ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-6654-0540-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICASSP43922.2022.9747529

IN PURSUIT OF PRESERVING THE FIDELITY OF ADVERSARIAL IMAGES

Joseph Clements and Yingjie Lao

Electrical and Computer Engineering, Clemson University
jfcleme@g.clemson.edu, ylao@clemson.edu

ABSTRACT

Adversarial examples have emerged as a severe concern for the security of neural networks. However, the ℓ_p-distances, typically used as a similarity constraint, often fail to capture human perceived similarity. Under challenging scenarios, such as attacking a defended model, this discrepancy leads to the severe degradation of image fidelity. In this paper, we find adversarial examples that better match the natural distribution of the input domain by integrating signal processing techniques into the attack framework, dynamically altering the allowed perturbation with a *Rule Adjustable Distance* (RAD_ρ). The framework allows us to easily incorporate structural similarity, Otsu's method, or variance filtering to increase the fidelity of adversarial images while still adhering to an ℓ_p-bound.

***Index Terms*—** Adversarial Examples, Visual Fidelity, Image Processing, Deep Learning, Robustness

1. INTRODUCTION

The study of machine learning, especially deep learning, has been making remarkable strides in recent years. Unfortunately, adversarial machine learning has also arisen in tandem with the field, tempering the allure of learning applications. Many attacks have been exploited, which include data poisoning attacks [1], backdoor injection [2], and training set privacy violations [3]. Of particular interest are adversarial examples, whose feasibility and effectiveness in compromising machine learning models have been demonstrated in many practical systems [4, 5]. These are inputs to a well-trained model that have been modified while remaining aesthetically similar to a natural input. However, when both are passed to the model, the original produces the expected response, but the adversarial example induces a malicious one.

A critical metric for judging the strength of an adversarial example entails its similarity to the original input. However, the suitability of using ℓ_p-norms for adversarial images has recently been drawn into question, as the measure is inconsistent with the common understanding of psychophysical similarity (i.e., the measure of human visual perception) [6].

This work is partially supported by the National Science Foundation award 2047384.

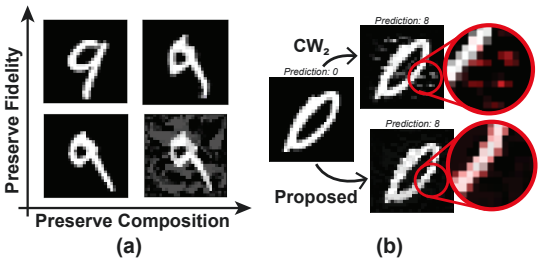

Fig. 1. Image composition and fidelity are orthogonal components of visual similarity. But, ℓ_p-based metrics well preserve input composition, not fidelity, as demonstrated in the images produced by previous methodologies. In contrast, our framework attempts to preserve both of these aspects.

In Fig. 1(a), we illustrate that adversarial similarity consists of two orthogonal aspects: the preservation of input fidelity (adherence to the characteristics of natural inputs) and input composition (coherence with the defining elements of a reference image). The ℓ_p-distances have some ability to preserve composition, as images localized around the original image tend to contain similar pixel patterns. As shown in Fig. 1(b), purely ℓ_p-based methods often synthesize an image that contains obvious visual distortions that do not exist naturally.

Previous works have generated methods of preserving visual similarity (e.g., Spacial-Chroma Shift [7], Wasserstein Attacks [8], and Perceptual Color [9]). However, such techniques cannot guarantee a bound on the distance required to achieve the adversarial image, which helps preserve input composition. Also, these methods are specific to the image domain and do not generalize to other fields. Motivated discussions on the limitations of ℓ_p-distances for adversarial example generation [6, 10], this paper proposes to develop an integrated methodology to increase the fidelity of adversarial inputs while maintaining an ℓ_p-distance guarantee.

2. SIMILARITY IN ADVERSARIAL IMAGES

2.1. Error Based Image Quality Measures

The problem of assessing image quality is considered to be synonymous with isolating and measuring an error signal in the image [11]. For example, root mean squared error (RMSE) [12] evaluates the images using the ℓ_2 distance.

