built against the VGG-16 network transfer to LPIPS with a VGG-16 backend. Taking $\epsilon = 0.05$ (the smallest we used for the PGD attack) we obtain a misclassification rate of 0.99 on adversarial samples and 0.072 on fake adversaries (recall: perturbed but not adversarial). The LPIPS score distributions are shown in Figure 2.

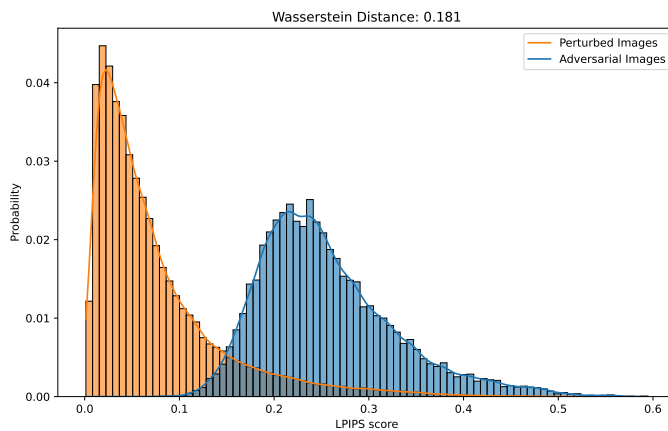


Figure 2. LPIPS score distributions for PGD attack with $\epsilon = 0.05$. Backend: VGG-16.

As expected, we see that LPIPS scores on adversarial samples tend to be higher than on fake samples, and quite unsurprisingly conclude that LPIPS metrics with a VGG-16 backend cannot be used for comparing adversarial attacks against VGG-16 networks, as the LPIPS metric itself is affected by adversarial perturbations.¹²

However, what if we use LPIPS with other backend networks? From the transferability property of adversarial attacks to different networks we hypothesize that these metrics will be affected as well. To test this, we run the same experiment using ALEXNET [KSH] and SQUEEZENET [IHM+] as backends for LPIPS. Whether or not the classification of an adversary is correct is determined by these target networks.

In Figure 3 we have the LPIPS scores with an ALEXNET backbone. We can see that the distributions are much closer and have a much smaller Wasserstein distance of 0.005, compared to the 0.181 with the VGG backend. In Figure 4 we observe a similar pattern.

From these plots we hypothesize that our initial assumption was incorrect, and the ℓ_∞ -PGD adversarial attacks do not transfer to LPIPS with these backends. However, in order to be able to use this metric for comparing PGD with Wasserstein methods, we must check transferability for the latter as well.

¹² For adversarial samples we only take into account those which were successful attacks, and for the fake adversaries only perturbed images which did not misclassify (successful fake attacks). Hence, when the cardinality of either of the sets is low, we will have lower number of samples, and hence, bigger bins for the histograms.

Figure 3. LPIPS score distributions for PGD attack with $\epsilon = 0.05$. Backend: **AlexNet**.

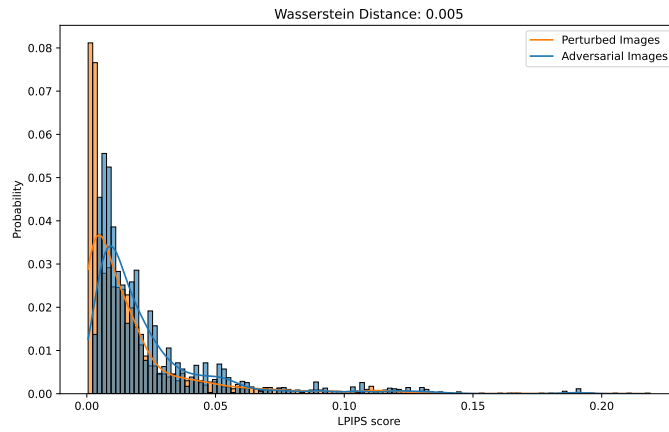
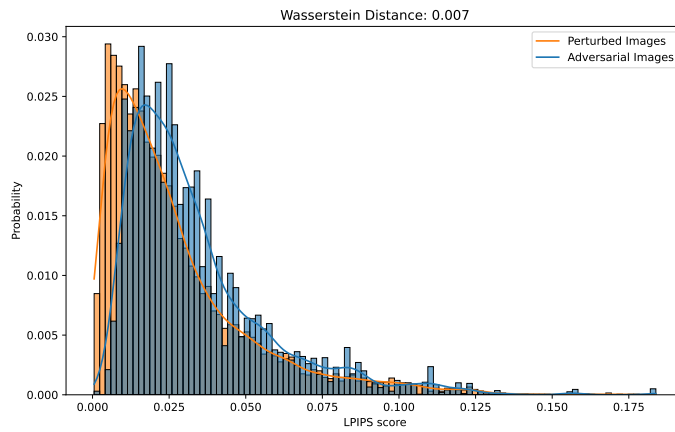


