Available online at: http://www.ijmtst.com/vol6issue04.html

International Journal for Modern Trends in Science and Technology

ISSN: 2455-3778 :: Volume: 06, Issue No: 04, April 2020

A Study on Generative Adversarial Perturbations Attacks

Swati C Thavrani[1] | Mosin I Hasan[1] | Kirtikumar J Sharma[1]

[1]Computer Engineering, Birla Vishvakarma Mahavidyalaya, Anand, Gujarat, India

To Cite this Article

Swati C Thavrani, Mosin I Hasan and Kirtikumar J Sharma, "A Study on Generative Adversarial Perturbations Attacks", International Journal for Modern Trends in Science and Technology, Vol. 06, Issue 04, April 2020, pp.:182-188.

Article Info

Received on 15-March-2020, Revised on 09-April-2020, Accepted on 11-April-2020, Published on 15-April-2020.

ABSTRACT

In last few years Generative model known as Generative adversarial networks (GANs). GANs are architecture to train generative models. GANs uses two models: Generative model and Discriminator model, where Generative model create new images by adding some random noise in existing image and Discriminator model check whether the image is real or fake. In Deep Convolution Neural Network, Generative Adversarial Network is one of the most dynamic analysis, possibilities, and its outstanding image generation capability has received wide attention. In GANs there are two approaches: Generator model and Discriminator model. An Adversarial Networks are classified as Targeted attack and Untargeted attack. This research summarized existing work of the Adversarial Networks from the Generative model and Discriminator model work. Nowadays, Adversarial Networks are commonly used in the industry.

KEYWORDS: *Generative adversarial network, targeted attacks, Untargeted attacks, Discriminator model, Genretor model, FGSM, PGD, CNN*

I. INTRODUCTION

Generative adversarial networks are the approach to generative data using deep convolution neural networks. A generative model is associated with unsupervised learning tasks in machine learning that involves mechanically discovering and learning the patterns in an input file in such a way that how the model will be able to generate or output new examples that may draw from the initial dataset.[1]

GANs can be divided into two parts which are the Generator model and the Discriminator model. The generator model is used to generating images and the discriminator model is used for classifying whether the image is real or generated. By using a Generative model, a Discriminative model is fooled. The generative model can create its image by adding some random noise. The discriminative model will check that if the image is real or fake.[1]

Adversarial examples are classified as Targeted and Untargeted attacks. A Targeted adversarial example is one that has been misclassified as a particular target class by an attacker. On the other hand, an Untargeted adversarial example is one that has been misclassified as any random class other than the original class. There are several advantages and disadvantages associated with the two methods. The targeted adversarial example displays the disadvantage of requiring more processing time and including greater misrepresent however, its advantage is that an attacker can perform classy attacks that can be

misclassified as the Target class by the attacker. The Untargeted adversarial example shows the disadvantage of not applying to classy attacks however, it exhibits the advantage of requiring less processing time and including lesser misrepresent.[2]

The most wide used to GAN arise from improve the efficiency of the machine by feeding it a lot of virtual feeding machine more amount of data by adding random noise on existing dataset. Also using GAN is getting popular image blending, image inpainting, image translation, semantic segmentation, Face aging, Video prediction, Cloth translation, and self -driving car.[3]

The figure-1 is showing that adversarial example. You can start with an image of a panda on the left some network to predict with 57.7% confidence is PANDA. The panda category is the also category is the highest confidence out of all categories, so the network concludes the object in the image is pandas. But by adding some amount of noise you get an image that looks exactly to a human but network think with 99.3% confidence is a **GIBBON.**[4]

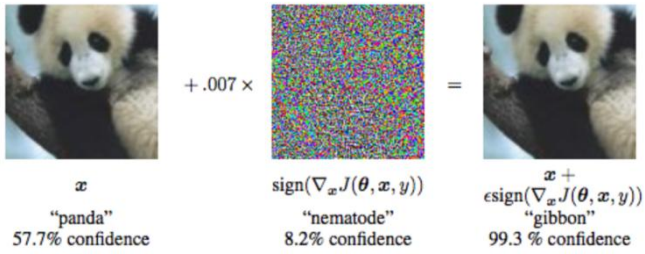

Figure 1 Example of Adversarial network[4]

