Robustness against adversary models on MNIST by Deep-Q Reinforcement Learning based Parallel-GANs

Rong Zhang and Pao-Chi Chang
Department of Communication Engineering, National Central University, Taoyuan City, Taiwan
Email: rzhang.vaplab@gmail.com, pcchang.vaplab@gmail.com

Abstract*—This paper presents a method of enhancing robustness of classification machine learning model. Robustness of computer programming is an important topic, not only affects the security of user information but also stability and performance of program itself. That is, a technique aims to misclassify the machine learning model is called Adversarial Attack. We present an experiment to increase the robustness of machine learning models. The dataset MNIST, which includes hand-writing digits, is used as the experiment subject. Several techniques are applied such as Generative Adversarial Networks in Parallel form, Reinforcement Learning in Deep-Q formation, Dynamic sampling is used as prediction of unknown attack. Black-box is set to be experimental scenario. The experimental results show that the average robustness of the system under several attack conditions is as high as 90%.

***Index Terms*—Robustness machine learning, GANs, Q-Learning, MNIST**

I. Introduction

Deep learning includes but not limited to Neural Networks has achieve as effective classification or regression methods recent days. Like other programs, robustness machine learning is also considered as a critical issue [1], [2]. As the evolution of hardware, recent works show that certain attack methods are effective against machine learning models by adding perturbations to attack inputs [3] - [5]. These methods can let the target machine learning models misclassify the inputs, such as Basic Iterative Method (BIM), Fast Gradient Sign Method (FGSM) and DeepFool algorithm, also called adversarial attack [6], [7].

Fig. 1 shows the attack from FGSM, the main goal of adversarial attack is adding a perturbation or noises to misclassify the target models but meanwhile stays visual realistic [8], [9]. By approaching this, those methods are limited by a coefficient and by verifying the Maximum distance between the output and input [10]. Although remain visual realistic also limits the effectiveness of attack, the methods mentioned above can still achieve at most 99% of attack success rate on un-protected machine learning models [11] - [13].

That is, adversarial attack has been considered as a critical threat of information security. To solve this problem, enhance the robustness of the machine learning models is necessary [14], [15]. We assume a scenario of unknown attack, which is black-box attack, need to be dealt with. A robust model is trained to encounter the attack [16], [17]. Since the attack remains un-detected, we can only predict them by generating random noises. In this case, the optimal methods that mentioned above are not applicable. We apply Generative Adversarial Networks (GANs) [18] to adopt this situation. Since GANs are capable of generating specific target we need it to. We apply similar structure to our program, in order to predict the attack.

Note that our GANs is in *Parallel* form, means there exist multiple generators with the losses associated with. That allows the network to select the most fitting sub-networks by applying algorithm like Evolutionary Learning or Reinforcement Learning [19].

To evaluate our model performance, we generate several adversarial examples, which are the output of attack, from the above and several state-of-the-art methods. We then test our model accuracy for the robustness training, in order to evaluate the prediction of attack. We show that the model is comparably more robust than the others, which remains high accuracy during the attack.

We also test our model performance in conventional method such as Optical Character Recognition (OCR) [20], the most commonly used method in identification system at the place like parking lots. The system can be misled by certain noises which will lead misclassification such as vehicle registration plate number.

The contributions of our work are listed below.

- We use specific form of GANs to predict the unknown attack, different from passive training of regular methods.
- We show that the proposed method can handle black-box attack, which means we do not need to know what attack looks like.
- We apply reinforcement learning to our model, which further increase the efficiency of model training, compare to the random selection of Evolutionary Learning.
- We test our model compared with conventional OCR method, the result indicates our model is more capable of defending unknown attack.

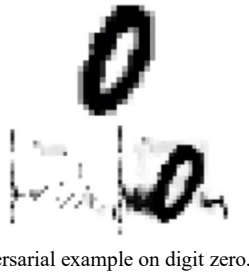

Fig. 1: The adversarial example on digit zero. First row: The original image. Second row: Distortion generated by FGSM (left), Adversarial Example by adding the distortion to original image (right).

II. Related Work

A. Parallel-GANs

Yamamoto *et al.* have announced a structure with multiple losses based GANs, which contains selection of various generators [21]. Due to the fact we need to generate the unknown perturbations, the feature extraction of this network is considered practical. Different from the prior work, we apply multiple generators to remain independency instead of generating stacked multiple resolution features. An image-to-image networks is used in this work. By adding adaptative distortion to the targets, we can achieve the goal of attack.

