EXPLOITING ONE-CLASS CLASSIFICATION OPTIMIZATION OBJECTIVES FOR INCREASING ADVERSARIAL ROBUSTNESS

Vasileios Mygdalis and Ioannis Pitas

Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki, 54124, Greece

ABSTRACT

This work examines the problem of increasing the robustness of deep neural network-based image classification systems to adversarial attacks, without changing the neural architecture or employ adversarial examples in the learning process. We attribute their famous lack of robustness to the geometric properties of the deep neural network embedding space, derived from standard optimization options, which allow minor changes in the intermediate activation values to trigger dramatic changes to the decision values in the final layer. To counteract this effect, we explore optimization criteria that supervise the distribution of the intermediate embedding spaces, in a class-specific basis, by introducing and leveraging one-class classification objectives. The proposed learning procedure compares favorably to recently proposed training schemes for adversarial robustness in black-box adversarial attack settings.

***Index Terms*—** Adversarial Robustness, One-class Classification, Adversarial Attacks, Adversarial Defense

1. INTRODUCTION

One of the most important drawbacks of the application of deep neural networks in sensitive image/video classification tasks is their limited robustness to adversarial attacks i.e., they are susceptible of being fooled by carefully crafted minor/humanly imperceptible perturbations. Adversarial attacks are methods that calculate such perturbations by exploiting the neural network backward pass to obtain gradient flow from the activations of the final (or even some intermediate) layer towards the input, using some loss function. When both the model architecture and parameters are known to the adversary, adversarial attacks are classified as white-box, while black-box/transferability attacks are devised from different host models or from the same architecture with different parameters. Up-to-date, there is a wealth of literature describing different forms of adversarial attacks that can be found in review papers [1, 2], where the reader is referred to.

Adversarial defenses are methods designed to counter adversarial attacks. The most prominent defenses so far are based on *adversarial training* [3, 4], which in simplified terms, involves training a deep neural network with adversarial examples of predefined noise margins, calculated implicitly or explicitly. Such approaches have two important disadvantages. First, they require a significantly added workflow during the model training process for generating and training with adversarial attacks, second, the resulting models seem to have decreased classification accuracy in clean data. On the contrary, another line of work [5, 6] achieves robustness by manipulating the properties of the learned feature space, by exploiting distance-based optimization criteria in the form of intermediate supervision functions. As a result, the learned representation has decreased within-class dispersion and increased between-class separation in the intermediate feature spaces, while such approaches can be used in conjunction with adversarial training, for added robustness benefits.

This work builds on the latter direction and extends the recently proposed Hyperspherical Class Prototypes (HCP) method [6], by incorporating novel optimization terms inspired by the present state-of-the-art in deep neural network-based one-class classification problems [7, 8, 9]. The proposed method does not imply modifications to the deep neural architectures or the creation of adversarial examples for training purposes. It is deployed in the form of alternative loss functions that supervise the distribution of final and intermediate layer activation values. It is shown that the proposed method increases (or at least does not hinder) the classification accuracy in clean examples, while it provides increased robustness to adversarial attacks at the same time. The proposed method is evaluated in black-box/transferability-based adversarial attack settings in image classification tasks, as this scenario excludes any potential robustness induced by gradient obfuscation [10].

The rest of the paper is structured as follows. Section 2 overviews existing adversarila defenses. Section 3 analytically describes the components of the proposed method. Section 4 describes the experiments conducted in order to evaluate the effectiveness of the proposed method in image classification problems in publicly available datasets. Finally, conclusions are drawn in Section 5.

This work has received funding from the European Union's European Union Horizon 2020 research and innovation programme under grant agreement 951911 (AI4Media). This publication reflects only the authors' views. The European Commission is not responsible for any use that may be made of the information it contains.

2. ADVERSARIAL DEFENSES

Adversarial defenses in classification systems aim to increase their ability to withstand or overcome input perturbation, generated by adversarial attacks. Assuming a classification system $y = f(\mathbf{x}; \boldsymbol{\theta})$, where f is the model decision function parametrized by $\boldsymbol{\theta}$, $\mathbf{x}$ are the model inputs and y is the model prediction, robustness is quantified by determining its tolerance to perturbation $\|\mathbf{p}\| < \epsilon$ per se, i.e., $f(\mathbf{x}; \boldsymbol{\theta}) = f(\mathbf{x} + \mathbf{p}; \boldsymbol{\theta})$. Here it should be noted that other definitions of adversarial robustness has been proposed in the past, that focus on altering the classification architecture, e.g., input filtering [11], Generative methods [12]. Using the above definition of robustness, we consider such methods irrelevant to the proposed one. Up to date, the perturbation levels required to fool neural network classifiers with adversarial attacks are very low, i.e., perturbed images are almost indistinguishable from the original ones to the human eye.

