# Defense-CycleGAN: A Defense Mechanism Against Adversarial Attacks Using CycleGAN to Reconstruct Clean Images

1st Zhixun He
*Electrical Engineering & Computer Science*
*University of California, Merced*
Merced, CA 95343

zhe5@ucmerced.edu

2nd Mukesh Singhal
*Electrical Engineering & Computer Science*
*University of California, Merced*
Merced, CA 95343

msinghal@ucmerced.edu

***Abstract*—Deep Neural Networks (DNNs) approaches have been used successfully in various computer vision tasks. However, they are particularly susceptible to adversarial attacks, which can cause incorrect predictions and raise security risk to real-world deep learning applications, e.g., autonomous driving and surveillance systems. Many state-of-the-art adversarial defense models use Variational Autoencoder Decoder (VAE) to reconstruct clean images to counter adversarial attacks. However, images reconstructed from VAEs are often blurry, thus, not only their use for classification is limited, but also such defense methods blur the benign inputs and impair the accuracy when there is no adversarial attack. We propose to use Cycle Consistency GAN (CycleGAN) as the image reconstruction block, which generates higher quality images for better classification performance. Our method achieves 5%-13% higher accuracy than the best VAE-based defense models on CIFAR10 and achieves 90%-98% accuracy for Fashion-MNIST across wide range of adversarial attacks.**

***Index Terms*—Adversarial Defense, Adversarial Learning, CycleGAN, VAE**

## I. Introduction

DNN-based computational learning approaches have delivered unprecedented success in various domains, such as videos [1], images [2], texts [3] and audios [4], etc. However, these deep learning based methods have been shown to be susceptible to adversarial attacks [5], [6]. Take image classifiers for example, a single [7] or multiple [6] pixels in an image can be carefully manipulated, such that the deep networks are misled to give wrong predictions. Such pixel perturbation is often small enough and human eyes can hardly tell the difference between the adversarial image and the original image [8].

Various defense mechanisms have been developed to make deep learning applications more robust against adversarial attacks [9]–[12]. They can be broadly classified into three major categories: (1) use of data augmentation to mix the training data with adversarial samples or to add noise in the training data [13]–[15] to make the classifier more robust; (2) use of high-level latent features from the neural network to detect adversarial samples, for example, the latent features for clean data forms its unique distribution, but an adversarial sample will output a latent feature as an outlier [16], [17]; (3) use of generative networks (e.g., GANs or VAEs) [9], [11], [18], [19] to clean adversarial noise or to recreate images that are similar to the clean data.

Using VAEs as the generative networks [9], [18], [20] to reduce adversarial noise has delivered promising results. However, there is a trade-off between the noise reduction and the quality of the reconstructed images from VAEs, and the reconstructed images from VAEs are almost always blurry. Such blurry effects also happen when the input data are not perturbed by adversarial attacks, and the classification accuracy on those generated blurry images drops even when there is no adversarial attack.

Generative Adversarial Network (GAN) [6] is a training strategy which includes a Discriminator network $D$ and a Generator network $G$. $G$ is trained to generate samples that cannot be distinguished from the original samples by $D$. The generated samples do not have to look the same as the original samples, but instead, $G$ tries to mimic the original samples' distribution and to generate images from this imitated distribution. GANs have been used in various areas [6], [21], [22] to generate high fidelity images. In the domain of adversarial defense, GAN-based methods [11], [23], [24] suffer when the adversarial noise varies in wide spectrum.

In this work, the authors propose a method to defend adversarial attacks, which trains a de-noise network in a way similar to train Cycle Consistency Generative Adversarial Network (CycleGAN) but with a focus on noise reduction. In order to generate high fidelity clean images, the generative network can take the advantage of latent features from both early layers and deep layers in the network to create detail rich and less blurry images. Our extensive experimental study shows that the proposed method outperforms the state-of-the-art adversarial defense methods across various adversarial attacks and wide spectrum of adversarial noise levels. Plus, when there is no adversarial noise in the input data, the proposed defense framework can still maintain nearly the same accuracy as that from the original classifier alone.

## II. Related work

**Variational Autoencoder Decoders (VAEs)** [25] have achieved impressive progress in image generation [26], [27], data compression [28] and interpolation between sentences [29]. Recent efforts in adversarial defense [9], [12], [18], [30] adopt the idea of compressing the input data into constrained multivariate latent distribution to reduce the adversarial noise, such as high frequency loss VAE [9] and Gaussian mixture VAE [20]. While VAEs are easy to train, the drawback is that the reconstructed images are blurry compared to its clean inputs [9], [26].