B. Deep-Q Networks

Wang *et al.* from Google issued a structure with Reinforcement Learning based Deep Neural Networks [22].

Most of the optimization methods used in regular deep learning are based on minimizing the loss from certain function, e.g. for regular cases, cross-entropy. By calculating the differential functions, we can navigate the model to approximate the loss function. Reinforcement Learning such as Q-Learning or Temporal Difference can predict the state, which gives us the ability to choose and estimate the performance of the sub-networks. Reinforcement Learning can approximate the minimum loss by given loss functions instead of constructing Q-Tables. In this experiment, cross-entropy is given as the loss function. We apply this to our work, make it further efficient than regular one.

C. Black-Box Attack

As the security has been gradually valued recent years, most of the machine learning models are restricting users to access their entries. The same as attacking, attackers will not leak their information to keep their attack effective. Therefore, a scenario of black-box is necessary [23]. Lukas *et al.* have set several conditions of the black-box attack, prevent the attacker's information from leaking [24], [25].

We follow the same setting, perform the black-box attack. By predicting the unknown perturbation, we can show that the proposed method is effective.

III. Methodology

A. System Approach

Suppose we have a set of clean input: $X \subseteq D$, where D denotes the dataset. Then for the clean label: $Y \subseteq R^D$, where R denotes the feature space of the associated dataset, and then for the attack method, we have an existed threat model $f_{threat}(x) \neq y$, as Fig. 2 shows below.

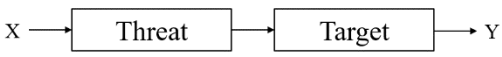

Fig. 2: Ideal blocks for Attack

The goal of our work is to effectively defend the misclassify from the Adversarial Attack. We train a robust model to encounter this problem [26]. That is, $f_{robust}(x) = y$. Then we test the accuracy of the model to evaluate the robustness. In this case, higher accuracy indicates higher robustness. The following figure, Fig. 3, shows the approach of our work.

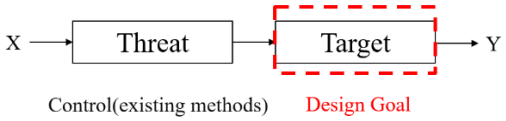

Fig. 3: Ideal blocks for robust training

As we mention above, we design a robust target to encounter the unknown threat which is existing methods such as BIM, MIM, or FGSM [28].

B. Main Structure

Fig. 4 shows the overall structure of the robust model in training stage.

Fig. 4: Stage of robust training

The structure contains three parts, which are Multi-Generators G_i, Deep-Q, and robust evaluation target f, respectively. In order to predict the unknown attack, we put random white noises as noisy features and clean inputs as auxiliary features as reference. Different from the previous work of training GANs, we set discriminator D as a clean model with non-robust training. Fig. 5 shows the detailed components of robust training.

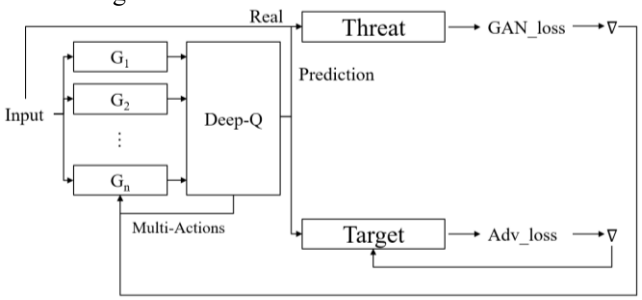

Fig. 5: Main structure of robust model training

As shown in Fig. 5 above, the threat does not need any requirement or feedback, indicating the attack is black-box.

The Adversarial loss is:

$$L_{GAN} = E_x \log f_{threat}(x) + E_x \log\left(1 - f_{threat}(G_i(x) + x)\right). \quad (1)$$

In this work, the discriminator is a target to evaluate the robustness by testing the effectiveness of distinguishing clean and attack inputs.

The Discriminator loss is:

$$L_D = E_x l_f(G_i(x) + x,\ t). \quad (2)$$

where l_f denotes the loss function of robust model. By minimizing the loss of discriminator, we can see the model predict accurately during attack.

As shown in Fig. 4, we have reinforcement learning applied

as selection of the model. The Deep-Q block also generate loss which is adaptative.