However, extensive evaluations indicate that RMSE is a weak measure of psychophysical similarity [13], inspiring the development of other similarity measures. Some, including the peak signal-to-noise ratio (PSNR), are extensions of RMSE and inherit similar drawbacks [14]. Recent developments like structural similarity index metric (SSIM) [11] have arisen to better represent human perceived quality. Recently deep learning has been applied to assessing image quality [15, 16].

2.2. Adversarial Example Generation

The generation of adversarial examples can be described using the unified optimization problem articulated in [4] as:

$$\begin{array}{ll} min & \mathcal{L}(F(\mathbf{x}'), \mathbf{o}_t) \\ s.t. & S(\mathbf{x}', \mathbf{x}) < \epsilon, \end{array} \tag{1}$$

where $\mathcal{L}(\cdot,\cdot)$ is a loss function connecting the model's behavior given a modified input, $\mathbf{x}'$, and a desired target output, $\mathbf{o}_t$. $S(\cdot,\cdot)$ is a metric that measures $\mathbf{x}'$ the similarity of $\mathbf{x}'$ to the original input, $\mathbf{x}$. If it is possible to optimize $\mathbf{x}'$ such that ϵ remains small, an adversarial example is produced.

A widely used method to solve this problem is through some variations on the projected gradient descent (PGD) algorithm, which can be easily adapted to many diverse scenarios [17, 18]. The method is conducted by choosing a step size, α, and iteratively perturbed the input as:

$$\mathbf{z}'_m = \mathbf{x}'_m + \alpha \bar{\mathbf{s}}_m, \tag{2}$$

where $\bar{\mathbf{s}}_m$ is a unit vector in the direction of perturbation, i.e., the unit step, as expressed in Equation (3). This vector is generated with the gradient of a loss function, $\mathcal{L}$, and a measure of distance, $\mathcal{D}$, often the euclidean distance.

$$\bar{\mathbf{s}}_m = \underset{\mathcal{D}(\mathbf{s})=1}{-argmin} \|\nabla_{\mathbf{x}} \mathcal{L}(F(\mathbf{x}'_m), \mathbf{o}_t) - \mathbf{s}\|. \tag{3}$$

But, there is no guarantee that the intermediate step, $\mathbf{z}'_m$, fulfills the similarity constraint. So PGD projects this input back into the set of feasible inputs by solving Equation (4).

$$\mathbf{x}'_{m+1} = \underset{\mathbf{x} \in \mathcal{B}_p(\mathbf{x}_0, \epsilon)}{argmin} \|\mathbf{z}'_m - \mathbf{x}\|, \tag{4}$$

where $\mathcal{B}(\mathbf{x}_0, \epsilon)$ is a constraint ball of radius with ϵ to bound the perturbation. Most works use ℓ_p-norms for similarity, thus projecting each steps onto $\mathcal{B}_p(\mathbf{x}_0, \epsilon) = \{\mathbf{x}' \mid \|\mathbf{x}' - \mathbf{x}_0\|_p < \epsilon\}$, an ℓ_p-ball [5]. Such methods produce inputs that differ greatly from the natural inputs. So, recent works have begun expanding these concepts to alternative metrics [19, 20].

Adversarial Threat Model. The adversarial threat model considered in this work is consistent with prior results on adversarial examples [21, 22]. Diverging from these, this work assumes a not just a well-defended, but also a well-observed model. Defended models increase the required magnitude of perturbation when generating adversarial examples beyond what is necessary for undefended settings. This defensive requirement amplifies the error signal and leading to obvious input distortions that an observer can easily see.