The figure-2 also illustrates the cityscapes dataset. The first row in the below image is the original image and segmentation predict by the network. The second row in the below image adds adversarial perturbation and the network predicts that image is a perturbed image. The prediction corresponds to our adversarial target segmentation and does not correspond to the input image.[5]

Figure 2 Example of Adversarial network of object detection[5]

II. Generative Adversarial Network System

The figure-3 is illustrate that the GAN model architecture involves two sub-models:[1] a generatormodel for generating new examples and a discriminatormodel for classifying whether generated examples are real or fake, generated by the generator model. The below figure a generative model can try to make misclassification from the discriminative model. From the figure, non-targeted adversarial attacks and targeted adversarial attacks it's type of attacks. In a non-targeted adversarial attack, it is the most general type of attack when all you want to do make the classifier give an incorrect result. The targeted adversarial attack is more difficult which aims to receive a particular class for input.

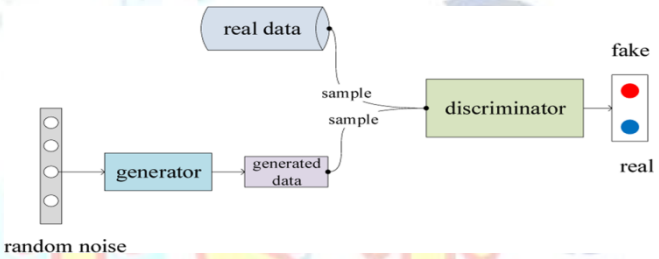

Figure 3 System Flow of GANs[6]

III. Survey of Existing Methods

Generative model: From the name it understands it is a generative algorithm. A generative model is inverse of convolution neural network, because in CNN from the dataset original image is given as input and classify label as an output, but in a generator, noise is given as input to the inverse CNN and an original image as expected image as output. In other words, a generator model generates data from the dataset using its images.[1]

Example: The figure-4 has contained the process of the generative model. Suppose in the image below a Dataset containing an image of a horse. This may need to create a model that will create a new image of a horse that has never existed but still, that image looks real because the model has learned to overall rules that govern the appearance of the horse. This kind of problem will be solved by the generative model.[7]

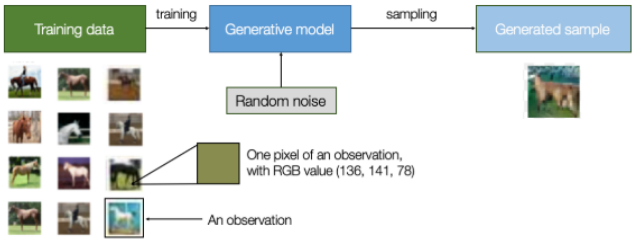

Figure 4 The Process of Generative model[7]

Discriminator Model: Discriminator is a convolution neural network consist of many hidden layers and one output layer, a major difference in the output layer of GANs has only two outputs. The output of discriminator model is either 0 or 1 because of a specially selected activation function for this task. If 0 then the provider data is fake, if the output is 1 then the provider data is real. A discriminator is trained on the real data so it learns to recognize how actual data looks like and what features should the data have to be classified as real.[1]

Working of both generative model and discriminator model together: Generator model starts to generate data from the random input so that generated information pass to discriminator model as input. Now discriminator model can analyses the generated data and check the data was real or fake. If the generative data doesn't contain enough option to be classified as real data by the discriminator model, than this generative data or weights sent back to the generative model by using backpropagation, so it will readjust the related to new data that is best from the previous one. This new generated data is again passed to the discriminator model and it can be continue until the generated data can be real contain by discriminator model.[1]

FGSM(Fast gradient sign method): The fast gradient sign methodology works by neglect the gradient of the neural network from an adversarial example. For the input image, this method has used the loss with relevance to the input image to form a replacement image that maximizes the loss. The new image is named the adversarial image.this will be summarized using the following equation [8]

Out_img = x+Ɛ*sign (∇_x J(P,x,y)) [8]

Out_img: Adversarial image.
x: Original input image
y: Original input label
Ɛ: Multiplier to ensure the perturbations are small
P: Model parameter
J: Loss[8]

