ADVERSARIAL EXAMPLES DETECTION BASED ON ERROR LEVEL ANALYSIS AND SPACE MAPPING

Sizhao Huang *† *Shuai Wang* *† *Jian Chen* *† *Guozhi Li* *† *Wenyi Wang* *

* School of Information and Communication Engineering University
of Electronic Science and Technology of China
†Yangtze Delta Region Institute of University
of Electronic Science and Technology of China

ABSTRACT

Deep neural network (DNN) shows impressive performance on many tasks but they usually suffer from adversarial examples with human eyes invisible slight perturbation. Such examples can not be distinguished by human but can mislead DNN classifiers leading to its important role in DNN attack and defense. Many adversarial examples detection methods perform well in identifying global perturbation adversarial examples but less efficiently for local perturbation ones. We observe both global perturbation and local perturbation adversarial examples have similar BOF histogram distribution after JPEG compression and Error Level Analysis (ELA) while these distributions are clearly different to clean example's distribution. Meanwhile, researchers have found that the stability of adversarial example after space mapping is worse than that of the clean example. Therefore, we propose a two-branch architecture to detect adversarial examples based on the aforementioned strategies. Experiments show that our method has achieved better or similar performance compared to several state-of-the-art methods in terms of the detection accuracy and generation property for adversarial examples with global and local perturbation.

Index Terms— Global perturbation, Local perturbation, Adversarial examples detection, Generalization property

1. INTRODUCTION

In recent years, deep learning has shown impressive performance in many tasks [1] such as face recognition, speech recognition, video classification, text classification, etc. However, the vulnerability of deep learning applications has attracted people's concern. Attackers can mislead the classifier through adversarial examples [2], which may bring many potential security issues. In addition, adversarial examples have cross-model generalization capability [2] and can be generated without understanding sample itself.

In order to solve these security problems of deep learning applications, many thought-provoking detection works have been proposed. Feature Squeezing (FS) [3] discriminates through the change of prediction of transformed samples. Local Intrinsic Dimensionality (LID) [4] confirms LID characteristics can be used to detect adversarial examples. Mahalanobis Adversarial Detection (MAD) [5] identifies adversarial examples based on the mahalanobis distance. Spatial Rich Model (SRM) [6] regards adversarial examples as a kind of accidental steganography and detects them by enhancing the steganographic features. Pixel Artifacts and Confidence Artifacts (PACA) [7] uses pixel artifacts and confidence artifacts to detect adversarial examples from the perspective of image stream and gradient stream respectively. However, many methods only achieve good performances in detecting global perturbation (GP) adversarial examples but the detection accuracy and generalization property for novel local perturbation (LP) adversarial examples are not good enough. This guides us to carry out this research.

JPEG compression are often used as a method for feature distillation [8] and recovery [9] [10] in adversarial example area. We find it is also an effective preprocessing method to discriminate the difference between clean examples and adversarial examples. Specifically, the difference is captured by Error Level Analysis (ELA) [11] in this paper.

Inspired by PACA [7], we propose a two-branch architecture to extract and analyze the local and global image features. The local features are extracted from compressed images by using SIFT-based Bag of Features (BOF) model. And global features analysis branch detects adversarial examples by comparing label change and probability divergence between the variants generated by space mapping methods and the original one.

In numerical experiments on ImageNet and Caltech-256 dataset, the detection accuracy and generation property of novel LP and GP adversarial examples outperforms several state-of-the-art detection methods in most cases, especially in terms of generalization property.

This work is supported in the special science foundation of Quzhou, 2020D008, 2021D023. Sizhao Huang and Shuai Wang contribute equally to this work. Wenyi Wang is the corresponding author.