The RL loss is:

$$L_{RL} = \max_{q} Q\acute{}\left(s(t), a(t)^{(1)} \dots a(t)^{n}\right). \tag{3}$$

We use epsilon-greedy as the learning policy, means the model will always follow the biggest profit based on current state.

$$Q \leftarrow \operatorname*{argmax}_{q} Q\acute{}\left(s(t), a(t)\right). \tag{4}$$

By calculating each generator neurons hidden output, we can estimate the action by approximating the given loss function l_f.

To estimate the full loss, we combine (1), (2), (3):

$$L = \alpha L_{GAN} + L_D + \beta L_{RL}. \tag{5}$$

where α and β are adjustable coefficients for the importance of each learning blocks. Note that we need to train model to anticipate the attack, so we need to optimize the full loss by choosing the minimum [27], [28]. Fig. 6 shows the block diagram of Deep-Q in hierarchical form in this experiment [29].

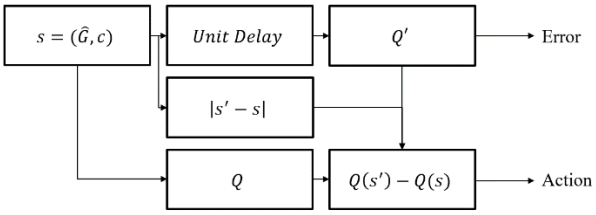

Fig. 6: Block diagram of hierarchical Q function

C. Black-box

We assume the condition we will face is black-box attack, that is unknown perturbation will be added to the clean image [30]. In order to predict the attack, we use white noises to generate certain perturbation by random sampling similar as Gibbs-sampling in Restricted Boltzmann Machine (RBM) [31]. The step is showed in Algorithm 1:

% Feature sampling
for step = 1: N do

$$P_{gen}^{step} = P_{D}^{step-1} P_{gen|D}$$
$$P_{D}^{step} = P_{gen}^{step-1} P_{D|gen}$$

Algorithm 1: Random sampling of training model

By given certain conditional probability, we can approximate the initial distribution by iteration method. Like Gibbs-sampling, by passing amount of epoch, final distribution can be updated as the original one, which is the balance between generators and discriminator.

The given conditional probability for the generator is:

$$P_{gen}^{step} = \operatorname*{argmax}_{gen} \min_{D} L_D + \alpha L_{GAN} + \beta L_{RL}. \tag{6}$$

Which is also greedy policy as reinforcement learning that mentioned above. By selecting the least ones of maximum full objective loss, we can optimize the model as a robust generator.

By updating the distribution of generators, we can approximate the one from the discriminator.

Also, the given conditional probability for the discriminator is:

$$P_{D}^{step} = \max_{k \in row} \sum_{k} \left| f_D(x) - f_{gen}(x) \right|. \tag{7}$$

By calculating the maximum distance between output and input, we can obtain loss for the discriminator.

The optimization method can reach balance between the generator and the discriminator. To generate the predicted perturbation, we use white noise as input. In this section, we use a table works like quantization to favor the balance between generators and discriminator.

Since the generator uses noise to predict the attack, it chooses the second-likely loss from original clean prediction. The random sampling then maintains the balance between the generator and the discriminator by calculating the loss. Theoretically, according to Gibbs-sampling, the distribution will approximate the initial condition of competition between GANs. That is, the trade-off between the generator and the discriminator.

In this method, the generator tries to anticipate the ground truth from existing third-party methods. The discriminator then is simplified by quantizing the images to maintain the competition of sub-networks in GANs. Fig. 7 shows the reconstruction of the model in order to defend the attack.

Fig. 7: Reconstruction from discriminator to defend the attack. First row: original image. Second row: reconstruction image

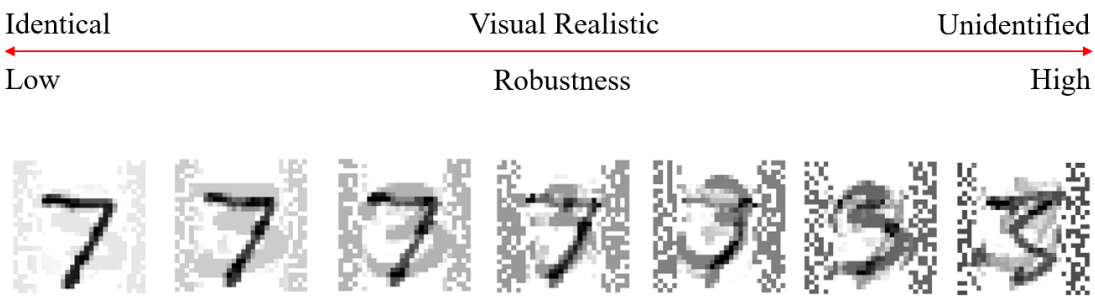

Fig. 9: Trade-off between distance and robustness

IV. Experimental Result

A. Dataset

In this experiment, we use MNIST [32], [33] from National Institute of Standards and Technology (NIST) to evaluate our model performance. This dataset contains 70,000 hand-writing digits includes 60,000 of training set and 10,000 of testing set, each of 784 pixels with 8 bits greyscale level. The dataset consists 0-9 hand-writing digits from 250 different people, with equal number of 6000 each. Examples are shown as in Fig. 8.