PGD (Projected gradient descent): Projected gradient decent method typically used once coping with a constraint optimization problems, wherever the constraint is compulsory on your possible set of the parameter. As optimize operator, would possibly take a step that takes outside of the possible set and seek out the way to correct for that. The way to attain it's to attain project the purpose back to the boundary of the possible set. This method is simply gradient descent, wherever everybody taking a step you project the ensuring purpose to the nearest possible set.[9]

CNN(Convolution neural network):CNN is widely used in areas like image classification, object detection, and recognition. In image classification, CNN processes an input image and categorizes it into classes. An input image passes through some convolutional layers. There are three types of layersinaCNN:Convolutional Layer, Pooling Layer, and Fully Connected Layer. Convolutional Layer is responsible for extracting features from the input image. It maintains the relation between pixels of an input image and performs convolution using an input image and a kernel. Based on the type of kernel, Feature Map is generated as an output. The activation function that is used by CNN is ReLU, which stands for the ReLU for a non- linear operation. It finds maximum from input. The pooling layer reduces the number of parameters from the image. Downsampling or subsamplingin CNN refers to which reduces the dimensions of a feature map but pertains to the important featuresof an image. Max Pooling and Min Pooling are used to find maximum and minimum among thekernel size respectively. The Fully Connected layer which uses softmax or sigmoid activationfunction to classify the outputs.[10]

Image Inpainting: Image inpainting is that the method of reconstructing missing components of a picture so that observers are unable to inform that these regions have undergone restoration. This method is usually accustomed takes away unwanted objects from an image broken parts of the previous image.[11]

IV. Existing Research On Adversarial Network

Chaowei Xiao, et.al[12], They propose AdvGAN to generate adversarial examples with GANs, that can learn and approximate the distribution of original examples. that can generate perturbations efficiently for any example, to possibly fast-track

adversarial training as defenses. They can be applied to Adversarial GAN in both semi-white box and black-box attack settings. The semi-white box attacks mean there is no need to access the original target model after the generator is trained, in contrast to traditional white-box attacks. The black-box attacks mean they dynamically train a refined model for the black-box model and optimize the generator accordingly. Adversarial examples generated by Adversarial GAN on different target models have a high attack success rate under state of-the-art defenses compared to other attacks. That attack has placed the first with 92.76% accuracy on a public MNIST black-box attack challenge. They apply AdvGAN to generate adversarial examples on different target models and test the attack success rate for them under the state-of-the-art defenses and show that our method can achieve higher attack success rates compared to other existing attack strategies. They generate all adversarial examples for different attack methods under an L1bound of 0.3 on MNIST and on CIFAR-10,for a fair comparison.

AdvGAN Framework:[12]

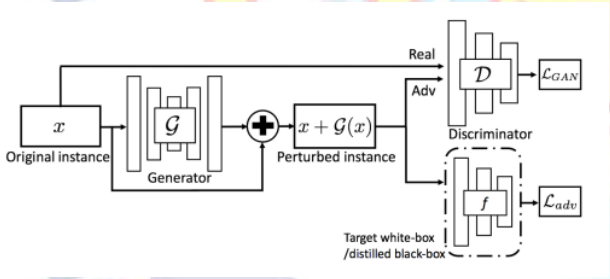

Figure 5 Overview of AdvGAN[12]

AdvGan mainly consist three parts it has a generator G, a discriminator D and the targeted neural network f. Here the generator G take input as original instance x and generates perturbation G(x). Then x + G(x) will be sent to the discriminator D, which is used to differentiate the generated dataand the original instance x. The goal of D is to inspire that the generated instance is in differentwiththedatafromitsoriginal class. To fulfill the goal of fooling a learning model, they first perform the white-boxattack, wherethetargetmodelis f in this case. f takes x + G(x) as its input and outputsits loss L_{adv}, which represents the distance between the prediction and thetargeted attack, or the opposite of the distance between the prediction and the untargeted attack.