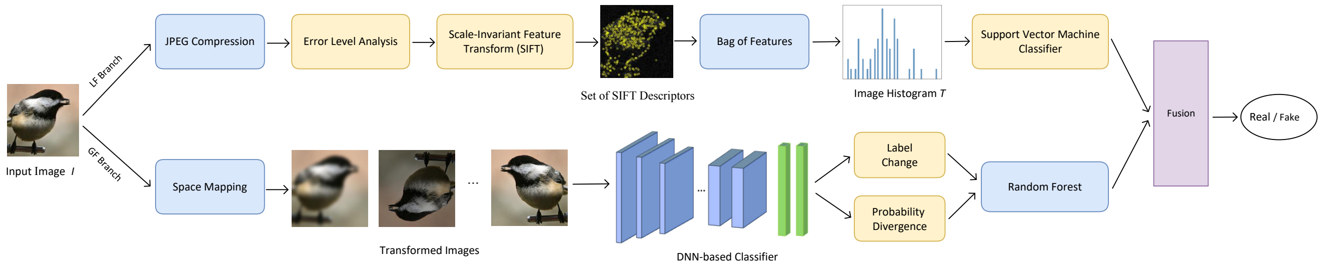


Fig. 1. The framework of our proposed algorithm which consists of Local Features analysis (LF) branch and Global Features analysis (GF) branch. LF branch makes the ELA for the input sample after JPEG compression and exploits BOF model to generates histogram for classification. GF branch detects adversarial example by comparing label change and probability divergence between the original one and the variants generated by space mapping methods.

2. PROPOSED METHOD

Fig.1 presents the proposed detection architecture which is composed of two branches and a fusion module. Local Features analysis (LF) branch makes the ELA for input samples after JPEG compression and then extracts SIFT features for classification. Global Features analysis (GF) branch identifies adversarial examples through space mapping (such as flip, twirl blur, gaussian filter, etc.) and comparing label change and probability divergence between the original one and variants generated by space mapping methods to get the detection score. The last part is the score fusion module, which merges the detection scores of two branches to obtain the final result of discrimination.

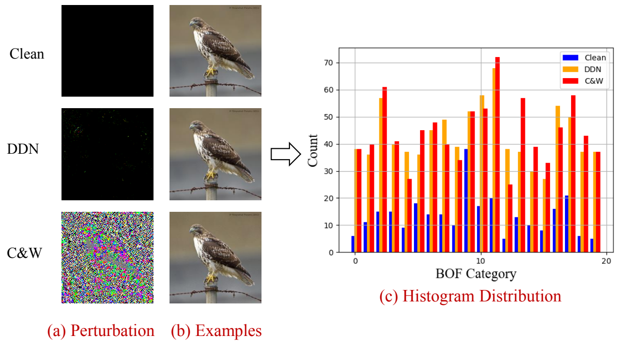
2.1. Local Features Analysis Branch

In the local feature detection module, we regard JPEG compression as an effective preprocessing module. As described by Das et al. [9], JPEG compression can remove high frequency components in the image block. This helps to remove additional perturbation and narrows the gap between GP adversarial examples and LP adversarial examples.

We make the ELA for input samples after JPEG compression. It divides the image into blocks and then perform a separate color space conversion for each small block. Repeated textures are emerged with similar colors. After this step, we can get the texture representation of the compressed image in the form of gray scale image.

Then we make texture analysis based on SIFT-based BOF model. In the training phase, SIFT features of training adversarial and clean examples are clustered to generate a BOF visual code book using k-means. According to the code book, both BOF histograms of adversarial and clean examples are generated and put into SVM to classify the adversarial and clean class.

Fig.2 shows clean, DDN [12] attack and C&W [13] attack adversarial example and their BOF histogram after JPEG compression and ELA. The perturbation images of DNN and C&W are also shown in Fig.2 (a) in normalized means as they are actually invisible in Fig.2 (b). It can be observed that there is a clear difference between clean example and both adversarial examples. In addition, the adversarial examples from DDN and C&W have similar BOF distribution.