Fig. 8: Clean examples from original MNIST dataset [32], [33]

B. Implementation

We first apply the proposed model with the training set. We use batch size of 256 and learning rate of 0.01. We then apply the discriminator with three convolution layers and two fully connected layers. According to Yamamoto *et al.,* we use leaky-ReLU as our activation with $\delta = 0.2$ in (5) [34]. In order to accelerate the processing of the networks, we use weight normalization to all layers in the framework [35].

For training stage, we use 60,000 of clean image from the training set. Then we apply inputs with random noise as main feature and clean images as auxiliary features. The noise is:

$$X_n \sim N(0,1). \tag{8}$$

The β is set to be 0.1 based on experience from several experiment. For the conditional probability in Algorithm 1, we use the sampling from each layer between iterations.

$$P_{G|D} = \aleph\left(\sigma\left(W_j^i x^i + b_j\right)\right). \tag{9}$$

where $\aleph$ is normalization and σ is activation function, in this case, Leaky-ReLU. We use similar computation as Gibbs-sampling to approximate the final distribution, which is the balance between the generator and the discriminator. The following equation shows the activation σ:

$$\sigma(x) = \begin{cases} x, & x > 0 \\ \delta x, & x \leq 0 \end{cases}. \tag{10}$$

In this case, $\delta = 0.2$ as we mention above.

For testing stage, we generate a set of 10,000 adversarial examples, which is from the testing set of MNIST. According to Goodfellow *et al.*, the L_∞ distance ϵ is set between 0 and 1. We set $\epsilon = 0.3$ as the L_∞ distance between original and poisoned images. Fig. 10 shows the block diagram the adversarial examples from FGSM [36].

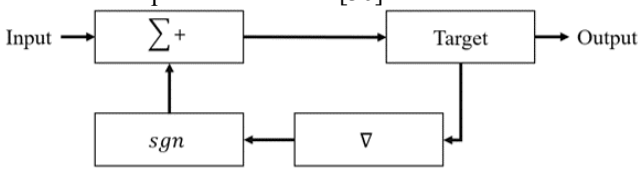

Fig. 10: Block diagram of FGSM

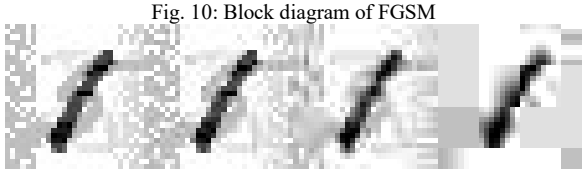

Fig. 11: Output of balance between generator and discriminator, from above to below, the model favors from discriminator to generator

Fig. 12: The testing set generated by state-of-the-art (FGSM), compare with the original and Adversarial examples

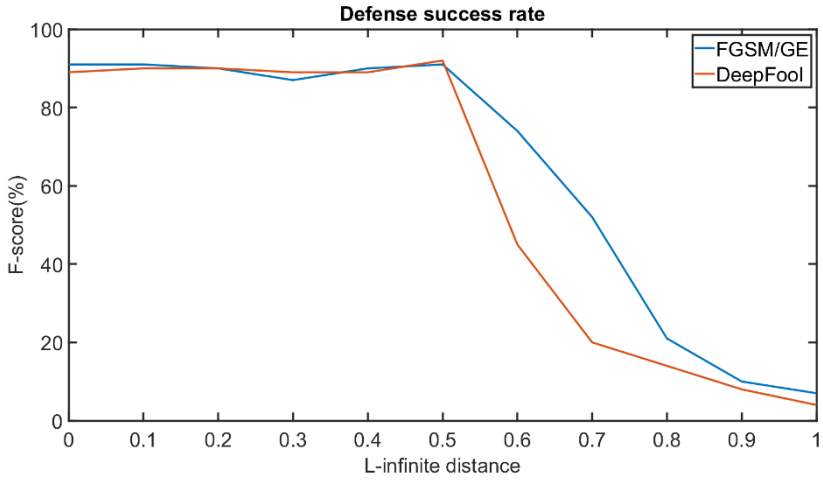

Fig. 13: Robustness of the model under two attack method, FGSM (blue) and DeepFool (orange)