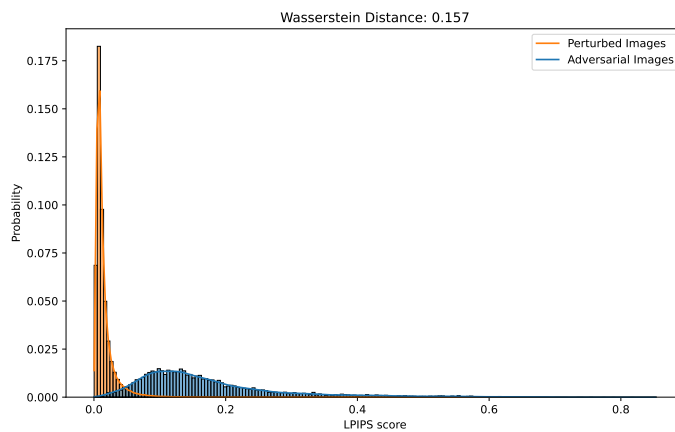
Figure 4. LPIPS score distributions for PGD attack with $\epsilon = 0.05$. Backend: **AlexNet**.



4.2 Can we use LPIPS to compare Wasserstein and ℓ_p attacks?

To make our experiments with Wasserstein attacks comparable, we use an ϵ that yields a VGG-16 misclassification rate similar to the one obtained with the PGD attack (98.94%).

Figure 5. LPIPS score distributions for FW+Dual LMO attack with $\epsilon = 0.0335$. Backend: **AlexNet**.



In Figures 5, 6, 7, and 8 we can clearly see that Wasserstein attacks on VGG-16 transfer successfully to other networks and hence affect the LPIPS score. This indicates that the LPIPS perceptual metric cannot be used for comparing Wasserstein adversarial attacks with other adversarial attack methods.

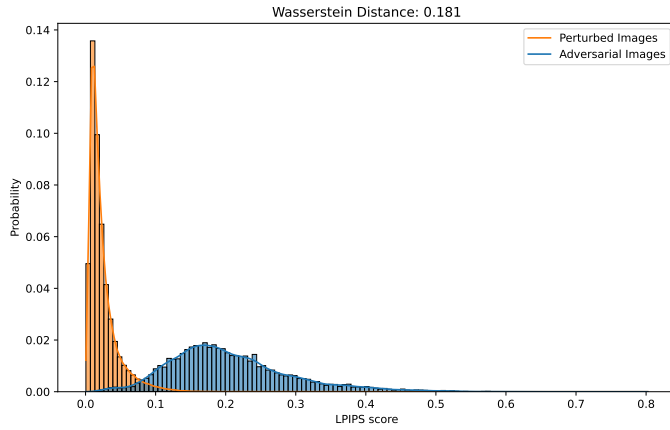


Figure 6. LPIPS scores distributions for FW+Dual LMO attack with $\epsilon = 0.0335$. Backend: SqueezeNet.

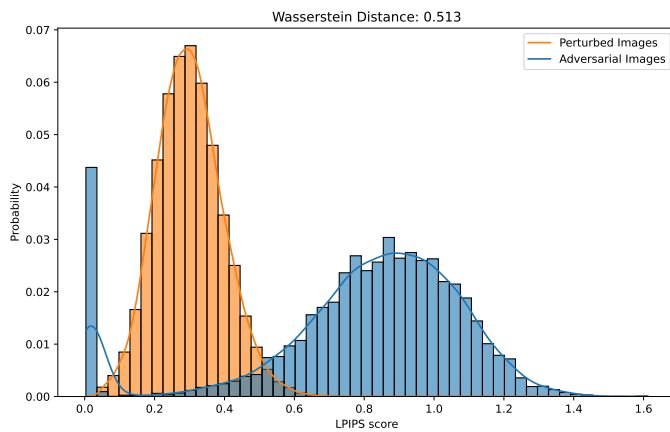


Figure 7. LPIPS scores distributions for IIW attack with $\epsilon = 0.32143$. Backend: AlexNet.

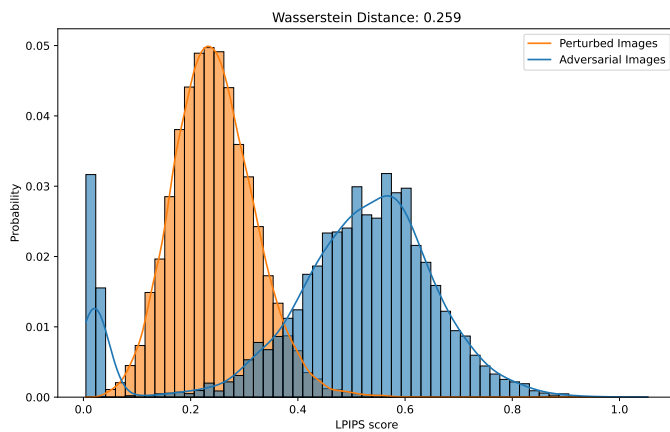


Figure 8. LPIPS scores distributions for IIW attack with $\epsilon = 0.32143$. Backend: SqueezeNet.