3. THE RULE ADJUSTED DISTANCE (RAD_ρ)

To better constrain adversarial perturbations, we develop a method to dynamically redirect them based on the localized information from the input space. Thus, we introduce the *Rule Adjusted Distance* (RAD_ρ), which is defined as follows:

$$RAD_\rho(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{n=1}^{N} (\rho_n(\mathbf{x}, \mathbf{y}) |x_n - y_n|)^\infty}. \tag{5}$$

Locally, RAD_ρ mimics a weighted ℓ_∞-distance, but diverges from it, when calculated over multiple steps, with guidance from a dynamic rule set, ρ. The advantage of this approach is that we are then able to integrate RAD_ρ into the existing adversarial example frameworks, such as the PGD, simply by redefining the unit step as expressed in Equation (6).

$$\bar{\mathbf{s}}_m = \underset{RAD_\rho(\mathbf{x}'_m, \mathbf{x}'_m + \mathbf{s}) = 1}{-argmin} \|\nabla \mathcal{L}(F(\mathbf{x}'_m), \mathbf{o}_t) - \mathbf{s}\| \tag{6}$$

Because RAD_ρ locally mimics the weighted ℓ_∞-distance, we can utilize the commonly used intuitive solution to the uniform ℓ_∞-bounded problem by clipping each component of the step vector to the value $\bar{s}_{m,i} = \max(\min(s_i, \frac{1}{|\rho_i|}), \frac{-1}{|\rho_i|})$. RAD_ρ could be altered slightly to mimic another ℓ_p-distance; however, a closed-form solution to Equation (6) may not be forthcoming, so an approximate or algorithmic solution is necessary to determine the unit step. With this perspective, we can consider $\rho(\cdot,\cdot)$ as a vector-valued function that quantifies the fidelity change given a desired change in $\mathbf{x}'$. We can dynamically direct the progression of finding adversarial examples with domain-specific information using targeted signal processing techniques. We illustrate the process of defining ρ in the image domain, using three different techniques: Otsu's method, variance filtering, and regional-SSIM.

3.1. Pixel Clustering (A_o)

Otsu's Method is a thresholding technique that clusters pixels while minimizing the in-class variance between clustered pixels [23]. A_o uses this technique to define a meaningful threshold between groups of pixels and then generate ρ to discourage pixel migration across these thresholds.

For binary clustering, Otsu's method finds a value, b, that clusters $x_n \leq b$ into the first cluster, k_1, and $x_n > b$ into the second cluster, k_2, while minimizing the constraint:

$$p(x_n \in k_1)\sigma^2_{k_1} + (1 - p(x_n \in k_2))\sigma^2_{k_2}, \tag{7}$$

where $\sigma^2_{k_j}$ is the variance of cluster k_j and $p(x_n \in k_j)$ represents the probability that x_n is in cluster k_j for all $x_n \in X$. The technique can also be extended to perform multi-level thresholding for finer processing, using K clusters. Then, we can define ρ such that it penalizes gradients that attempt to push pixel values past a threshold. This ensures that the adversarial image retains the image's relative pixel distribution.

Algorithm 1 A_s: Adversarial Perturbations with SSIM

Require: $\mathbf{x}_0, \mathbf{o}_t, F(\cdot), \mathcal{L}(\cdot,\cdot), W$

1: $\mathbf{x}'_m = \mathbf{x}_0$
2: $V = \lfloor \frac{W}{2} \rfloor$
3: **while** $F(\mathbf{x}'_m) \neq \mathbf{o}_t$ **do**
4: $\quad \mathbf{g} = \frac{-\nabla \mathcal{L}(F(\mathbf{x}'_m), \mathbf{o}_t)}{\|\nabla \mathcal{L}(F(\mathbf{x}'_m), \mathbf{o}_t)\|}$
5: $\quad$ **for** $\forall r \in [1, R]$, $\forall c \in [1, C]$ and $\forall l \in [1, L]$ **do**
6: $\quad\quad \omega_1 = \mathbf{x}'_m[r - V : r + V, c - V : c + V, l]$
7: $\quad\quad \omega_2 = (\mathbf{x}'_m + \mathbf{g})[r - V : r + V, c - V : c + V, l]$
8: $\quad\quad \mathbf{SM}[r, j, c] = \text{SSIM}(\omega_1, \omega_2)$
9: $\quad$ **end for**
10: $\quad \rho = \frac{\mathbf{SM}}{\max(\mathbf{SM})}$
11: $\quad$ Solve Equation (6) for $\bar{\mathbf{s}}_m$.
12: $\quad \mathbf{z}' = \mathbf{x}'_m + \alpha \bar{\mathbf{s}}_m$
13: $\quad \mathbf{x}'_{m+1} = \underset{\mathbf{x} \in \overline{\mathcal{B}_p(\mathbf{x}_0, \epsilon)}}{argmin} \|\mathbf{z}' - \mathbf{x}\|$
14: **end while**
15: **return** $\mathbf{x}'_m$