(a) Perturbation (b) Examples

(c) Histogram Distribution

Fig. 2. The BOF histogram distribution comparison between clean example and adversarial examples. (a) shows the perturbations. (b) shows the clean example and adversarial examples which are difficult to differentiate by human eyes. (c) shows that BOF distributions of adversarial examples are clearly different to the clean example's distribution.

In the inference phase, the BOF histogram of the input image x's SIFT features is generated based on the code book. Then the adversarial probability ($F_L(x)$) of the input is gotten using the trained SVM based on its BOF histogram.

2.2. Global Features Analysis Branch

In the global features detection module, inspired by the observation that the stability of the adversarial example after space mapping is not as good as that of the clean example [14], we design a detection framework based on space mapping shown in Fig.3.

We define the original input image and space mapping method as x and Φ. For s space mapping methods, we can

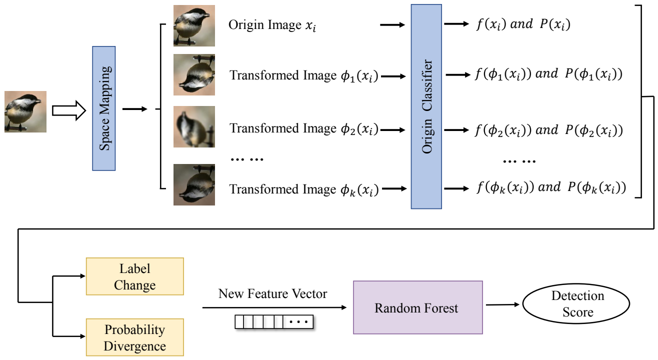

Space Mapping
Origin Image x_i
Transformed Image $\phi_1(x_i)$
Transformed Image $\phi_2(x_i)$
……
Transformed Image $\phi_k(x_i)$
Origin Classifier
$f(x_i)$ and $P(x_i)$
$f(\phi_1(x_i))$ and $P(\phi_1(x_i))$
$f(\phi_2(x_i))$ and $P(\phi_2(x_i))$
……
$f(\phi_k(x_i))$ and $P(\phi_k(x_i))$
Label Change
Probability Divergence
New Feature Vector
Random Forest
Detection Score

Fig. 3. Detection framework based on global features. We use multiple mapping methods (such as flip, twirl blur, gaussian filter, etc.) to generate variants of the input image. The label change and probability divergence between the original one and variants are regard as a new feature vector to classification.

get s transformed samples: $\Phi_1(x), \Phi_2(x), \ldots, \Phi_s(x)$. Refer to Tian et al. [14] and Li et al. [15], we finally choose 7 space mapping methods as shown in Table 1. For any space mapping method $k \in \{1, \ldots, s\}$, we can calculate label change as:

$$E\left(x, \Phi_k(x)\right)=\left\{\begin{array}{l} 0, \text{ if } f(x)=f\left(\Phi_k(x)\right) \\ 1, \text{ if } f(x) \neq f\left(\Phi_k(x)\right) \end{array}\right. , \tag{1}$$

where $f(x)$ refers to the classification label of the sample x on the neural network f.

We define the softmax prediction probability of f as $P(x)$. It is a $n-$dimension vector while n is the class numbers of f's output. We use Jensen-Shannon divergence (JSD) [16] to denote the probability divergence, in order to avoid the problem of numerical overflow, we add the temperature $T > 0$ when calculating the probability distribution:

$$P(x)=\frac{\exp \left(L_i(x) / T\right)}{\sum_{j=1}^n \exp \left(L_j(x) / T\right)}, \tag{2}$$

where $L_i(x)$ is the logit value of x with label i. Finally, we can get the new feature vector:

$$\begin{aligned} V(x)=(&E(x, \Phi_1(x)), JSD(P(x), P(\Phi_1(x))), \ldots, \\ &E(x, \Phi_s(x)), JSD(P(x), P(\Phi_s(x)))). \end{aligned} \tag{3}$$

Then we use $V(x)$ of both clean examples and adversarial examples to train the Random Forest (RF) in training phase. In inference phase, we can get the prediction probability $F_G(x)$ from RF to determine whether x is adversarial.