**Generative Adversarial Networks (GANs)** [31] have shown excellent capability of generating high fidelity images [21], [32], style transferring [33], [34] and image editing [22]. GANs used in adversarial defense could be trained either to simulate the adversarial attacks' noise pattern [24] or to imitate the clean training samples' distribution [11] The major differences between VAEs and GANs are: 1) VAEs usually consist of an Encoder that compresses the input data to intermediate vector which has much smaller dimension, and a Decoder that uses the compressed vector as the only source to reconstruct data, while GANs can use latent features from any depth of the network [35] or use random noise to help the data generation process [21]. 2) GANs' generative networks tend to imitate the data distribution, instead of generating data exactly same as the training data, while VAEs tend to generate data same as the input to the network. This is because GANs' training objective is to generate samples that confuse its discriminator network in stead of imitating any training data, but VAEs' main objectives is to lower the Euclidean distance between input and output data.

GANs have been used in different adversarial defense work, for example, [23] uses GAN to eliminate adversarial noise, [24] uses a generative network to create adversarial noise and trains a classifier to be robust to such noise, and [11] tries to generate images that belong to the same category as the adversarial samples. The proposed method is different from the above methods in that: 1) Compared to [23], our method trains the classifier using the generative network's output instead of feeding the generated images back to the original classifier. The outputs from a single source are distributed more consistently and a classifier can gain better performance after it's trained using such data, which gives the proposed method better classification accuracy when the adversarial perturbation levels are high compared to [23]. 2) Compared to [11] and [24], our method feeds forward the unknown inputs into the de-nosie network and classifies the reconstructed outputs directly, instead of making the classifier robust to noise or finding similar samples that belong to the same category as the unknown input. 3) The proposed method's data generating process takes the advantage of the latent features from both early layers and deep layers of the network to reconstruct high fidelity images.

**Cycle Consistency** Regularizing deep learning by enforcing cycle consistency has been extensively studied for more than a decade [36]. By encouraging the consistency on the forward-backward loop, the networks learn a mapping between two domains. For example, [37] learns a dense correspondence between 3D projection and 2D pixel; [38] learns the dense correspondence of instances between the real images and a synthetic data set. This learning approach alleviates the need for extensive manual labeling of data, like corresponding locations on 3D model [37]; [33], [39] learns the style translation between images; By enforcing the global cycle consistency in addition to supervised learning, it helps learn more robust models, e.g., [40] imposes a cycle consistency loss to form a more reliable visual questioning application. Our proposed method enforces cycle consistency loss to de-noise potential adversarial noise through an image-to-image translation, generating cleaner images which are nearly as identical as the original data.

**Data Augmentation** has been demonstrated to be an effective regularization technique to alleviate over-fitting issues and expand the training data set [41]. It includes color space transformation [42], geometric transformation [41] and noise injection, etc. Efforts in provable adversarial defense [13], [14], [43] try to provide theoretical ground for why data augmentation makes DNNs more resilient to adversarial noise.

## III. The proposed method

The sequence of flow of actions in the proposed method is shown in Figure 1. After training, the input images $X = \left\{x^{(i)}\right\}_{i=1}^{N}$ are first fed into the CycleGAN to reconstruct their de-noised counterparts: $f_{CycleGAN} : X \rightarrow X^{\sim}$. The reconstructed images $X^{\sim}$ are classified by the original classifier: $f^* : x \rightarrow \underset{j}{argmax}\ c_j$, where $c_j \in [0,1]^k$, $k$ is the total number of categories, $j = \{0, 1, ..., k\}$ is the index of a category, and $c_j$ is the sotfmax layer output from the classifier for each class $j$. And $X^{\sim}$ are also fed into a group of $M$ post-CycleGAN classifiers $F = \left\{f^{(i)}\right\}_{i=1}^{M}$ for their predictions, $O = \left\{f^{(i)}(x^{\sim})\right\}_{i=1}^{M}$. For the final decision, the output from the original classifier, $f^*(x)$, and the outputs from $M$ post-CycleGAN classifiers, $O$, will be jointly considered through the Bayeian update process that is discussed later in section III-C. The loss used for training all classifiers is categorical cross entropy loss $L_{cross\_entropy}$.