3.2. Variance Filtering (A_v)

The second technique that we explore is based on variance filtering [24], a method for analyzing the localized pixel variation in an image. As adversarial perturbations often manifest as random noise, applying them to regions with low pixel variance can degrade image quality. In short, A_v attempts to maintain pixel variance throughout the image.

To integrate variance filtering into the proposed framework, we decompose the images into sliding windows, Ω, of width, $W = 2V + 1$. We first define a window as $\omega = \{x_{ijk} \mid i = [r - V : r + V]; j = [c - V : c + V]\}$ as the neighborhood of pixel values centered at a specific (r), column (c) and channel (k). Ω then is the collection of all such windows for a given image. Thus, $\forall \omega \in \Omega$, the average pixel value and pixel variance are found as:

$$\overline{x} = \frac{1}{W^2} \sum_{i=1}^{W^2} \omega_i, \qquad \sigma^2 = \frac{1}{W^2} \sum_{i=1}^{W^2} (\overline{x} - \omega_i)^2. \tag{8}$$

The variance map, σ^2, defines the local channel-wise pixel variance at each position in the image. Finally, we normalize the output such that $\rho_i = \frac{\sigma^2}{\max(\sigma^2)}$. The normalized variance map is then used in RAD_ρ to constrain perturbations in low variance regions to preserve the local statistical properties, and thereby the visual fidelity, of the adversarial image.

3.3. Structural Similarity Index Metric (A_s)

As discussed in the previous section, SSIM is developed to measure better the human understanding of visual similar [11]. Integrating it into the framework also could discern the natural perturbations when the psychophysical quality of the adversarial image is essential. Like A_v, A_s directs adversarial perturbations by identifying critical regions of the image and minimize perturbations of those pixels. However, it does this through the SSIM [11] which captures the concepts of contrast, luminosity, and structure, rather than pixel variance. As these properties are significant to an image, A_s tends to generate adversarial perturbations that better adhere to our understanding of visual perception.

SSIM breaks down quality assessment based on: luminosity (l), contrast (c), and structure (s), which are defined as:

$$l(\mathbf{x}, \mathbf{y}) = \frac{2\mu_\mathbf{x}\mu_\mathbf{y} + \eta}{\mu_\mathbf{x}^2 + \mu_\mathbf{y}^2 + \eta}, \quad c(\mathbf{x}, \mathbf{y}) = \frac{2\sigma_\mathbf{x}\sigma_y + \eta}{\sigma_\mathbf{x}^2 + \sigma_\mathbf{y}^2 + \eta}, \quad s(\mathbf{x}, \mathbf{y}) = \frac{\sigma_\mathbf{xy} + \eta}{\sigma_\mathbf{x}\sigma_\mathbf{y} + \eta}, \tag{9}$$

where μ_x, σ_x represent the mean and variance of image x, respectively, while σ_{xy} represents the covariance between images x and y. All three of these metrics are combined in:

$$\text{SSIM}(\mathbf{x}, \mathbf{y}) = l(\mathbf{x}, \mathbf{y})^\alpha c(\mathbf{x}, \mathbf{y})^\beta s(\mathbf{x}, \mathbf{y})^\gamma. \tag{10}$$

The variables α, β, and γ are used to balance the three base properties. We build off the sliding window method from A_v for implementing SSIM in our integrated framework. We produce two sliding windows, one for the adversarial image, Ω', and the other for the original image, Ω. Then, with the corresponding windows of both, we generate an SSIM map.

$$\mathbf{SM} = \text{SSIM}(\omega, \omega') \qquad \forall (\omega, \omega') \in (\Omega, \Omega') \tag{11}$$

We then define ρ by normalizing $\mathbf{SM}$, i.e., $\rho_n = \frac{\mathbf{SM}}{\max(\mathbf{SM})}$. Algorithm 1 demonstrates our specific implementation of A_s. A_o and A_v are implemented by replacing lines 5-10 with their respective image processing techniques.