5 Conclusions

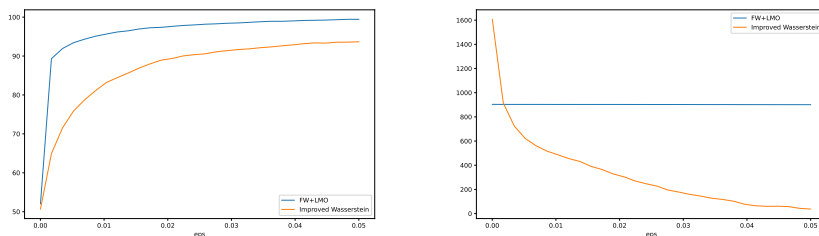
In this article we investigated whether adversarial attacks can be compared using LPIPS. We crafted adversarial samples against the VGG-16 network, and also crafted non-adversarial samples with similar perturbations, which are classified correctly. Comparing the LPIPS score distributions between the source images and the adversarial samples, and source images and the perturbed but not adversarial images, we observed that the distributions are significantly different from each another, which suggests that although the perturbations were similar, the adversarial perturbation changed the activation values in a way that makes LPIPS not suitable for comparing adversarial attacks, since it is also being fooled. Therefore, human surveys are still necessary for now.

We also saw that while for PGD attacks with ℓ_∞ norm the attack did not transfer well to LPIPS with other networks as backends, Wasserstein attacks achieved much higher transferability. This is a point in favour of Wasserstein attacks, since the attacker can create samples without knowing the underlying architecture of the target network, and still hope for good results.

A Comparing FW+Dual LMO and IIW attacks

As we mentioned earlier, to the best of our knowledge there is no comparison between the FW+Dual LMO [WWY] and IIW [HSSY] attacks, although they both are improvements of the original Wasserstein Attack [WSK]. We report in passing a quick comparison of *duration* and *misclassification rate* as a function of budget $\epsilon > 0$. We run the attacks on the same set of images from IMAGENETTE using their default parameters.

Figure 9. Misclassification rates and Time (sec/batch) of Wasserstein attacks as a function of budget ϵ .



In Figure 9 we see that FW+Dual LMO has much better misclassification rate, but IIW finds adversarial samples much faster for larger budgets, while the difference of durations is comparably very small for FW+Dual LMO attacks. On average the IIW attack is almost twice as fast (329 sec/batch) than FW+Dual LMO (902 sec/batch). From Figures 7, 8, 9, and 10, we can say that IIW attempts smaller changes to the activations, but also achieves lower misclassification rate, while FW+Dual LMO makes bigger changes and achieves a higher misclassification rate.

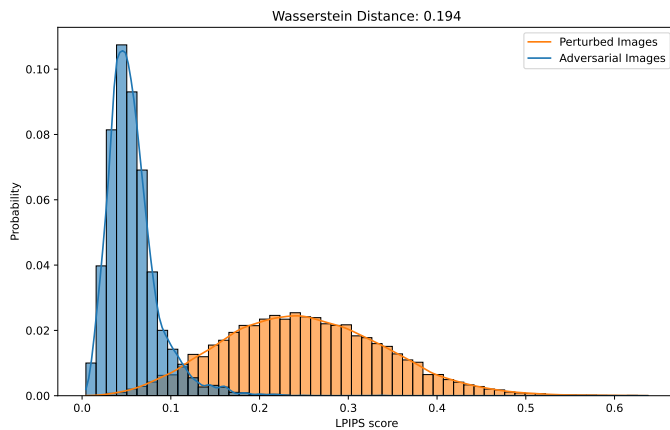


Figure 10. LPIPS scores distributions for IIW attack with $\epsilon = 0.0005$. Backend: SqueezeNet.

B Bi-modality in the distance distributions

An intriguing detail in Figures 7 and 8 is that the score distributions on adversarial images (blue) have two modes: one very close to 0, and another one beyond the mode of the fake adversaries (orange). To investigate this behaviour further, we check the score distributions of the IIW attack under different values of ϵ . For $\epsilon = 0.0005$ (Figure 10) we notice that there is only one mode for the adversarial distribution, but the scores are lower than the ones of the fake adversarial distribution.