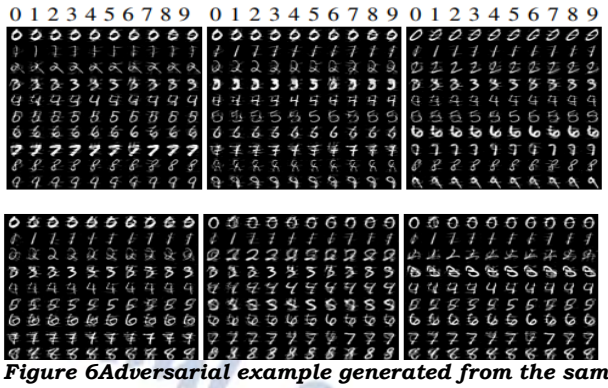

Figure 6Adversarial example generated from the same original image to different targets by AdvGAN on MNIST[12]

Yang song, et.al.[13], Discuss in this paper they propose unrestricted adversarial examples, which means any image like face, digits, etc. FGSM & PGD method is used for unrestricted or untargeted adversarial attacks. In this paper, the author observes the f(x) is approximately linear and proposes the fast gradient sign method, which applies first-order approximation of f(x) to speed up the generation of the adversarial image. This process can be repeated several times to gives to the stronger attack named projected gradient descent method. They mainly focused on their investigation of unrestricted adversarial networks. The implemented method wast tested on MNIST, CelebA, and SVHN.

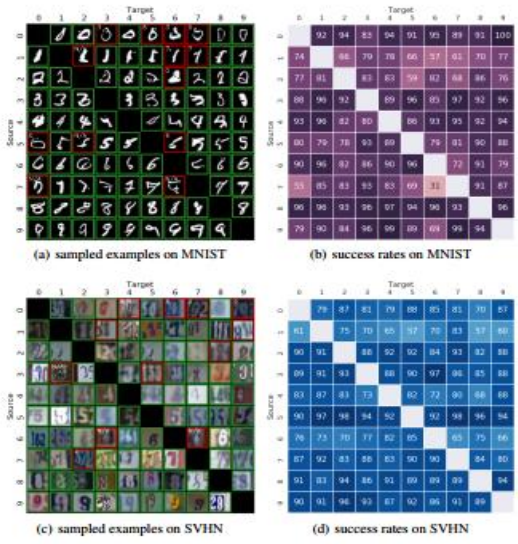

Figure 7 (a)(c) Random sample of targeted unrestricted adversarial example (b)(d)the success rate of targeted unrestricted attack[13]

Zheng-An Zhu,et.al.[14], Discuss in this paper A Generative Adversarial Networks based attack method is proposed by generating makeup images from non-makeup images. FGSM algorithm adds a

small amount of noise to the gradient of the input image to affect the classification result of the neural network model. PGD is to improve the original FGSM result. In this paper, ImageNet is used to a pre-train dataset and the makeup face dataset is used to fine-tuning these models. The makeup effect can be applied only to eyes. For attack they used two networks it has makeup transfer sub-network and adversarial attack sub-network. Makeup transfer sub-network proposed to based on CycleGan to transfer makeup face to non-makeup face. Where adversarial attack sub-network generates adversarial example that can attack target network.

Makeup transfer sub-network: A Makeup Transfer Sub-network is proposed to transfer face images in non-makeup domain to makeup domain. Following the setting of CycleGANs,two generators G and F are exploited to produce makeup and non-makeup images, respectively. G adds makeup effect to non-makeupimage is remove makeup effect to maintaining original identity.The system framework is showed in figure 8[14]

L_{GAN} (G,F,Dx,Dy)= -(Ex~px~py[log Dy(y)+ log Dx(x)+ log(1 - Dy(G(x))) + log(1 - Dx(F(y)))])[14]

Where x is the real non-makeup input,y is real makup input and network G generate result of G(x) that obfuscate discriminator Dy.F= YX

It is generates results of non- makeup F(y) which is also complicates for Dx.[14]

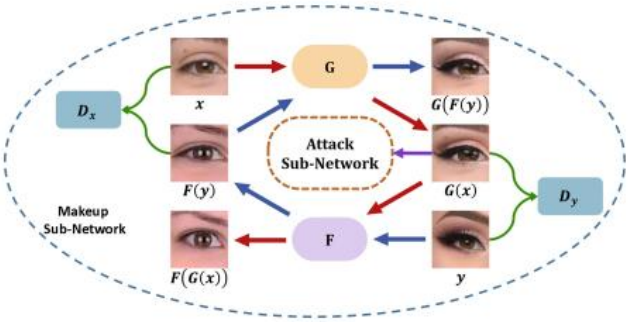

Figure 8 System framework of Makeup attacks including two sub-network[14]

Adversarial attack sub-network: Adversarial Attack Sub-networks is designed as GANs for generating adversarial examples. The eye regions with makeup G(x) are first mixed with the original non-makeup images x by the transformation function T. Then, it attends as the input to the generator H. H aims to generate output image H(T(x; G(x))) with perturbation noises that can deceive both the target network A and the discriminator Dh.Target network A is well-trained face recognition model to be attacked by the adversarial examples. Weights of model A are fixed all the time. It aims to generate adversarial examples to make the target network misclassified.