### A. CycleGAN

**Adversarial Loss** [31] is used as the objective to guide the two identical generative networks, $G_A$ and $G_B$, to generate samples that are indistinguishable from the target domain. If we denote the training samples as $X$, the generated samples as $X_g = G_A(X)$, and the two discriminator networks as $D_A$ and $D_B$ (for $G_A$ and $G_B$, respectively), the training loss for $D_A$ (similar to $D_B$) is the Least Squares [44] between $X$ and $X_g$, so the loss for $D_A$, $L_{D_A}$ (similar for $D_B$) is:

$$\begin{aligned} L_{D_A} = &\mathbb{E}_{x\sim P(X)}\left[\sqrt{\left(1 - D_A(x)\right)^2}\right] + \\ &\mathbb{E}_{x\sim P(X)}\left[\sqrt{\left(D_A(G_A(x))\right)^2}\right] \end{aligned} \tag{1}$$

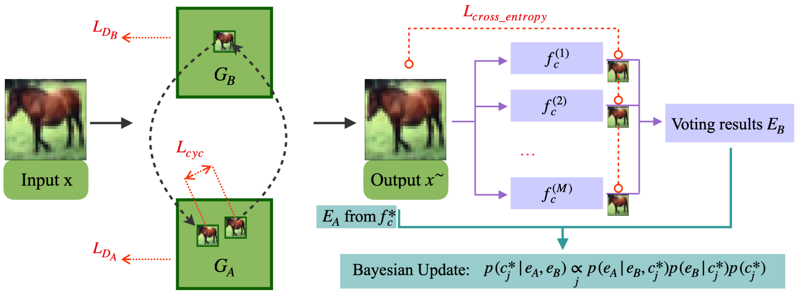

Fig. 1. The proposed defense mechanism.

where $D : x \to [0, 1]$, and $G : x \to x_g$

The generative network $G$'s objective is to generate samples $X_g$ that are similar to the domain targets $X$. The loss for training the generative network $G_A$, $L_{G_A}$, (similar to $G_B$) is as below:

$$L_{G_A} = \mathbb{E}_{x \sim P(X)} \left[ \sqrt{(1 - D_A(G_A(x)))^2} \right] \tag{2}$$

**Cycle Consistency Loss**. A generative network with large enough capacity can map an input image to various images that are similar to those images from the target domain, but an image is likely transformed into a different category. This causes an issue when the goal is classification, since the adversarial training does not guarantee that the generated image belongs to the same category as that of the input image.

Thus, in order to reliably train $G$ to generate images that fall within the same category as that of the input images, we propose to enforce a cycle consistency loss between the input images, $X$, and the reconstructed images, $G_A(G_B(X))$, as well as that in between $X$ and $G_B(G_A(X))$:

$$\begin{aligned} L_{cyc} = &\mathbb{E}_{x \sim P(X)} \left[ |x - G_A(G_B(x))| \right] + \\ &\mathbb{E}_{x \sim P(X)} \left[ |x - G_B(G_A(x))| \right] \end{aligned} \tag{3}$$

The use of cycle consistency loss helps train the network to make predictable transformation from $X$ to $G_B(G_A(X))$, and from $X$ to $G_A(G_B(X))$. With the objective, $L_{cyc}$, the easiest way for $G_B$ to generate $G_B(G_A(X))$ that are identical to the input $X$ is that $G_A$ focuses only on the output's distribution but maintains the content of the images relatively stable such that $G_B$ can also focus only on the output's distribution and recreate images identical to $X$.

### B. U-Net

U-Net [35] is the generative network framework used in image generation process. VAEs tend to compress input data to much smaller dimension, and this compressed latent feature is the sole information used to reconstruct the data with original dimension. This process often creates noticeable blurry effect on the generated images [9], [20], [30], thus, it limits the classification accuracy on those images. Compared to VAEs, U-Net, shown in the Figure 3, incorporates the hidden features from each depth level in the network into the image generating process. It concatenates early hidden features to deep hidden features in an iterative fashion. Thus, U-Net can use those primitive features from the early layers to generate detailed rich images, almost identical to the target domain images, which is further discussed in section IV-C.