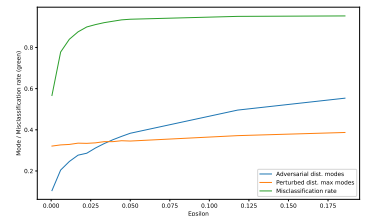


Figure 11. Fitted modes of the adversarial (blue) and perturbed (orange) distributions (using a Gaussian Mixture Model, and a single Gaussian respectively), and misclassification rate (green) for the IIW attack and VGG as LPIPS backend. While at some point the misclassification rate plateaus, the mode of the adversarial distribution keeps increasing.

This indicates that under low values of ϵ , the IIW attack manages to successfully create adversarial samples with smaller changes in the activations than similar random perturbations. When ϵ increases, the attack is allowed to add more aggressive perturbations up to the allowed budget. While for some small set of images the activations remain small, for most of them more perturbation changes the activations more, and hence, the blue distribution shifts to the right as in Figures 7 and 8. See Figure 11 for a quick empirical check.

Another check to substantiate our reasoning behind the two modes is to visually compare images from each one. In Figures 12 and 13 we see samples from both parts of the bi-modal adversarial distribution. We can see that the ones which have LPIPS distance from source images closer to the smaller mode (i.e. $d_{\text{lpiips}} < 0.2$), are almost indistinguishable from the source images, while the ones with LPIPS distance closer to the second mode (i.e. $d_{\text{lpiips}} > 0.2$) have very noticeable perturbations.



Figure 12. Adversarial samples and their source images crafted with IIW attack with $\epsilon = 0.32143$, with low LPIPS scores between the source and the resulting adversarial sample ($d_{\text{lpiips}} < 0.2$).

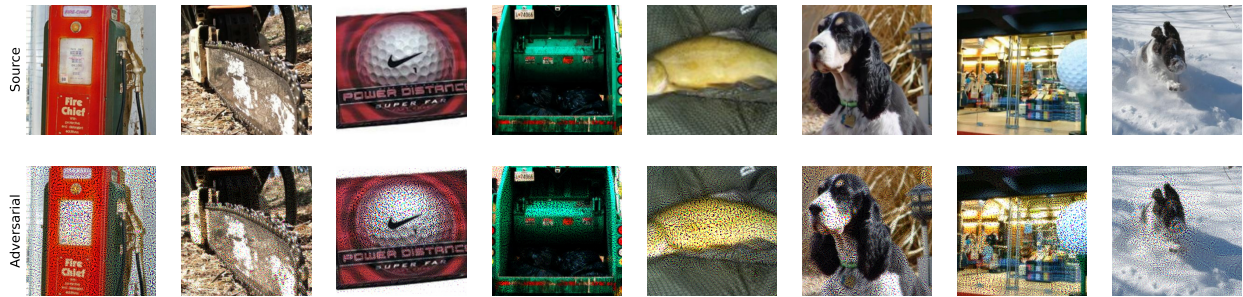


Figure 13. Adversarial samples and their source images crafted with IIW attack with $\epsilon = 0.32143$, with high LPIPS scores between the source and the resulting adversarial sample ($d_{\text{lpiPS}} \geq 0.2$).

BIBLIOGRAPHY

- [AI] Fast AI. Imagenette Dataset.
- [DDS+] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.
- [GSS] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and Harnessing Adversarial Examples.
- [HSSY] J. Edward Hu, Adith Swaminathan, Hadi Salman, and Greg Yang. Improved Image Wasserstein Attacks and Defenses.
- [IHM+] Forrest N. Iandola, Song Han, Matthew W. Moskewicz, Khalid Ashraf, William J. Dally, and Kurt Keutzer. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size.
- [KGBa] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial Machine Learning at Scale.
- [KGBb] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial examples in the physical world.
- [KHL] Markus Kettunen, Erik Härkönen, and Jaakko Lehtinen. E-LPIPS: Robust Perceptual Image Similarity via Random Transformation Ensembles.
- [KSH] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc.
- [PGM+] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library.
- [SBR] Mahmood Sharif, Lujo Bauer, and Michael K. Reiter. On the Suitability of Lp-Norms for Creating and Preventing Adversarial Examples. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1686–16868.
- [SZ] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition.
- [WBSS] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. 13(4):600–612.
- [WSK] Eric Wong, Frank R. Schmidt, and J. Zico Kolter. Wasserstein Adversarial Examples via Projected Sinkhorn Iterations.
- [WWY] Kaiwen Wu, Allen Houze Wang, and Yaoliang Yu. Stronger and Faster Wasserstein Adversarial Attacks.
- [ZIE+] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric.
- [ZZMZ] Lin Zhang, Lei Zhang, Xuanqin Mou, and David Zhang. FSIM: A Feature Similarity Index for Image Quality Assessment. 20(8):2378–2386.