The discriminatorDh is to ensure the generated image to remain in makeup style. Therefore, real makeup face photos are also usedas input to the discriminator with a mask to retain using eye regions only. The discriminator Dh is the same as the discriminatorDy. To train Dh, Dy is used as pre-train weights toinitialize Dh. The network is shown in figure 9.[14]

Ltarget GAN(H; Dh) = -(Ey~Py[log Dh(y))] +Ex~Px[log(1 - Dh(H(~ x)))])[14]

Above equation can be define as loss for GANs

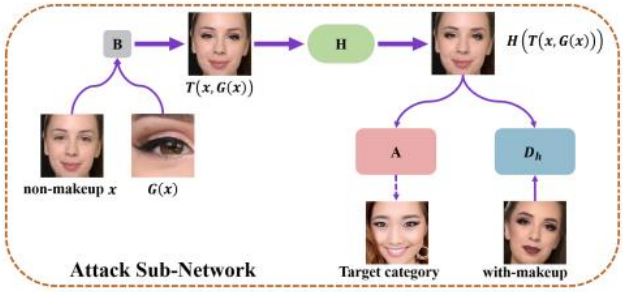

Figure 9 Proposed adversarial attacks sub-network[14]

Pascal Sch¨ottle, et.al.[15], Discuss in this paper author Adversarial classification is the task of performing robust classification in the presence of a strategic attacker. Which are slightly altered versions of benign images. They adopt a linear filter, similar to early steganalysis methods, to detect adversarial examples that are generated with the projected gradient descent (PGD) method, the state-of-the-art algorithm for this task. They are specifically crafted to be misclassified with a very high probability by the classifier under attack. in this paper that the detection of adversarial examples crafted against CNN-based classifiers can draw on long-established methods from steganalysis. Dataset was used in this paper is MNIST. The minimum accuracy of the combined approach over all parameters is 96%, almost at par with the accuracy of the tested CNNs for benign images.

Karim Armanious, et.al.[16]. In this work, they introduce the inpainting of medical images to complete missing or distorted information. This is beneficial for further image post-processing tasks, such as PET/MRI attenuation correction and

radiation therapy planning, rather than for diagnostic purposes. To achieve this goal, an adversarial framework is proposed which incorporates two patch-based discriminator networks and additional non-adversarial losses. Natural image inpainting widely uses to context encorder.they are based on an encoder-decoder network with an adversarial discriminative network. Many image inpainting techniques is used ,like CE (Context Encoder), GLCIC (Globally and locally consistent image completion), MedGAN (Medical Image),ip-MedGAN(Inpainting of arbitrarily region).

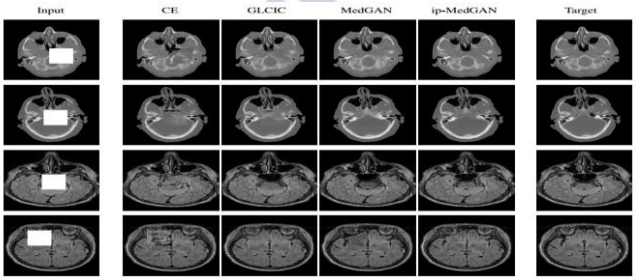

Figure 10 Qualitative comparison of the inpainting results between the proposed ip-MedGAN framework and other adversarial inpainting techniques. The first and last two rows represent inpainting of CT and MRI modalities respectively[16]