### C. Collective voting and Bayesian update

Combining the results from multiple classifiers not only alleviates adversarial attacks but also increases the accuracy of final results [9]. After a group of psot-CycleGAN classifiers provide their predictions, $O$, a majority vote is taken. This voted prediction will be used as one of the evidences in Bayesian update and is denoted as $E_B$. The prediction from the original classifier on the reconstructed image, $f^*(x^{\sim})$, will be used as another evidence for Bayesian update and is denoted as $E_A$. The original classifier's prediction on the unknown input image, $f^*(x)$, is the prior in the Bayesian process that will be updated using $P(E_A = e_A)$ and $P(E_B = e_B)$, where $e_A$ and $e_B$ are the category index. Using the two evidences above, the posterior (the final prediction result) for an unknown input is calculated using Bayesian update [9] as follows:

$$\begin{aligned} &p(c_j|e_A, e_B) \\ &= \frac{p(e_A|e_B, c_j)p(e_B|c_j)p(c_j)}{\sum_j p(e_A|e_B, c_j)p(e_B|c_j)p(c_j)} \\ &\underset{j}{\propto} p(e_A|e_B, c_j)p(e_B|c_j)p(c_j) \end{aligned} \tag{4}$$

## IV. Experiments

### A. Experimental Setup

To evaluate the proposed defense mechanism, extensive experiments were conducted on CIFAR10 [45] and Fashion-MNIST [46] data sets. Fashion-MNIST contains 10 classes of hand-written digits, with 60,000 training samples and 10,000 testing samples. Each sample is a 28 × 28 gray-scaled image. CIFAR10 contains 10 categories of animals and objects, with 50,000 training samples and 10,000 testing samples, and each sample is an RGB image with size of $32 \times 32 \times 3$. The networks were trained using the training samples and all experiments were evaluated using the full 10,000 testing samples from each data set.

We compared the proposed method with three other existing defense methods: High Frequency Loss VAE (VAE) [9], Defense-VAE [12] and Defense-GAN [11]. All these defense mechanisms belong to the similar category, that the defense mechanism includes a generative network to de-noise the

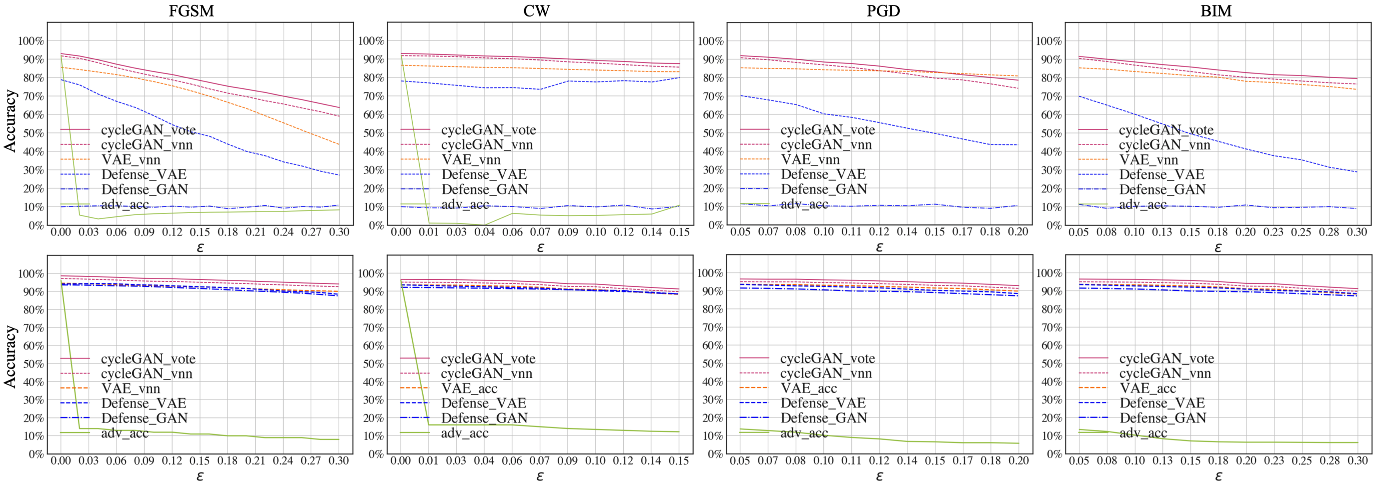

Fig. 2. Comparison of four defense methods under FGSM, CW, PGD and BIM on CIFAR10 (top graphs) and Fashion-MNIST (bottom graphs). The x-axis is the adversarial noise level and the y-axis is the defense mechanism's accuracy on testing data. Caption: blue dash-dot line: Defense-GAN, blue dash: Defense-VAE, orange dash line: VAE, red solid line: the proposed method, red dash line: the average voting classifier's accuracy without Bayesian update, green line: the accuracy without defense.