4. EXPERIMENTAL EVALUATION

4.1. Experimental Setup

We build our experiments around the Cleverhans [25] implementation of the PGD algorithm by injecting RAD_ρ to constrain the adversarial unit step. We frequently compare our results with those generated by PGD, as it is widely used and highly analogous to our methodology. For Cifar10, we use **ResNet-20** and **WRN-40-4**. And, **ResNet-50** for ImageNet. Further, we adopt the Cifar10 (**Defended**$_\mathbf{C}$) [26] provided by Madry Labs, trained to be resistant to adversarial examples.

4.2. Directed and Natural Perturbations

We present representative samples of adversarial examples on the ImageNet [27] dataset displayed in Fig. 2. Note that it is significantly easier to produce low-perturbation adversarial examples in this undefended setting, so these examples are bounded by $\|\mathbf{x} - \mathbf{x}'\|_\infty < 0.03$. We amplify the adversarial noise by $10\times$ to more easily observe the visual distortions. We then normalize the strength of our attacks against these results by adjusting α in Equation 3 until, on average, we

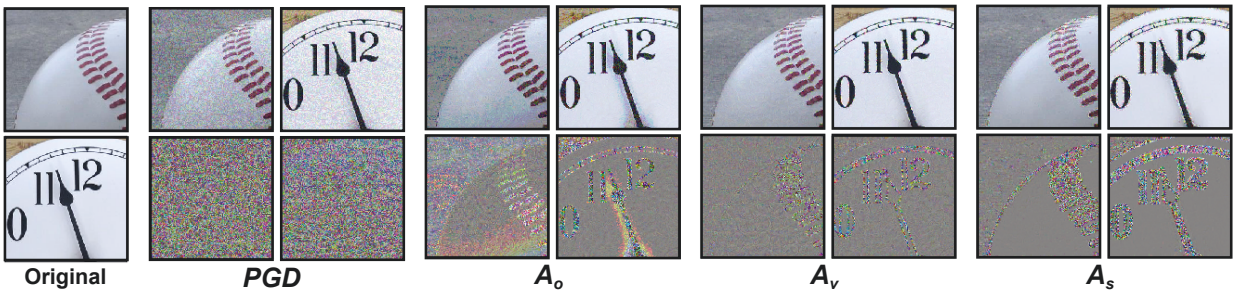

Fig. 2. Adversarial examples for the ImageNet classifier, **ResNet-50**. It can be seen that the proposed attacks are significantly better at blending the adversarial perturbation with the original image. Further, each attack's adversarial perturbations are distinctive, indicating that each captures unique visual information from its image processing techniques.

Table 1. Whitebox (Diagonals) and Blackbox (Off-Diagonals) Success Rates for Cifar10 Classifiers.

	ResNet-20				**WRN-40-4**				**Defended$_C$**			
Source Model	PGD	A_o	A_v	A_s	PGD	A_o	A_v	A_s	PGD	A_o	A_v	A_s
ResNet-20	**76.9**	**87.7**	**92.3**	**93.9**	69.7	71.3	78.2	83.8	14.9	16.0	19.9	18.8
WRN-40-4	82.0	84.9	86.9	87.9	**59.7**	**63.5**	**74.5**	**77.1**	23.4	27.8	26.9	29.7
Defended$_C$	53.7	57.3	61.6	61.5	62.4	66.9	71.1	68.3	**63.4**	**68.1**	**76.0**	**79.6**

achieve the same attack success rate in the same number of iterations. We conduct these attacks on the **ResNet-50** classifier and remark that the level of amplification used on the adversarial noise visually reflects the distortions that could be expected in a well-defended setting. For reference, we generated adversarial images using PGD. We observed during these experiments that the proposed methodology exhibits a higher success rate in both whitebox and blackbox settings at equivalent levels of image quality.

