Robust Self-Supervised Learning for Adversarial Attack Detection

Anonymous Author(s)
Affiliation
email

Abstract

In this paper, we propose a self-supervised representation learning framework for the adversarial attack detection task to address this drawback. Firstly, we map the pixels of augmented input images into an embedding space. Then, we employ the prototype-wise contrastive estimation loss to cluster prototypes as latent variables. Additionally, drawing inspiration from the concept of memory banks, we introduce a discrimination bank to distinguish and learn representations for each individual instance that shares the same or a similar prototype, establishing a connection between instances and their associated prototypes. We propose a parallel axial-attention (PAA)-based encoder to facilitate the training process by parallel training over height- and width-axis of attention maps. Experimental results show that, compared to various benchmark self-supervised vision learning models and supervised adversarial attack detection methods, the proposed model achieves state-of-the-art performance on the adversarial attack detection task across a wide range of images.

1 Introduction

Given an image potentially perturbed by an attack algorithm, the goal of adversarial attack detection is to distinguish between adversarial and normal samples using the differences between them. Adversarial attack detection is an important security topic applicable in real-world applications such as autonomous driving systems, object detection, medical image processing, and robotics (1; 2; 3; 4) among many others. Recent deep learning-based adversarial attack detection techniques (5; 6; 7) are predominantly trained in a supervised manner, where a large number of labeled adversarial and normal samples are provided as input to neural networks. The model is then trained to reconstruct the corresponding clean sample and compare it with the input sample to provide the detection result. Consequently, supervised learning-based adversarial attack detection approaches suffer from three main drawbacks.

Firstly, human-imperceptible adversarial attacks on images are challenging to label manually. This process can be time-consuming and may introduce errors, particularly when the annotator lacks familiarity with the task. Secondly, the trained adversarial attack detection models may need to be deployed in previously unseen conditions, including novel attack algorithms and datasets. Consequently, there is a strong likelihood of a mismatch between the training and testing conditions. In such cases, we lack the ability to leverage recorded test data to improve the model's performance in the unseen test setting. Thirdly, prototype-based adversarial attack detection methods (8; 5) estimate an object's category (e.g., cats or dogs) as the prototype. These methods calculate the degree of similarity between new data samples and autonomously chosen prototypes to classify images as adversarial or normal samples. However, each prototype may potentially consists of multiple instance samples, which often leads to a neglect of the rich intrinsic semantic relationships between prototypes of individual objects in images. For example, while the model may be trained on some tank images,

Submitted to 38th Conference on Neural Information Processing Systems (NeurIPS 2024). Do not distribute.

it may struggle to classify new tanks or entirely new classes of objects when faced with previously unseen types of tanks.

To overcome these drawbacks, we propose a self-supervised representation learning framework aimed at extracting feature representations for the downstream task, i.e., adversarial attack detection. Building upon pixel mapping and contrastive estimation, we propose a discrimination bank to distinguish individual instances for each prototype from the embedding space. We demonstrate that the instance-wise feature maps capture richer information compared to the prototype-based approach, resulting in performance improvements.

2 Proposed Method

Our proposed framework is presented in Figure 1.

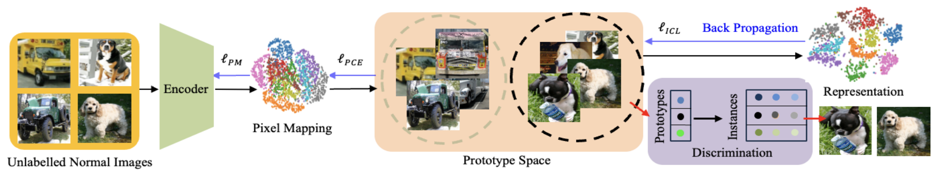

Figure 1: Self-supervised representation learning framework.