adversarial noise or to reconstruct images for classification. We tested all defense methods under four adversarial attacks: Projected Gradient Descent (PGD) [47], Carlini and Wagner Method (CW) [48], Fast Gradient Sign Method (FGSM) [6], Basic Iterative Method (BIM) [49]. For the implementation of generating adversarial samples of the four different attacks, we used the Cleverhans package [50]. Except that the perturbation level $\epsilon$ of the four adversarial attacks was varied in order to generate adversarial samples with various noise, the rest of the parameters of those algorithms were set to their default values.

For the post-CycleGAN classifiers, we adopted three different types of structures (Residual-Networks [51], Wide-Residual-Networks [52] and DenseNet [53]). The implementation details of those classifiers were inspired by the public Github repository [54].

During Bayesian update, the Conditional Probability Tables (CPT) in $P(e_B|c_j)$ and $P(e_A|e_B, c_j)$ were calculated using the full training set. Because when the original classifier is under adversarial attack, the original classifier's prediction on a potential adversarial sample, $P(C = c_j)$, becomes unreliable to use, during experiment, we assumed $P(C = c_j)$ to be the same for all its categories. The final prediction for each test image was given by the final prediction from equation (4).

### *B. White-box Attacks on Classifiers*

Under white-box attacks, where the adversary has access to the classifier's configuration to generate adversarial samples, the performance of different methods against FGSM, CW, PGD and BIM over a wide range of adversarial noise level is shown in Figure 2.

Fashion-MNIST data set includes mono colored images of simple items, like shoes, T-shirts and pants so no, and there is no much variation in sizes, directions or background changes, while the CIFAR10 data set consists of RGB colored images, like cats, birds and flights etc. CIFAR10 is much more complex than Fashion-MNIST, because not only the outline of objects in CIFAR10 images varies a lot, but also the colors and patterns of the objects in each category have much more variations than Fasion-MNIST images. Thus, we see the performance of all four defense methods is better in Fasion-MNIST, where they all achieve above 90% accuracy throughout a wide range of adversarial attacks.

On CIFAR10 data set, all four defense mechanisms struggle to maintain high accuracy when the adversarial noise level is high. The CycleGAN method outperformed Defense-VAE and Defense-GAN by a margin nearly 30% on average, and it outperformed VAE method by 5%-13% across a wide range of adversarial perturbation levels. And the Baeysian update using the collective voting result from multiple classifiers $f^{(i)}$ gives extra 3-5% accuracy on top of the accuracy given by a single classifier.

A good defense mechanism should maintain its performance on clean data when there is no adversarial attack. Figure 2 shows that the original classifier can reach around 97% accuracy on clean testing images for Fashion-MNIST and 91% for CIFAR10 when there is no attack. For Fashion-MNIST, Defense-VAE, Defense-GAN and VAE reach around 92% accuracy, except the proposed method which reaches an identical 97% accuracy as that of the original classifier. For CIFAR10, the Defense-GAN and Defense-VAE both fall short under 80% and the VAE reaches around 86% on clean testing data, except the proposed method which can still maintain the same level of accuracy, 91%. The proposed method does not sacrifice its performance on clean data in exchange for better defense features.

### *C. Image Reconstruction*

In order to better understand the advantage that CycleGAN provides over the VAE in image reconstruction when there is no adversarial attack, we compared the average L1 norm for the difference between the original images and the reconstructed images that are generated from these generative networks.

Note that the main network structures difference between VAE and CycleGAN is that VAE is fully sequential in terms of its network layers connections, but the generative network (U-Net) in CycleGAN concatenates early layer's latent feature outputs to deep layer's outputs, as shown in Figure 3. It's well known in computer vision community that the early layers in DNNs extract some primitive features in images, like edges, corners and colors [55].