4.3. The Adversarial Accuracy and Transferability

We also evaluate the efficacy of the work under both blackbox and whitebox settings. This experiment synthesizes adversarial examples on a source model and calculates the percentage of adversarial examples that successfully fool a target model. The bold-faced diagonal entries represent the typical whitebox attack (adversarial examples are produced and intended to fool a single model). The off-diagonal entries represent the blackbox settings (adversarial examples are produced on a source model but intended to fool a target model). We produce the PGD examples with a bound specific to the scenario, and normalize our attacks by targeting the same SSIM score. Table 1 present these results on the Cifar10 classifiers. These results demonstrate that our methodology generates high-quality adversarial images with a higher success rate than traditional methodologies.

4.4. Quantitative Comparison

We further validate the fidelity of the adversarial examples produced for Cifar10 and ImageNet using image quality metrics. We also record an Opinion Score (OS), a human-centric metric, to complement these measures. Following the approach recently presented in [28], we measure OS based on the double stimulus impairment scale. OS quantifies the visual quality of an image based on the aggregated opinions of 86 human participants. This opinion score gives a subjective perspective on the quality of an adversarial example with a rating from 1 to 5, with 1 being the best score possible. From the values recorded in Table 2, it is apparent that the proposed attacks can maintain a higher level of image fidelity.

Table 2. The Visual Quality of Adversarial Examples.

Cifar10 (**Defended$_C$**)				
	PGD	A_O	A_V	A_S
Avg. SSIM	0.885	0.936	**0.980**	0.948
Avg. PSNR	32.3	35.6	**37.2**	36.3
Avg. RMSE	0.024	0.017	**0.014**	0.015
Avg. OS	3.87	2.99	**2.90**	2.92
ImageNet (**ResNet-50**)				
	PGD	A_O	A_V	A_S
Avg. SSIM	0.958	0.969	**0.975**	0.961
Avg. PSNR	22.86	24.25	**25.26**	23.20
Avg. RMSE	0.072	0.061	**0.055**	0.069
Avg. OS	2.24	1.94	**1.58**	1.67

5. CONCLUSIONS

This paper develops a methodology for preserving input composition and fidelity of adversarial examples. Our experiments demonstrate that we can integrate image processing techniques that direct perturbations to better preserve human perceived similarity.

6. REFERENCES

[1] Matthew Jagielski, Alina Oprea, Battista Biggio, Chang Liu, et al., "Manipulating machine learning: Poisoning attacks and countermeasures for regression learning," *Symposium on Security and Privacy*, pp. 19–35, 2018.

[2] Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov, "How to backdoor federated learning," *23rd International Conference on Artificial Intelligence and Statistics*, pp. 2938–2948, 2020.

[3] Nicolas Papernot, Patrick McDaniel, Arunesh Sinha, and Michael P Wellman, "Sok: Security and privacy in machine learning," *European Symposium on Security and Privacy*, pp. 399–414, 2018.

[4] Nicholas Carlini and David Wagner, "Towards evaluating the robustness of neural networks," *Symposium on Security and Privacy*, pp. 39–57, 2017.

[5] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu, "Towards deep learning models resistant to adversarial attacks," *6th International Conference on Learning Representations*, 2018.

[6] Mahmood Sharif, Lujo Bauer, and Michael K Reiter, "On the suitability of lp-norms for creating and preventing adversarial examples," *Computer Vision and Pattern Recognition Workshops*, pp. 1605–1613, 2018.

[7] Ayberk Aydin, Deniz Sen, Berat Tuna Karli, Oguz Hanoglu, and Alptekin Temizel, "Imperceptible adversarial examples by spatial chroma-shift," *arXiv preprint arXiv:2108.02502*, 2021.

[8] Kaiwen Wu, Allen Wang, and Yaoliang Yu, "Stronger and faster wasserstein adversarial attacks," in *International Conference on Machine Learning*. PMLR, 2020, pp. 10377–10387.