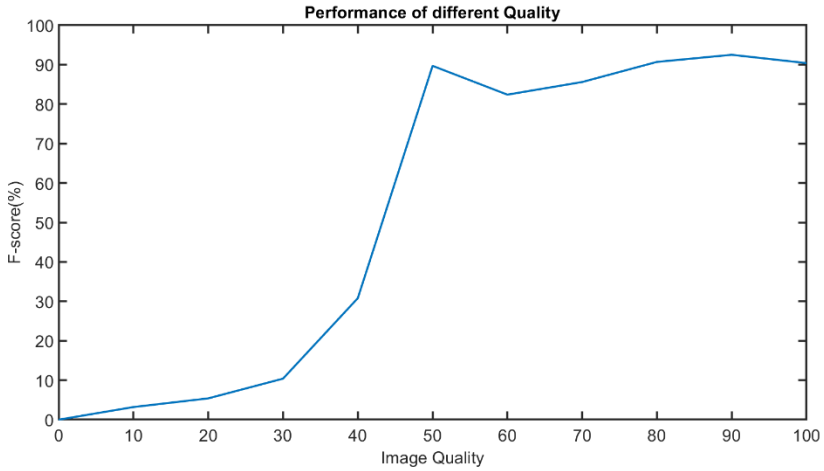

Fig. 14: The effectiveness of model under different balance of generator and discriminator

We use the block from Fig. 6 and DeepFool method [37] to generate effective adversarial examples, e.g. the examples from Fig. 12. Each example with maximum distance 0.3 according to Goodfellow *et al*.

C. *Trade-off*

As we mention in section I, there exists a coefficient ϵ limits the maximum distance (Chebyshev distance) L_∞ between the adversarial examples and original images. The following equation shows the computation of maximum distance [38].

$$L_\infty = \max_k(X_k^{'} - X_k). \tag{11}$$

The larger the distance is, the farther away from visual realistic, means it is more difficult for human eyes to identify. Although increasing the toleration of the distance can improve the robustness of the model, it also decreases the realistic of images, making them hardly identified. In this experiment, the sub-network in GANs compete with each other to generate the perturbation from clean inputs. Fig. 9 shows the trade-off between these two factors, distance (distortion) and robustness.

As shown in Fig. 9 from above, the original image is number 7, after training it predicts as number 3. For the robustness, the larger distance, which is controlled by epsilon, the more robust it is. However, the image is completely unidentified as 7, means the image is ineffective for defending. That is, finding a middle point to remain balance of the two factors is crucial.

As Fig. 11 shows, for visual realistic, discriminator tries to predict the result from clean input. That is, the outcome of discriminator is closer to simplify the noise from the generator. For robustness, the generator predicts more noise from original inputs, the result comes with distorted images with maximum distance limit. As Fig. 14 shows, we choose sampling quality as 50% to present the best effectiveness of the model.

D. *Performance*

We first evaluate our model under different attack methods. In this experiment, we apply two methods, which are FGSM and DeepFool. Both methods process with numerical gradient estimation to accelerate generating perturbation. The following figure shows the robustness of the model after inserting the adversarial examples from the attack methods mentioned above.

The evaluation we use is F_1 score (F-measure), it contains two factors, precision p and recall r.

The precision is:

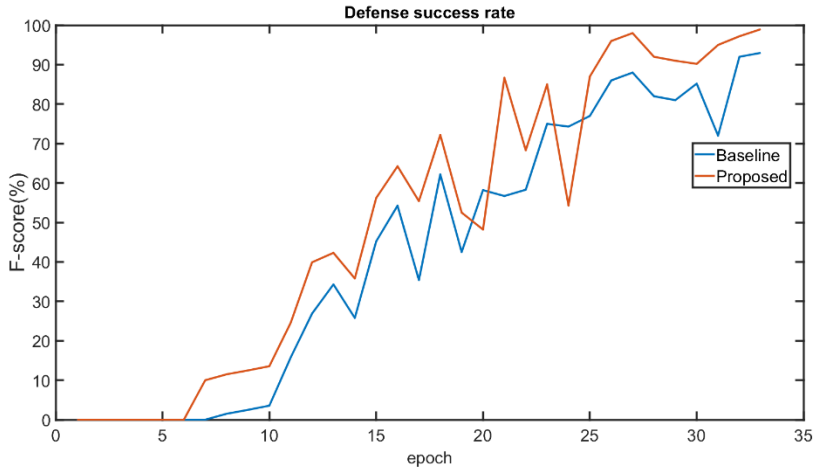

Fig. 15: Robustness between Proposed and Baseline methods

TABLE I
THE RESULT BETWEEN CNN, OTHERS AND PROPOSED METHOD

	CNN	Madry Lab.	Binary ABS	Proposed
Clean	99.1	98.8	99	98
FGSM/GE	0.1/21	0.48/89	0.49/85	0.54/91
DeepFool/GE	0.09/0	0.53/90	0.46/78	0.55/89
All Attacks	0.95/10.5	0.5/89.5	**0.475**/81.5	0.545/**90**

$$\mathrm{p} = \frac{TP}{TP + FP}. \tag{12}$$

represents the condition that model identify the result as true.

The recall is:

$$\mathrm{r} = \frac{TP}{TP + FN}. \tag{13}$$

represents the condition of the input data as true.

The equation of F-score is:

$$\mathrm{F} = \frac{(1 + \phi^2)(p \times r)}{\phi^2(p + r)}. \tag{14}$$

Note that F-score is affected by a parameter ϕ in (14). Usually by our evaluation, we set it as $\phi = 1$ so the parameter can be neglected.

As Fig. 13 shows, we can obtain that the model efficiency decreases after the distance passes 0.5. That is, our model can handle the distortion within 0.5. From section IV-C, we can see the trade-off between the attack and defense, the result from the figure above shows the model is able to increase the robustness.

The adversarial examples with maximum distance over 0.5 are highly unidentifiable by human eyes. That is, defending the images above certain distance is not necessary since the goal of adversarial attack is remaining visual realistic meanwhile effectively misclassify the machine learning models.

We implement the model *binary-ABS* from Lukas et al. as the baseline, then we evaluate the robustness during the training stage. The following figure shows the result of robustness by successfully classifying the predicted attack.

As the Fig. 15 above, the proposed method (orange) is comparably upper than baseline (blue). Indicates the proposed method is more efficient to anticipate the attack. Note that after 30 epochs the proposed can achieve close to 98%, means it can almost defend every attack from generators. The existing methods count as effective do not participate as a role in this section, that is, the experiment scenario is a black-box

After the training stage, we enter testing stage. By sending the examples like Fig. 10, we evaluate the robustness of proposed model and baseline one. The following table shows the result of proposed and other existing models' performance under certain effective attack.

As Table I shows, the proposed method is comparably higher robustness than others. Note that as pervious section mentions, there's a trade-off between the distance and robustness. As the result, we can obtain this phenomenon clearly.

TABLE II
THE CLASSIFICATION BETWEEN CNN, OCR AND PROPOSED METHOD

Model	**CNN**	**OCR**	**Proposed**
Result	623	523	**578**

TABLE III
THE AVERAGE ACCURACY BETWEEN CNN, OCR AND PROPOSED METHOD

Model	CNN	OCR	Proposed
Acc. (MNIST)	0	12	**90**
Acc. (Vehicle number)	5	20	**90**

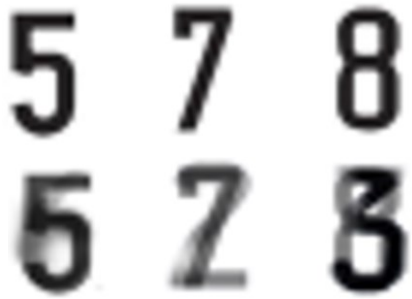

Fig. 17: One set of attacking on a vehicle number plate from 578 to 623

In this experiment, we also test the robustness of current applications, in this case, Optical Character Recognition (OCR). The system is designed for identifying numbers from actual world like vehicle registration plate, which is commonly used in place such as parking lot. The core of this system is not machine learning since it's been developed in early 1990, the hardware is not strong enough to handle computation from machine learning models [39]. On the contrary, OCR uses conventional methods like character cropping and compare the most-likely symbol with a default set database. That is, the system is fixed, means it has more vulnerabilities than machine learning.