2.3. Score Fusion

Fusion score $F(x)$ with weight factor λ is defined as:

$$F(x)=\lambda * F_L(x)+(1-\lambda) * F_G(x). \tag{4}$$

Table 1. Detailed parameter settings of space mapping.

Flip: LR and TB	Twirl Blur Num: 15
Gaussian Filter: 5×5	Saturation Factor: 0.1
Lightness: 0.2	Contrast: 0.6

We regard the larger possibility of whether x is adversarial corresponding to $max(F(x))$ as the final result of discrimination.

3. EXPERIMENTS

3.1. Datasets and training details

We made the comparisons on scaled images with size of 224 × 224 × 3 from ImageNet [17] and Caltech-256 [18]. There are 1000 classes and 256 classes for ImageNet and Caltech-256 respectively. Resnet50 is used as classification model on both datasets. The top-1 accuracy are 76% and 80% respectively. We set 1:1 and 2:1 to train and test the defense model on ImageNet and Caltech-256 respectively. The same amount of adversarial examples and clean examples are in training and inference phase. Dimension $k = 100$ is set to train the SIFT-based BOF model in LF branch. What's more, two branches can be trained at the same time. The final detection accuracy changes with different fusion factor λ, we set $\lambda = 0.6$ to obtain the best performance.

3.2. Attack Methods

We consider adversarial examples with both GP and LP. GP attacks include C&W [13] and Project Gradient Descent (PGD) [19] and LP attacks include Elastic-net Attacks to DNNs (EAD) [20], Decoupled Direction and Norm (DDN) [12] and SparseFool (SF) [21]. Adversarial examples generated through Foolbox and Advertorch. In order to compare with PACA [7], EAD, DDN and SF are set with default parameters. For C&W attacks, we use L_2 norm, confidence $\kappa = 1$ and 500 iterations. We use L_∞ norm on PGD with $\epsilon = 0.03, \alpha = 0.005$ and the maximum iterations is 10.

3.3. Comparison with the State-of-the-art Methods

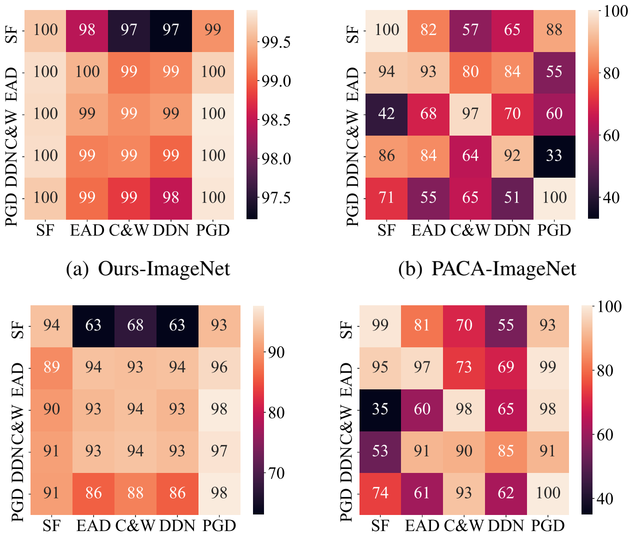
We design experiments compared with FS [3], LID [4], MAD [5], SRM [6] and PACA [7], the results are shown in Table 2 and Table 3. Detection accuracy is defined as the ratio of the numbers of the samples are identified correctly to the total samples including clean and adversarial examples. Our method can achieve average detection accuracy more than 99% for all adversarial examples and have the state-of-the-art performance on the ImageNet dataset. For Caltech-256, although our method only has a leading property in detecting EAD adversarial examples, it keeps the gap within 5.5% compared with current state-of-the-art methods.

Table 2. Detection accuracy(%) on ImageNet.