[9] 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*, 2020, pp. 1039–1048.

[10] Can Kanbak, Seyed-Mohsen Moosavi-Dezfooli, and Pascal Frossard, "Geometric robustness of deep networks: analysis and improvement," *Computer Vision and Pattern Recognition*, pp. 4441–4449, 2018.

[11] Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero Simoncelli, et al., "Image quality assessment: from error visibility to structural similarity," *Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[12] James Mannos and David Sakrison, "The effects of a visual fidelity criterion of the encoding of images," *Transactions on Information Theory*, vol. 20, no. 4, pp. 525–536, 1974.

[13] Zhou Wang and Alan C Bovik, "Mean squared error: Love it or leave it? a new look at signal fidelity measures," *Signal Processing Magazine*, vol. 26, no. 1, pp. 98–117, 2009.

[14] Umme Sara, Morium Akter, and Mohammad Shorif Uddin, "Image quality assessment through fsim, ssim, mse and psnr—a comparative study," *Journal of Computer and Communications*, vol. 7, no. 3, pp. 8–18, 2019.

[15] Dounia Hammou, Sid Ahmed Fezza, and Wassim Hamidouche, "Egb: Image quality assessment based on ensemble of gradient boosting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 541–549.

[16] Sharib Ali, Felix Zhou, Adam Bailey, Barbara Braden, James E East, Xin Lu, and Jens Rittscher, "A deep learning framework for quality assessment and restoration in video endoscopy," *Medical Image Analysis*, vol. 68, pp. 101900, 2021.

[17] Yan Jiang, Guisheng Yin, Ye Yuan, and Qingan Da, "Project gradient descent adversarial attack against multisource remote sensing image scene classification," *Security and Communication Networks*, vol. 2021, 2021.

[18] Oliver Bryniarski, Nabeel Hingun, Pedro Pachuca, Vincent Wang, and Nicholas Carlini, "Evading adversarial example detection defenses with orthogonal projected gradient descent," *arXiv preprint arXiv:2106.15023*, 2021.

[19] Bo Luo, Yannan Liu, Lingxiao Wei, and Qiang Xu, "Towards imperceptible and robust adversarial example attacks against neural networks," *32nd Association for the Advancement of Artificial Intelligence*, pp. 1652–1659, 2018.

[20] Eric Wong, Frank R Schmidt, and J Zico Kolter, "Wasserstein adversarial examples via projected sinkhorn iterations," *36th International Conference on Machine Learning*, pp. 6808–6817, 2019.

[21] Nicholas Carlini, Guy Katz, Clark Barrett, and David L Dill, "Provably minimally-distorted adversarial examples," *arXiv preprint arXiv:1709.10207*, 2017.

[22] Yonggang Zhang, Xinmei Tian, Ya Li, Xinchao Wang, and Dacheng Tao, "Principal component adversarial example," *Transactions on Image Processing*, vol. 29, pp. 4804–4815, 2020.

[23] Nobuyuki Otsu, "A threshold selection method from gray-level histograms," *Transactions on Systems, Man, and Cybernetics: Systems*, vol. 9, no. 1, pp. 62–66, 1979.

[24] Grzegorz Sarwas and Sławomir Skoneczny, "Object localization and detection using variance filter," *Image Processing and Communications Challenges*, vol. 313, pp. 195–202, 2015.

[25] Nicolas Papernot, Fartash Faghri, Nicholas Carlini, Ian Goodfellow, et al., "Technical report on the cleverhans v2.1.0 adversarial examples library," *arXiv preprint arXiv:1610.00768*, 2018.

[26] Dimitris Tsipras and Aleksandar Makelov, "Cifar10 adversarial examples challenge," 2017.

[27] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," *Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

[28] Sid Ahmed Fezza, Yassine Bakhti, Wassim Hamidouche, and Olivier Déforges, "Perceptual evaluation of adversarial attacks for CNN-based image classification," *11th International Conference on Quality of Multimedia Experience*, pp. 1–6, 2019.