To investigate the influence of the early latent features on the reconstructed images' quality, we gradually removed the connections between early layers' features and deep layers' features in CycleGAN. We first broke the *concatenate 1* in Figure 3 while keeping *concatenate 2, 3 and 4* untouched. After *concatenate 1* was disconnected, to keep the network output dimension to be the same as before, the sequential layers after *concatenate 2* were adjusted by doubling the number of its filters to imitate the same output dimension as that from *concatenate 1*. We denoted this generative network as Cyc_1. Similarly, the experiments decoupled the *concatenate 1&2*, the *concatenate 1&2&3*, and the *concatenate 1&2&3&4*, and denoted each of the new network as Cyc_2, Cyc_3 and Cyc_4, respectively. The L1 norm for the difference between the original images and the reconstructed images from those networks when there is no attack, averaged over 10,000 testing samples, is shown in Table I.

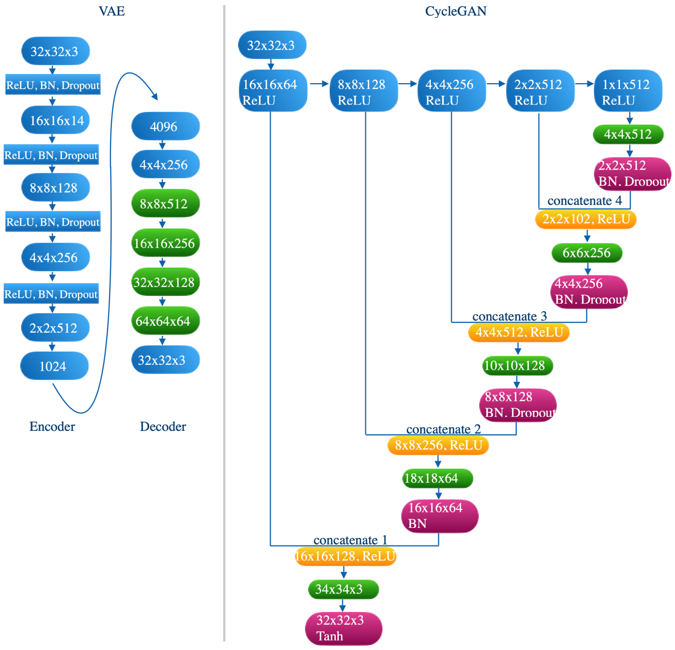

Fig. 3. The generative networks' structures for VAE (left) and CycleGAN (right). For both networks, all the blue boxes with dimension output are Conv2D layers, and they all use kernel size 4, strides 2 and ReLU activation as their layer parameters; all the green boxes are Con2DTranspose layers, and kernel size 4 and strides 2 are used as their layer parameters. For the CycleGAN, the red boxes represent Crop operation to trim the output dimension before they are fed into the yellow boxes which denote a Concatenation operation and their output dimension.

| Network | VAE | Cyc | Cyc_1 | Cyc_2 | Cyc_3 | Cyc_4 |
|---|---|---|---|---|---|---|
| L1 Norm | 126.0 | 54.3 | 103.9 | 225.9 | 358.6 | 518.8 |

TABLE I

L1 Norm of the difference between original images and reconstructed images from various generative networks.

The smaller the L1 norm, the closer the reconstructed images to the original images. From Table I, we observe that the L1 for the images generated from CycleGAN is 54.3, noticeably less than 126.0 for the VAE-generated images, which helps explain why the classifier can reach higher accuracy on those images generated from CycleGAN compared to those from VAE when there is no attack. From Cyc_1 to Cyc_4, the L1 increases from 103.9 to 518.8, and after Cyc_2, the L1 has become worse than that from the VAE. Note that Cyc_4 has already become a fully sequential network after breaking all the concatenation layers, which has a similar structure as VAE. Cyc_4 and VAE share a similar feature, that there is a bottleneck layer in the network, where its output size is much smaller than that of network's input and output. Although both VAE and Cyc_4 impose a reconstruction loss during training, VAE regularizes the bottleneck layer's output to imitate multivariate Gaussian distribution while Cyc_4 does not. Because of the lack of regularization in Cyc_4's bottleneck layer, the accuracy on image reconstruction from Cyc_4 suffers. From Cyc_1 to Cyc_4, the main difference is the availability of early layer's latent features for image reconstruction, and the reason for an increasing L1 from Cyc_1 to Cyc_4 is that there are less primitive features from the early layers to provide accurate details for generating high fidelity images to the input images.