	FS	LID	MDA	SRM	PACA	Ours
SF	61.30	59.67	66.38	53.39	98.30	**99.64**
EAD	54.75	60.88	68.70	74.82	89.22	**99.52**
C&W	55.47	64.25	68.87	87.24	96.05	**99.36**
DDN	69.63	61.41	68.52	65.62	91.47	**99.04**
PGD	95.55	99.21	99.55	99.68	99.32	**99.86**

Table 3. Detection accuracy(%) on Caltech-256.

	FS	LID	MDA	SRM	PACA	Ours
SF	59.97	73.94	80.61	82.82	**97.19**	93.72
EAD	52.07	71.42	74.08	83.48	90.71	**93.77**
C&W	58.25	70.29	73.24	97.59	**97.97**	92.62
DDN	61.17	69.49	74.19	78.10	**97.29**	93.39
PGD	**100**	97.13	99.97	99.78	99.32	97.58



(a) Ours-ImageNet

(b) PACA-ImageNet

(c) Ours-Caltech256

(d) PACA-Caltech256

Fig. 4. The detection accuracy (%) of a cross combination of 5 attack methods between our method and PACA on two datasets. Each row of each subgraph indicates adversarial example generation methods in training phase and each column indicates adversarial example generation methods in inference phase.

3.4. Generalization Experiments

Generalization property is an important indicator to evaluate the ability of a detector to resist unknown attacks. PACA has shown better generalizability than other methods in most cases [7]. Fig.4 shows the generalization heatmap of our method and PACA on ImageNet and Caltech-256 respectively. Fig.4 (a) and Fig.4 (b) show the generalization property on ImageNet. Our method achieves average detection accuracy of 99.21% regardless of adversarial attack method of training and testing, which has obvious advantages compared with PACA (73.4%). Besides, the average accuracy numerical variance of our method and PACA are 0.46 and 19.31. The result indicates our method has achieved the state-of-the-art performance in terms of generalization property on ImageNet. As shown in Fig.4 (c) and Fig.4 (d), our method still has the advantage on Caltech-256. When training adversarial examples generated by EAD, C&W, DDN and PGD, our average detection accuracy (92.53%) is higher than PACA (79.45%) and the accuracy variance are 2.66 and 18.89, which means that our method has better generalization property . As for SF examples training, the average detection accuracy of test adversarial examples is slight lower (3.44%) but the variance value (14.21) is lower than the performance corresponding to PACA (17.71).

3.5. Ablation Study

There are three key parts in our model, which are ELA module, BOF module and space mapping (SM) module. In order to verify the effectiveness of each module, we successively remove each functional module and conduct ablation study by observing the detection accuracy. We carry out the experiment by testing C&W adversarial examples with different confidence values. As shown in Table 4, we find that with the strengthening of perturbation, the role of SM module gradually weakens. We find that make the ELA of the compressed image can effectively improve detection accuracy. The model performs best using all modules.

Table 4. Ablation study on ImageNet dataset.

Remove	CW0	CW10	CW20	CW30	CW40
ELA+SM	73.60	73.70	73.54	73.78	72.84
ELA+BOF	84.46	84.45	84.20	83.22	83.82
ELA	88.16	87.43	87.42	87.32	86.86
SM	98.84	99.06	99.42	99.48	99.77
None	**99.36**	**99.46**	**99.66**	**99.68**	**99.86**

4. CONCLUSION

This paper improves the detection accuracy and generalization performance of global perturbation and novel local perturbation adversarial examples. Both types of adversarial examples are invisible to human eyes. We propose a two-branch architecture to extract and analyze the local and global image features. Experiments indicate our method has better detection performance and generalization property in most cases.

5. REFERENCES

[1] Ian J. Goodfellow, Yoshua Bengio, and Aaron C. Courville, "Deep learning," 2016.

[2] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy, "Explaining and harnessing adversarial examples," in *3rd International Conference on Learning Representations (ICLR)*, 2015.