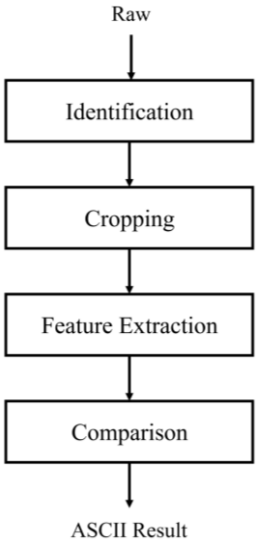

Fig. 16: Block diagram of Optical Character RecognitionFig. 16 shows the block diagram of OCR. OCR uses traditional methods to identify and classify character from raw images, it has fixed database to compare the most-likely calculation of alphabets and digits. The algorithm uses cropping to separate the character from objects such as vehicle registration plates [40]. Then it compares the feature with its own database, by calculating the most-likely objects, the algorithm chooses the output in ASCII or UNICODE.

Note that the scenario of OCR is set noise-free, indicates it's more likely misclassify in realistic. We generate 20 sets of corrupted data by using FGSM and evaluate the performance between proposed, CNN and OCR methods.

Fig. 17 shows one of the number sets generated by the attack method, using the effective method to generate an example of attack vehicle registration plate. We test the proposed, clean CNN and conventional OCR.

Table II shows one of the 20 classification results of the adversarial attack. CNN and OCR both misclassify the input attack. The proposed method still remains correct. That is, the proposed method has comparably strong robustness. And Table III shows the average accuracy after the methods classify the attach inputs generated by the attack methods. We generate 20 data as we mention above, and the accuracy is the average correctness of whole experiment. We also issue a threat that indicates the intentionally attack on conventional methods, in this experiment, OCR.

V. CONCLUSION

In this paper, we propose a method with several innovations to increase the robustness of machine learning models. In our model, we use GAN as main framework, reinforcement learning as model optimization and random sampling as predicting the unknown attack. We verify the performance by using several state-of-the-art and application to test the accuracy of the model. The results show that the proposed method is effective and efficient. By this experiment, we also notify there's potential issue of security in machine learning.

REFERENCE

[1] Anish Athalye, Nicholas Carlini, and David Wagner, "Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples.", arXiv preprint arXiv:1802.00420, 2018.

[2] W. Brendel and M. Bethge, "Comment on "biologically inspired protection of deep networks from adversarial attacks".", arXiv preprint arXiv:1704.01547, 2017.

[3] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus, "Intriguing properties of neural networks.", In *International Conference on Learning Representations,* 2014.

[4] Shumeet Baluja and Ian Fischer. "Adversarial transformation networks: Learning to generate adversarial examples.", arXiv preprint arXiv:1703.09387, 2017.

[5] Metzen, J. H., Genewein, T., Fischer, V., and Bischoff, B, "On detecting adversarial perturbations", In *International Conference on Learning Representations*, 2017.

[6] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy, "Explaining and harnessing adversarial examples." In *International Conference on Learning Representations*, 2015.

[7] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song, "Delving into transferable adversarial examples and black-box attacks.", In *International Conference on Learning Representations*, 2017.

[8] Wieland Brendel, Jonas Rauber, and Matthias Bethge, "Decision-based adversarial attacks: Reliable attacks against black-box machine learning models.", In *International Conference on Learning Representations*, 2018.

[9] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Dan Boneh, and Patrick McDaniel, "Ensemble adversarial training: Attacks and defenses.", arXiv preprint arXiv:1705.07204, 2017.

[10] Sinha, A., Namkoong, H., and Duchi, J., "Certifiable distributional robustness with principled adversarial training.", In *International Conference on Learning Representations*, 2018.

[11] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami, "Practical black-box attacks against deep learning systems using adversarial examples." *arXiv preprint*, 2016.

[12] Anh Nguyen, Jason Yosinski, and Jeff Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images.", 2015.

[13] Anh Nguyen, Jason Yosinski, and Jeff Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images.", In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[14] Alexey Kurakin, Ian Goodfellow, Samy Bengio, Yinpeng Dong, Fangzhou Liao, Ming Liang, Tianyu Pang, Jun Zhu, Xiaolin Hu, Cihang Xie, et al, "Adversarial attacks and defences competition." arXiv preprint arXiv:1804.00097, 2018.

[15] Moustapha Cisse, Piotr Bojanowski, Edouard Grave, Yann Dauphin, and Nicolas Usunier, "Parseval networks: Improving robustness to adversarial examples.", arXiv preprint arXiv:1704.08847, 2017.

[16] Eric Wong and Zico Kolter, "Provable defenses against adversarial examples via the convex outer adversarial polytope.", In *International Conference on Machine Learning,* pages 5283–5292, 2018.