## V. Conclusion

An innovative adversarial defense mechanism, Defense-CycleGAN, was proposed to de-noise adversarial samples and generate clean images for classification. CycleGAN concatenates network's early layer's primitive features to those from deep layers, allowing better fidelity image reconstruction. We conducted extensive experiments to compare Defense-CycleGAN with three well-known adversarial defense methods under four different adversarial attacks with a wide range of adversarial perturbation levels on CIFAR10 and Fashion-MNIST. The experiments showed that the images reconstructed from CycleGAN are more accurate than those from the VAE-based approach. It also showed that the proposed method achieves 90%-98% accuracy for Fashion-MNIST across different attacks and reaches above 80% accuracy under CW, PGD and BIM attacks on CIFAR10. The proposed method outperforms three other methods by a 5%-30% margin across different attacks. When there is no attack, the other three methods showed a 6%-10% drop in classification accuracy compared to that of the original classifier, but the proposed method showed an identical accuracy to the original classifier's.

# References

[1] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal processing magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[3] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," in *ICML*, 2011.

[4] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Suk thankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2014, pp. 1725–1732.

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

[6] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[7] J. Su, D. V. Vargas, and K. Sakurai, "One pixel attack for fooling deep neural networks," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 5, pp. 828–841, 2019.

[8] K. Ren, T. Zheng, Z. Qin, and X. Liu, "Adversarial attacks and defenses in deep learning," *Engineering*, vol. 6, no. 3, pp. 346–360, 2020.

[9] Z. He and M. Singhal, "Adversarial defense through high frequency loss variational autoencoder decoder and bayesian update with collective voting," in *2021 17th International Conference on Machine Vision and Applications (MVA)*. IEEE, 2021, pp. 1–7.

[10] N. Akhtar and A. Mian, "Threat of adversarial attacks on deep learning in computer vision: A survey," *IEEE Access*, vol. 6, pp. 14 410–14 430, 2018.

[11] P. Samangouei, M. Kabkab, and R. Chellappa, "Defense-gan: Protecting classifiers against adversarial attacks using generative models," *arXiv preprint arXiv:1805.06605*, 2018.

[12] X. Li and S. Ji, "Defense-vae: A fast and accurate defense against adversarial attacks," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2019, pp. 191–207.

[13] E. Wong and Z. Kolter, "Provable defenses against adversarial examples via the convex outer adversarial polytope," in *International Conference on Machine Learning*. PMLR, 2018, pp. 5286–5295.

[14] J. Cohen, E. Rosenfeld, and Z. Kolter, "Certified adversarial robustness via randomized smoothing," in *International Conference on Machine Learning*. PMLR, 2019, pp. 1310–1320.

[15] A. Raghunathan, J. Steinhardt, and P. Liang, "Certified defenses against adversarial examples," *arXiv preprint arXiv:1801.09344*, 2018.

[16] G. Goswami, A. Agarwal, N. Ratha, R. Singh, and M. Vatsa, "Detecting and mitigating adversarial perturbations for robust face recognition," *International Journal of Computer Vision*, vol. 127, no. 6-7, pp. 719–742, 2019.

[17] J. Lu, T. Issaranon, and D. Forsyth, "Safetynet: Detecting and rejecting adversarial examples robustly," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 446–454.

[18] D. Meng and H. Chen, "Magnet: a two-pronged defense against adversarial examples," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017, pp. 135–147.

[19] S. Gu and L. Rigazio, "Towards deep neural network architectures robust to adversarial examples," *arXiv preprint arXiv:1412.5068*, 2014.

[20] P. Ghosh, A. Losalka, and M. J. Black, "Resisting adversarial attacks using gaussian mixture variational autoencoders," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 541–548.

[21] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4401–4410.

[22] J.-Y. Zhu, P. Krähenbühl, E. Shechtman, and A. A. Efros, "Generative visual manipulation on the natural image manifold," in *European conference on computer vision*. Springer, 2016, pp. 597–613.

[23] G. Jin, S. Shen, D. Zhang, F. Dai, and Y. Zhang, "Ape-gan: Adversarial perturbation elimination with gan," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 3842–3846.

[24] H. Lee, S. Han, and J. Lee, "Generative adversarial trainer: Defense to adversarial perturbations with gan," *arXiv preprint arXiv:1705.03387*, 2017.