2.1 Pixel Mapping

As the first major component of the encoder, a PAA-based network with parameter θ is exploited to transform training set $X = \{x_1, x_2, ..., x_n\}$ of n image samples to feature vectors $V = \{v_1, v_2, ..., v_I\}$, such that V best describes X. Different from previous work, we propose a pixel mapping loss with data augmentation, $\mathcal{L}_{\text{PM}}$, to learn an invariant representation of x_i by minimizing the risk $\sum_i \mathcal{L}\left(x_i, v_i; \theta\right)$. To achieve that, we use a pair of transformations, denoted as t and s, in some set of transformations $\mathcal{T}$ (e.g. geometric transformations) to x_i, to produce the augmentation as $x_i^{t_i}$ and $x_i^{s_i}$. We define this process as $V = f_{PM}(X)$ with the loss as:

$$\mathcal{L}_{\text{PM}} = -\log \frac{\exp\left(f_{PM}\left(x_i^{t_i}\right)^T \cdot f_{PM}\left(x_i^{s_i}\right)/\tau\right)}{\sum_{b=1}^{B} \exp\left(f_{PM}\left(x_b^{t_b}\right)^T \cdot f_{PM}\left(x_i^{s_i}\right)/\tau\right)} \tag{1}$$

where T and B are the transpose symbol and batch size, respectively. It is highlighted that all the embeddings in the loss function are L2-normalized (9). While previous data augmentation studies (10) have shown that the choice of transformation techniques plays an important part in self-supervised representation learning, most previous works do not give much consideration to the individual choice of t_i and s_i on pairs of images, which are simply uniformly sampled over $\mathcal{T}$. Therefore, in the proposed pixel mapping technique, we aim to overcome this limitation and select the optimal transformation algorithm for each sample x_i. To achieve this, we select transformation algorithms that maximize the risk defined by the loss $\mathcal{L}^{\text{PM}}$:

$$\{t_i, s_i\} = \underset{\{t_i, s_i\} \in \mathcal{T}}{\arg\max} \sum_{i=1}^{n} \mathcal{L}_{\text{PM}}\left(x_i^{t_i}, x_i^{s_i}; \theta, \mathcal{T}\right) \tag{2}$$

In the proposed pixel mapping technique, we prioritize the difference between t_i and s_i for each image over their absolute values.

2.2 Prototype-wise Contrastive Estimation

We assume that the observed data x_i are related to latent variable $P = \{p_i\}$ which denotes the prototypes of the data. We aim to find a network parameter that maximizes the log-likelihood function of the observed n samples by a prototype-wise contrastive estimation (PCE). To achieve that, we use the local peaks of the density (11) as the prototype, in other words, the most representative data samples of X. The loss, namely $\mathcal{L}_{\text{PCE}}$, is defined as:

$$\mathcal{L}_{\text{PCE}} = \frac{1}{|\mathcal{M}|} \sum_{p_i^+ \in \mathcal{M}} - \log \frac{\exp\left(v_i \cdot p_i^+ / \gamma\right)}{\sum_{p_i^- \in \mathcal{N}} \exp\left(v_i \cdot p_i^- / \gamma\right)} \tag{3}$$

where $\mathcal{M}_i$ and $\mathcal{N}_i$ are prototype collections of the positive and negative samples, respectively. As aforementioned, inspired from previous supervised learning work (12)(13), we find different levels of concentration distributes around each prototype embeddings. Therefore, we exploit γ as the concentration level around the prototype p^m within the m-th cluster as:

$$\gamma = \frac{\sum_{i=1}^{n} \|p^m - v_i^m\|_2}{n \log(n + \beta)} \tag{4}$$

where the momentum features are $\{v_i^m\}_{i=1}^n$ within the same cluster as a prototype p. We set a smooth parameter β to ensure that small clusters do not have an overly-large γ. Then, γ acts as a scaling factor on the similarity between an embedding v and its prototype p.

2.3 Instance-Wise Contrastive Learning

The core of our method lies in establishing a connection between prototype and instance features to facilitate instance clustering. Initially, we create K independent discrimination banks to enhance instance discrimination across clusters. Similar to a memory bank, the discrimination bank aids in contrastive learning, leveraging extensive data to acquire robust representations. We assume a contrastive set J_i for the t-th bank A_t as:

$$J_i = \{z_i' \mid z_i' \in A_t \forall t \in [1, C]\} \tag{5}$$

where z_i' is the estimated representation of x_i. Specifically, for each training batch with B samples and M prototypes, our discrimination memory is built with size $M \times B \times D$, where D is the dimension of pixel embeddings. The (p^m, b)-th element in the discrimination memory is a D-dimensional feature vector obtained by average pooling all the embeddings of pixels labeled as p^m prototype in the b-th batch. To update the discrimination bank, we enqueue each instance to the nearest prototype and add the new one in each back propogation cycle:

$$\mathcal{L}_{\text{ICL}} = \frac{\exp(\cos(v_i, z_i) \cdot \cos\left(v_i, p_i^m / \phi\right))}{\sum_{z' \in A_t} \sum_{j=0}^{r} \exp(\cos(v_i, z_j') \cdot \cos\left(v_i, p_j^m / \phi\right)) \cdot J_i} \tag{6}$$

where $\cos(\cdot, \cdot)$ is the cosine similarity between a pair of representations. The concentration level of $\mathcal{L}_{\text{ICL}}$ is presented as ϕ and estimated similar as γ in (4) but we replace v_c' to z_c'. With the loss, we discriminate representations belongs to the same bank. To discover the underlying concepts with unique visual characteristics, we infer their decision boundaries by reducing the visual redundancy among clusters, namely maximising the visual similarity of samples within the same clusters and minimising that between clusters. The overall cost-function used to train the MAE is now a combination of the above loss terms with hyper-parameters λ_1 and λ_2 as $\mathcal{L} = \mathcal{L}_{\text{PM}} + \lambda_1 \cdot \mathcal{L}_{\text{PCE}} + \lambda_2 \cdot \mathcal{L}_{\text{ICL}}$.

3 Experiments

3.1 Datasets and Attacks

We randomly select 50,000 images from ImageNet (14) and 10,000 images from ImageNet Large Scale Visual Recognition Challenge (ILSVRC) (15) for the training and validation, respectively. As aforementioned, we evaluate the competitor and proposed models with unseen datasets. In the test stage, we extensively perform experiments on 10,000 random images from each CIFAR-10 (16) and COCO (17).

We select seven attack algorithms (18)(19)(20)(21)(22)(23)(24) in the test stage because they are robust to novel adversarial attack detection and defense techniques.

3.2 Implementation Details

In the experiment, we implement the network with a ResNet-50 (25) whose last fully-connected layer outputs a 128-D and L2-normalized feature with a parallel axial-attention (PAA) block (26). We multiply all the channels by 1.5 and 2, resulting in PAA-ResNet-M, L, respectively. We always use 8 heads in multi-head attention blocks (27). In order to avoid careful initialization of weights (W_Q, W_K, W_V) and location vectors (r^q, r^k, r^v), we use batch normalizations (28) in all attention layers. To evaluate and compare the adversarial attack detection accuracy, we use the detection rate (DR).

The proposed model is trained by using the SGD optimizer with a weight decay of 0.0001, a momentum of 0.9, and a batch size of 256. We train the networks for 200 epochs, where we warm-up the network in the first 20 epochs by only using the pixel-mapping loss. The initial learning rate is 0.03, and is multiplied by 0.1 at 120 and 160 epochs. In terms of the hyper-parameters, we set $\tau = 0.1$, $\beta = 10$, $r = 16000$, $\lambda_1 = 1$ and $\lambda_2 = 1$ based on grid search.

3.3 Results

We assess the learned representation over CIFAR-10 and COCO. Tables 1 $\&$ 2 show the results.

Table 1: Comparison on CIFAR-10.

Models	Clean (%)	Attacked (%)
TiCo (29)	81.4	78.0
MAE (30)	89.9	74.2
Mugs (31)	90.5	73.7
Unicom (32)	92.6	84.1
DINOV2 (33)	94.3	86.7
ESMAF (34)	73.8	56.4
TS (6)	89.7	59.5
sim-DNN (13)	82.0	65.7
DTBA (35)	87.0	74.1
TLC (36)	84.9	72.4
SimCat (37)	88.0	77.3
PAA-ResNet-S	92.7	84.4
PAA-ResNet-M	94.1	87.8
PAA-ResNet-L	**94.8**	**89.0**

Table 2: Comparison on COCO.