YANG YANG, et.al.[17], In this paper, they propose an effective method, named OCC-GAN, to address the one-class classi_cation problem. By introducing GAN architecture and modifying it with dense block structure, the proposed model is capable of concatenating and reusing multi-level features. To optimize the training process, several practical training strategies are applied in their work. Further more, as traditional evaluation metrics are unsuitable for measuring outliers, an index called CRI is designed to evaluate the performances on both targets and outliers. Experimental results demonstrate that model applies to the MNIST dataset and the SVHN dataset, and it is larger to some other one-class classification algorithms.[17]

HYUN KWON, et.al.[2], In this paper, they proposed a selective untargeted adversarial example with a 100% attack success rate and minimum misrepresentation. The proposed scheme creates a selective untargeted an adversarial example that is misclassified by model M as a wrong class other than specific avoided classes while maintaining a minimal misrepresentation distance from the original sample. The experimental results demonstrate that the proposed scheme can create a selective untargeted adversarial example with a 100% attack success rate and minimum misrepresentation is 1.325 and 34.762 using the MNIST and CIFAR-10 datasets, respectively, even when the number of avoided classes is five.

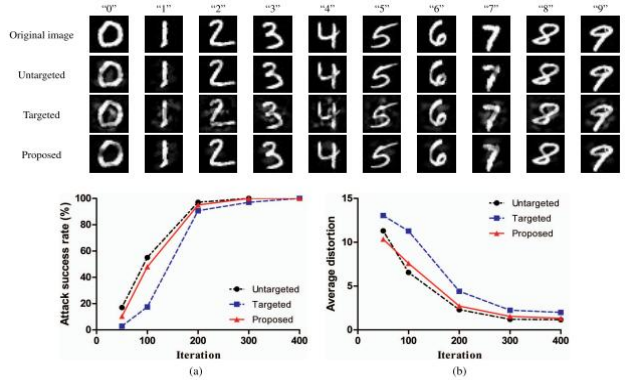

Figure 11 Random Targeted, untargeted attacks and proposed adversarial attacks on MNIST Dataset[2]

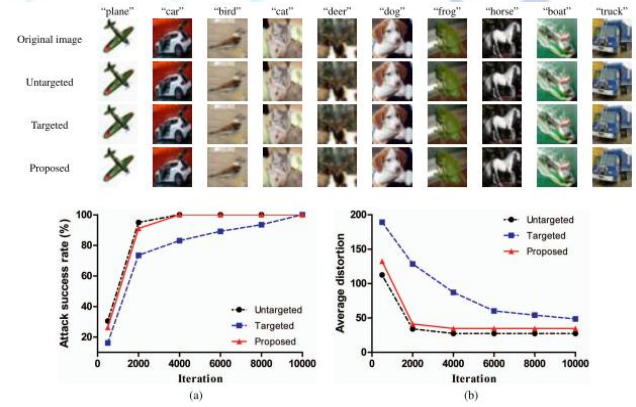

Figure 12 Random Targeted,untargeted attacks and proposed adversarial attacks on CIFAR-10 Dataset[2]

Table 1 Accuracy table of literature survey

Title	**Accuracy**		
Generating Adversarial Examples with Adversarial Networks	AdvGANs White-box attacks on MNIST	88.93%	
	AdvGANs black-box attacks on MNIST	92.76%	
Constructing Unrestricted Adversarial Examples with Generative Models	Dataset	w/o noise	w/noise
	MNIST	85.2	85.0
	SVHN	84.2	91.6
	CelebA	91.1	86.7
Generating Adversarial Examples By Makeup Attacks on Face Recognition	Erorr Rate	FGSM	PGD
	AVG	1.07	29.02

Detecting Adversarial Examples - a Lesson from Multimedia Security	Accuracy of combinig approach(Secrate and natural model) is 96%			
Adversarial Inpainting of Medical Image Modalities	Model	CT	MRI	
	IP-Med GAN	0.8346	0.3818	
One-Class Classification Using Generative Adversarial Networks	Model	OTR	F1 Score	Avg.Acc.
	OCC_GAN	0.1	0.9524	0.9091
Selective Untargeted Evasion Attack: An adversarial example that will not be classified as certain avoided classes	Dataset		Accuracy	
	MNIST		96%	
	CIFAR-10		90%	

V. Conclusion

In this paper, study about a developing of popular network of Generative adversarial network, and study about different methods applying on different Dataset some authors has used FGSM, PGD methods for adding random noise. From this two methods comparatively PGD method gives accurate result.