[3] Weilin Xu, David Evans, and Yanjun Qi, "Feature squeezing: Detecting adversarial examples in deep neural networks," in *25th Annual Network and Distributed System Security Symposium (NDSS)*, 2018.

[4] Xingjun Ma, Bo Li, Yisen Wang, Sarah M. Erfani, Sudanthi N. R. Wijewickrema, Grant Schoenebeck, Dawn Song, Michael E. Houle, and James Bailey, "Characterizing adversarial subspaces using local intrinsic dimensionality," in *6th International Conference on Learning Representations (ICLR)*, 2018.

[5] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin, "A simple unified framework for detecting out-of-distribution samples and adversarial attacks," in *Neural Information Processing Systems (NeurIPS)*, 2018.

[6] Jiayang Liu, Weiming Zhang, Yiwei Zhang, Dongdong Hou, Yujia Liu, Hongyue Zha, and Nenghai Yu, "Detection based defense against adversarial examples from the steganalysis point of view," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[7] Kejiang Chen, Yuefeng Chen, Hang Zhou, Chuan Qin, Xiaofeng Mao, Weiming Zhang, and Nenghai Yu, "Adversarial examples detection beyond image space," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2021, IEEE.

[8] Zihao Liu, Qi Liu, Tao Liu, Nuo Xu, Xue Lin, Yanzhi Wang, and Wujie Wen, "Feature distillation: Dnn-oriented JPEG compression against adversarial examples," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[9] Nilaksh Das, Madhuri Shanbhogue, Shang-Tse Chen, Fred Hohman, Li Chen, Michael E. Kounavis, and Duen Horng Chau, "Keeping the bad guys out: Protecting and vaccinating deep learning with JPEG compression," *arXiv preprint arXiv:1705.02900*, 2017.

[10] Gintare Karolina Dziugaite, Zoubin Ghahramani, and Daniel M. Roy, "A study of the effect of JPG compression on adversarial images," *arXiv preprint arXiv:1608.00853*, 2016.

[11] Wei Wang, Jing Dong, and Tieniu Tan, "Tampered region localization of digital color images based on JPEG compression noise," in *International Workshop on Digital Watermarking (IWDW)*. Springer, 2010, pp. 120–133.

[12] Jérôme Rony, Luiz G. Hafemann, Luiz S. Oliveira, Ismail Ben Ayed, Robert Sabourin, and Eric Granger, "Decoupling direction and norm for efficient gradient-based L2 adversarial attacks and defenses," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[13] Nicholas Carlini and David A. Wagner, "Towards evaluating the robustness of neural networks," in *2017 IEEE Symposium on Security and Privacy (SP)*. 2017, pp. 39–57, IEEE.

[14] Shixin Tian, Guolei Yang, and Ying Cai, "Detecting adversarial examples through image transformation," in *Association for the Advancement of Artificial Intelligence (AAAI)*, 2018.

[15] Xurong Li, Shouling Ji, Juntao Ji, Zhenyu Ren, Chunming Wu, Bo Li, and Ting Wang, "Adversarial examples detection through the sensitivity in space mappings," 2020, vol. 14, pp. 201–213.

[16] Christopher Manning and Hinrich Schutze, "Foundations of statistical natural language processing," 1999, MIT press.

[17] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[18] Gregory Griffin, Alex Holub, and Pietro Perona, "Caltech-256 object category dataset," 2007, California Institute of Technology.

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

[20] Pin-Yu Chen, Yash Sharma, Huan Zhang, Jinfeng Yi, and Cho-Jui Hsieh, "EAD: elastic-net attacks to deep neural networks via adversarial examples," in *Association for the Advancement of Artificial Intelligence (AAAI)*, 2018.

[21] Apostolos Modas, Seyed-Mohsen Moosavi-Dezfooli, and Pascal Frossard, "Sparsefool: A few pixels make a big difference," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.