[17] Shixiang Gu and Luca Rigazio, "Towards deep neural network architectures robust to adversarial examples.", *arXiv preprint arXiv:1412.5068*, 2014.

[18] Ian Goodfellow, Jean PougetAbadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets.", In *Conference on Neural Information Processing Systems*, pages 2672–2680, 2014.

[19] Hanjun Dai, Hui Li, Tian Tian, Xin Huang, Lin Wang, Jun Zhu and Le Song, "Adversarial Attack on Graph Structured Data.", In *International Conference on Machine Learning*, 2018.

[20] J. Memon, M. Sami, R. A. Khan and M. Uddin, "Handwritten Optical Character Recognition (OCR): A Comprehensive Systematic Literature Review (SLR)," in *IEEE Access*, vol. 8, pp. 142642-142668, 2020, doi: 10.1109/ACCESS.2020.3012542.

[21] Ryuichi Yamamoto, Eunwoo Song and Jae-Min Kim, "Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram", In *International Conference on Acoustics, Speech, & Signal Processing*, 2020.

[22] Ziyu Wang, Tom Schaul, Matteo Hessel, Hado van Hasselt, Marc Lanctot , and Nando de Freitas, "Dueling Network Architectures for Deep Reinforcement Learning", *arXiv preprint*, 2015.

[23] Andrew Ilyas, Logan Engstrom, and Aleksander Madry, "Prior convictions: Black-box adversarial attacks with bandits and priors.", arXiv preprint arXiv:1807.07978, 2018.

[24] Lukas Schott, Jonas Rauber, Matthias Bethge, and Wieland Brendel, "Towards the first adversarially robust neural network model on MNIST", In *International Conference on Learning Representations*, 2019.

[25] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami, "Practical black-box attacks against deep learning systems using adversarial examples.", arXiv preprint, 2016.

[26] Papernot, N., McDaniel, P., Wu, X., Jha, S., and Swami, A., "Distillation as a defense to adversarial perturbations against deep neural networks.", In *Security and Privacy (SP), 2016 IEEE Symposium on*, pp. 582–597. IEEE, 2016.

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

[28] Hein, M. and Andriushchenko, M., "Formal guarantees on the robustness of a classifier against adversarial manipulation.", In *Advances in Neural Information Processing Systems*, 2017.

[29] Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., and Swami, A., "Practical black-box attacks against deep learning systems using adversarial examples.", In *Proceedings of the 2017 ACM Asia Conference on Computer and Communications Security*, 2017.

[30] Siyuan Li, Rui Wang, Minxue Tang, and Chongjie Zhang, "Hierarchical Reinforcement Learning with Advantage-Based Auxiliary Rewards", In *NeurIPS*, 2019.

[31] Behnoush Abdollahi and Olfa Nasraoui, "Explainable Restricted Boltzmann Machines for Collaborative Filtering", In *ICML Workshop on Human Interpretability in Machine Learning (WHI 2016)*, 2016.

[32] Yann LeCun and Corrina Cortes. The MNIST database of handwritten digits. 1998.

[33] Feiyang Chen, Nan Chen, Hanyang Mao, and Hanlin Hu, "Assessing Four Neural Networks on Handwritten Digit Recognition Dataset (MNIST)", In *CHUANGXINBAN JOURNAL OF COMPUTING*, 2018.

[34] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li, "Empirical Evaluation of Rectified Activations in Convolutional Network", *arXiv preprint*, 2015.

[35] T. Salimans and D. P. Kingma, "Weight normalization: A simple reparameterization to accelerate training of deep neural networks," In *Proceedings of Neural Information Processing Systems*, 2016, pp. 901–909.

[36] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok, "Synthesizing robust adversarial examples.", In *International Conference on Machine Learning*, pages 284–293, 2018.

[37] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard, "DeepFool: a simple and accurate method to fool deep neural networks", In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[38] Peck, J., Roels, J., Goossens, B., and Saeys, Y., "Lower bounds on the robustness to adversarial perturbations.", In *Advances in Neural Information Processing Systems*, pp. 804–813. 2017.

[39] Muhammad Tahir Qadri, Muhammad Asif, "Automatic Number Plate Recognition System for Vehicle Identification Using Optical Character Recognition", In *International Conference on Education Technology and Computer*, 2009.

[40] Lu, J., Sibai, H., Fabry, E., and Forsyth, D., "No need to worry about adversarial examples in object detection in autonomous vehicles.", *arXiv preprint arXiv:1707.03501*, 2017.