Models	Clean (%)	Attacked (%)
TiCo (29)	78.9	67.3
MAE (30)	88.9	73.5
Mugs (31)	89.0	73.3
Unicom (32)	90.2	82.8
DINOV2 (33)	**91.7**	83.9
ESMAF (34)	75.4	55.6
TS (6)	76.7	56.8
sim-DNN (13)	80.6	62.2
DTBA (35)	85.3	68.8
TLC (36)	80.8	71.5
SimCat (37)	82.6	70.1
PAA-ResNet-S	90.9	83.7
PAA-ResNet-M	91.5	84.9
PAA-ResNet-L	**91.7**	**85.6**

On both datasets, our models show strong detection performance: accuracy improves considerably with the proposed algorithm. Additionally, our results outperforms both the self-supervised and supervised results by large margins on clean images detection.

Furthermore, we perform experiments to evaluate the robustness of our work. Table 3 shows the detection accuracy results (in %) with CIFAR-100 (16) and ImageNet-R (38).

Table 3: Adversarial attack detection performance (clean / attacked images) on seen and unseen datasets.

Training	ImageNet-R		ILSVRC		CIFAR-100	
Test	ImageNet-R	CIFAR-10	ILSVRC	CIFAR-100	CIFAR-100	ImageNet-R
Unicom (32)	91.9 / 82.7	91.0 / 80.4	94.7 / 88.5	92.0 / 81.1	93.3 / 82.7	89.3 / 77.9
DINOV2 (33)	93.4 / 84.5	92.4 / 81.7	96.2 / 90.0	93.4 / 82.6	95.1 / 84.0	90.5 / 79.4
DTBA (35)	92.2 / 85.2	85.3 / 76.9	96.0 / 90.3	86.8 / 78.2	94.7 / 83.1	88.2 / 69.9
PAA-ResNet-L	**93.5 / 87.9**	**92.9 / 85.7**	**97.1 / 90.5**	**94.2 / 87.0**	**96.0 / 87.6**	**92.1 / 83.4**

Compared to supervised learning-based methods (34)(6)(35)(13), the proposed SSL representation learning method experiences relatively less performance degradation.

4 Conclusion

In this paper, we have proposed a self-supervised representation learning approach for adversarial attack detection, offering an effective alternative to traditional supervised pipelines. We establish a connection between prototype and instance features through the use of a discrimination bank, thereby enriching the information available to enhance the proposed model's ability to detect adversarial attacks. Our evaluation with different datasets and attacks has demonstrated the robust performance of the proposed method on unseen datasets.

References

[1] J. Liu, A. Levine, C. P. Lau, R. Chellappa, and S. Feizi, "Segment and complete: defending object detectors against adversarial patch attacks with robust patch detection," *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[2] V. Raina and M. Gales, "Residue-based natural language adversarial attack detection," *Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2022.

[3] X. Wang, S. Li, M. Liu, Y. Wang, and A. Roy-Chowdhury, "Multi-expert adversarial attack detection in person re-identification using context inconsistency," *Proceedings of IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[4] Y. Yang, S. Yang, J. Xie, Z. Si, K. Guo, K. Zhang, and K. Liang, "Multi-head uncertainty inference for adversarial attack detection," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023.

[5] A. L. Pellcier, Y. Li, and P. Angelov, "PUDD: Towards Robust Multi-modal Prototype-based Deepfake Detection," *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[6] S. Kiani, S. Awan, C. Lan, and B. L. F. Li, "Two souls in an adversarial image: towards universal adversarial example detection using multi-view inconsistency," *Asia-Pacific Computer Systems Architecture Conference (APCSAC)*, 2021.

[7] Y. Li, P. Angelov, and N. Suri, "Domain generalization and feature fusion for cross-domain imperceptible adversarial attack detection," *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, 2023.

[8] A. L. Pellcier, K. Giatgong, Y. Li, N. Suri, and P. Angelov, "UNICAD: A unified approach for attack detection, noise reduction and novel class identification," *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, 2024.

[9] M. Yang, Z. Meng, and I. King, "FeatureNorm: L2 feature normalization for dynamic graph embedding," *Proceedings of IEEE International Conference on Data Mining (ICDM)*, 2020.