References

[1] A Gentle Introduction to Generative Adversarial Networks (GANs)." [Online]. Available: https://machinelearningmastery.com/what-are-generative-adversarial-networks-gans/. [Accessed: 01-Jan-2020].

[2] H. Kwon, Y. Kim, H. Yoon, and D. Choi, "Selective Untargeted Evasion Attack: An adversarial example that will not be classified as certain avoided classes," *IEEE Access*, pp. 1–1, 2019, doi: 10.1109/ACCESS.2019.2920410.

[3] J. Patel, M. Pandya, and V. Shah, "Review on Generative Adversarial Networks," vol. 4, p. 1230, Jul. 2018.

[4] M. L. @ Berkeley, "Tricking Neural Networks: Create your own Adversarial Examples," *Medium*, 07-Mar-2019. [Online]. Available: https://medium.com/@ml.at.berkeley/tricking-neural-networks-create-your-own-adversarial-examples-a61eb7620fd8. [Accessed: 01-Jan-2020].

[5] "Universal Adversarial Perturbations | Bosch Center for Artificial Intelligence." [Online]. Available: https://www.bosch-ai.com/research/research-applications/universal-adversarial-perturbations/. [Accessed: 01-Jan-2020].

[6] Generative Modeling - Generative Deep Learning [Book]." [Online]. Available: https://www.oreilly.com/library/view/generative-deep-learning/9781492041931/ch01.html. [Accessed: 06-Jan-2020].

[7] Adversarial example using FGSM | TensorFlow Core," *TensorFlow*. [Online]. Available: https://www.tensorflow.org/tutorials/generative/adversarial_fgsm. [Accessed: 06-Jan-2020].

[8] What is projected gradient descent and when do we prefer to use it over normal gradient descent? - Quora." [Online]. Available: https://www.quora.com/What-is-projected-gradient-descent-and-when-do-we-prefer-to-use-it-over-normal-gradient-descent. [Accessed: 06-Jan-2020].

[9] C. T. Bs. H. MIAP, "An introduction to Convolutional Neural Networks," *Medium*, 27-May-2019. [Online]. Available: https://towardsdatascience.com/an-introduction-to-convolutional-neural-networks-eb0b60b58fd7. [Accessed: 06-Jan-2020].

[10] M. Erofeev, "Image Inpainting: Humans vs. AI," *Medium*, 06-Nov-2018. [Online]. Available: https://towardsdatascience.com/image-inpainting-humans-vs-ai-48fc4bca7ecc. [Accessed: 06-Jan-2020].

[11] C. Xiao, B. Li, J. Zhu, W. He, M. Liu, and D. Song, "Generating Adversarial Examples with Adversarial Networks," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, Stockholm, Sweden, 2018, pp. 3905–3911, doi: 10.24963/ijcai.2018/543.

[12] Y. Song, R. Shu, N. Kushman, and S. Ermon, "Constructing Unrestricted Adversarial Examples with Generative Models," p. 12.

[13] Z.-A. Zhu, Y.-Z. Lu, and C.-K. Chiang, "Generating Adversarial Examples By Makeup Attacks on Face Recognition," in *2019 IEEE International Conference on Image Processing (ICIP)*, Taipei, Taiwan, 2019, pp. 2516–2520, doi: 10.1109/ICIP.2019.8803269.

[14] P. Schottle, A. Schlogl, C. Pasquini, and R. Bohme, "Detecting Adversarial Examples - a Lesson from Multimedia Security," in *2018 26th European Signal Processing Conference (EUSIPCO)*, Rome, 2018, pp. 947–951, doi: 10.23919/EUSIPCO.2018.8553164.

[15] K. Armanious, Y. Mecky, S. Gatidis, and B. Yang, "Adversarial Inpainting of Medical Image Modalities," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brighton, United Kingdom, 2019, pp. 3267–3271, doi: 10.1109/ICASSP.2019.8682677.

[16] Y. Yang, C. Hou, Y. Lang, G. Yue, and Y. He, "One-Class Classification Using Generative Adversarial Networks," *IEEE Access*, vol. 7, pp. 37970–37979, 2019, doi: 10.1109/ACCESS.2019.2905933.