[25] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[26] L. Cai, H. Gao, and S. Ji, "Multi-stage variational auto-encoders for coarse-to-fine image generation," in *Proceedings of the 2019 SIAM International Conference on Data Mining*. SIAM, 2019, pp. 630–638.

[27] A. Razavi, A. van den Oord, and O. Vinyals, "Generating diverse high-fidelity images with vq-vae-2," in *Advances in neural information processing systems*, 2019, pp. 14 866–14 876.

[28] W. Williams, S. Ringer, T. Ash, J. Hughes, D. MacLeod, and J. Dougherty, "Hierarchical quantized autoencoders," *arXiv preprint arXiv:2002.08111*, 2020.

[29] S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R. Jozefowicz, and S. Bengio, "Generating sentences from a continuous space," *arXiv preprint arXiv:1511.06349*, 2015.

[30] J. Zhu, G. Peng, and D. Wang, "Dual-domain-based adversarial defense with conditional vae and bayesian network," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 1, pp. 596–605, 2020.

[31] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[32] A. Brock, J. Donahue, and K. Simonyan, "Large scale gan training for high fidelity natural image synthesis," *arXiv preprint arXiv:1809.11096*, 2018.

[33] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

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

[35] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[36] Z. Kalal, K. Mikolajczyk, and J. Matas, "Forward-backward error: Automatic detection of tracking failures," in *2010 20th international conference on pattern recognition*. IEEE, 2010, pp. 2756–2759.

[37] N. Kulkarni, A. Gupta, and S. Tulsiani, "Canonical surface mapping via geometric cycle consistency," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 2202–2211.

[38] T. Zhou, P. Krahenbuhl, M. Aubry, Q. Huang, and A. A. Efros, "Learning dense correspondence via 3d-guided cycle consistency," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 117–126.

[39] Y.-C. Chen, P.-H. Huang, L.-Y. Yu, J.-B. Huang, M.-H. Yang, and Y.-Y. Lin, "Deep semantic matching with foreground detection and cycle-consistency," in *Asian Conference on Computer Vision*. Springer, 2018, pp. 347–362.

[40] M. Shah, X. Chen, M. Rohrbach, and D. Parikh, "Cycle-consistency for robust visual question answering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6649–6658.

[41] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, pp. 1–48, 2019.

[42] A. Mikołajczyk and M. Grochowski, "Data augmentation for improving deep learning in image classification problem," in *2018 international interdisciplinary PhD workshop (IIPhDW)*. IEEE, 2018, pp. 117–122.

[43] M. Balunovic and M. Vechev, "Adversarial training and provable defenses: Bridging the gap," in *International Conference on Learning Representations*, 2019.

[44] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley, "Least squares generative adversarial networks," *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. [Online]. Available: http://dx.doi.org/10.1109/ICCV.2017.304

[45] A. Krizhevsky, V. Nair, and G. Hinton, "Cifar-10 (canadian institute for advanced research)," 2009. [Online]. Available: http://www.cs.toronto.edu/ kriz/cifar.html

[46] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," 2017.

[47] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," *arXiv preprint arXiv:1706.06083*, 2017.

[48] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 ieee symposium on security and privacy (sp)*. IEEE, 2017, pp. 39–57.

[49] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," *arXiv preprint arXiv:1607.02533*, 2016.

[50] N. Papernot, F. Faghri, N. Carlini, I. Goodfellow, R. Feinman, A. Kurakin, C. Xie, Y. Sharma, T. Brown, A. Roy *et al.*, "Technical report on the cleverhans v2. 1.0 adversarial examples library," *arXiv preprint arXiv:1610.00768*, 2016.

[51] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[52] S. Zagoruyko and N. Komodakis, "Wide residual networks," *CoRR*, vol. abs/1605.07146, 2016. [Online]. Available: http://arxiv.org/abs/1605.07146

[53] G. Huang, Z. Liu, and K. Q. Weinberger, "Densely connected convolutional networks," *CoRR*, vol. abs/1608.06993, 2016. [Online]. Available: http://arxiv.org/abs/1608.06993

[54] W. Li, "cifar-10-cnn: Play deep learning with cifar datasets," https://github.com/BIGBALLON/cifar-10-cnn, 2017.

[55] A. Ng, J. Ngiam, C. Y. Foo, and Y. Mai, "Deep learning," *CS229 Lecture Notes*, pp. 1–30, 2014.