[10] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," *Proceedings of International Conference on Machine Learning (ICML)*, 2020.

[11] P. Angelov and E. Soares, "Towards explainable deep neural networks (xDNN)," *Neural Networks*, vol. 130, p. 185–194, 2020.

[12] Y. Gong, S. Wang, X. Jiang, L. Yin, and F. Sun, "Adversarial example detection using semantic graph matching," *Applied Soft Computing*, vol. 141, p. 110317, 2023.

[13] E. Soares, P. Angelov, and N. Suri, "Similarity-based deep neural network to detect imperceptible adversarial attacks," *Proceedings of IEEE Symposium Series on Computational Intelligence (SSCI)*, 2022.

[14] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: a large-scale hierarchical image database," *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[15] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and F. Li, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision (IJCV)*, 2015.

[16] A. Krizhevsky, "Learning multiple layers of features from tiny images," *Master's thesis*, 2009.

[17] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, "Microsoft COCO: common objects in context," *Proceedings of European Conference on Computer Vision (ECCV)*, 2014.

[18] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," *Proceedings of International Conference on Machine Learning (ICML)*, 2017.

[19] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *Proceedings of International Conference on Learning Representations (ICLR)*, 2015.

[20] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: a simple and accurate method to fool deep neural networks," *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

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

[22] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," *IEEE Symposium on Security and Privacy*, 2017.

[23] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, "The Limitations of Deep Learning in Adversarial Settings," *IEEE Symposium on Security and Privacy*, 2016.

[24] C. Luo, Q. Lin, W. Xie, B. Wu, J. Xie, and L. Shen, "Frequency-driven imperceptible adversarial attack on semantic similarity," *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[26] Y. Li, P. Angelov, and N. Suri, "Self-supervised representation learning for adversarial attack detection," *Proceedings of European Conference on Computer Vision (ECCV)*, 2024.

[27] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *International Conference on Neural Information Processing Systems (NeurIPS)*, 2017.

[28] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," *Proceedings of International Conference on Machine Learning (ICML)*, 2015.

[29] J. Zhu, R. Moraes, S. Karakulak, V. Sobol, A. Canziani, and Y. LeCun, "Tico: transformation invariance and covariance contrast for self-supervised visual representation learning," *arXiv preprint arXiv: 2203.14415*, 2022.

[30] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[31] P. Zhou, Y. Zhou, C. Si, W. Yu, T. K. Ng, and S. Yan, "Mugs: a multi-granular self-supervised learning framework," *arXiv preprint arXiv: 2203.14415*, 2022.

[32] X. An, J. Deng, K. Yang, J. Li, Z. Feng, J. Guo, J. Yang, and T. Liu, "Unicom: universal and compact representation learning for image retrieval," *Proceedings of International Conference on Learning Representations (ICLR)*, 2023.

[33] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. Huang, S.-W. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski, "Dinov2: learning robust visual features without supervision," *arXiv preprint arXiv: 2304.07193*, 2023.

[34] J. Chen, T. Yu, C. Wu, H. Zheng, W. Zhao, L. Pang, and H. Li, "Adversarial attack detection based on example semantics and model activation features," *Proceedings of International Conference on Data Science and Information Technology (DSIT)*, 2022.

[35] P. Qi, T. Jiang, L. Wang, X. Yuan, and Z. Li, "Detection tolerant black-box adversarial attack against automatic modulation classification with deep learning," *IEEE Transactions on Reliability*, vol. 71, no. 2, pp. 674–686, 2022.

[36] C. C. Chyou, H.-T. Su, and W. H. Hsu, "Unsupervised adversarial detection without extra model: training loss should change," *Proceedings of International Conference on Machine Learning (ICML)*, 2023.

[37] M. Moayeri and S. Feizi, "Sample efficient detection and classification of adversarial attacks via self-supervised embeddings," *Proceedings of IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[38] D. Hendrycks, S. Basart, N. Mu, S. Kadavath, F. Wang, E. Dorundo, R. Desai, T. Zhu, S. Parajuli, M. Guo, D. Song, J. Steinhardt, and J. Gilmer, "The many faces of robustness: a critical analysis of out-of-distribution generalization," *Proceedings of IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.