Our work focuses on adversarial defenses that modify the training process of neural network, while maintaining the same neural network architecture, only by trying to derive in different parameters i.e., $f(\mathbf{x}; \tilde{\boldsymbol{\theta}})$. The straightforward approach to this end is to fine-tune or re-train the model by exploiting adversarial samples, derived by employing one or more adversarial attack methods [3, 4]. This process can be applied during training by employing an additional objective function inspired by adversarial attacks. For instance, the Fast Gradient Sign [3] objectives have been employed for adversarial training in the following manner:

$$\mathcal{L}_{AT} = \lambda \mathcal{L}_{CE}(f(\mathbf{x}; \boldsymbol{\theta}), y) + (1 - \lambda)\mathcal{L}_{CE}(f(\tilde{\mathbf{x}}; \boldsymbol{\theta}), y), \quad (1)$$

where $\tilde{\mathbf{x}}$ is an adversarial sample derived from $\mathbf{x}$ using the Fast Gradient Sign method, $\mathcal{L}_{CE}$ is the standard cross entropy loss function and $0 \leq \lambda \leq 1$ is hyperparameter that controls the learning balance between clean and adversarial samples (a value equal to 0.1 has been proposed showing good results [3]). A more sophisticated variant [4] generalized the adversarial training approach by incorporating combinations of general adversarial attacks and remains up to date, the most efficient defense mechanism. The problem of adversarial robustness can also be treated from a domain adaptation point of view [13]. That is, intermediate layer clean and adversarial data representations are projected to a subspace by employing a Graph Neural Network [14], and the divergence between them is minimized by computing an approximation of the Wasserstein distance [15]. The main disadvantages of these approaches are the introduced workflow for calculating the adversarial examples, while at the same time, model classification accuracy in clean data is negatively affected. Moreover, due to the adversarial attack-specific nature, there is no guarantee [16] that such defenses remain effective against different types of adversarial defense.

Ultimately, the effectiveness of adversarial defense methods that fall into the above category seem to rely on achieving the production of as similar intermediate data representations as possible for both clean and adversarial images belonging to the same specific class. Recently proposed adversarial defenses [5, 6] showed that incorporating distance-based optimization criteria might achieve this goal, without requiring re-training the model with adversarial examples. The second advantage of such methods is that they might employ adversarial training as a complementary step, providing increased robustness to specific adversarial attacks. Inspired by the Nearest Centroid Classifier [17] and combining ideas related to the triplet-loss [18] and center-loss [19] functions, the classification model is encouraged to produce class data representations that lie close to some learned class prototype vectors, leading to increased robustness in adversarial attacks, by only having minor degradation in classification accuracy for clean samples. More specifically, recently proposed adversarial defenses achieve this goal by learning class prototype vectors in the intermediate hidden layer spaces, and minimize the distances between the class data representations and the prototype vectors. For instance, assuming $\mathbf{g}_k(\mathbf{x}; \boldsymbol{\theta})$ to be the k-th layer representation of some input $\mathbf{x}$, and $\mathbf{a}_j$ the j-th class prototype vector, the Center Loss [19] criteria are optimized as follows:

$$\mathcal{L}_{CL} = \|\mathbf{g}_k(\mathbf{x}; \boldsymbol{\theta}) - \mathbf{a}_j^{(k)}\|^2, \quad (2)$$

leading to more compact data representations for elements belonging to the j-th class. Here, it should be noted that this specific function has some drawbacks related to the representation collapse problem as pointed out in recent work [8, 9]; that is, the loss might lead to trivial solutions after some optimization steps. To counteract such effects, modified versions of it have been proposed in one-class classification settings, e.g., early stopping criteria [7, 8], as well as in adversarial robustness methods, including regularizers and contrastive loss formulations [5, 6].

3. ROBUST ONE-CLASS CLASSIFICATION-BASED TRAINING LOSS

The relevance of one-class classification methods to adversarial robustness stems from the fact that adversarial samples may be considered as outliers to the standard training data distribution. Moreover, in contrast to multi-class classifiers, one-class classifiers are not obliged to output a specific class for each of their input; if the input data fall outside all one-class model distributions, they are considered as outliers, by definition. These facts have been demonstrated in [7, 20] where one-class classifiers had been employed as adversarial sample detectors. This work does not employ one-class classification as adversarial sample detectors, but only as a vehicle to construct the robust feature learning process.

The first objective of the proposed learning process is to derive tight class boundaries in the deep representation space. We adopt the HCP optimization problem [6] to this end. That is, the optimal tight class boundaries are determined by enclosing feature space class data representations with hyperspheres, and thereby minimize the respective hypersphere volumes. This method alters the training procedure of a standard neural network architecture, by training in-parallel, an additional layer that includes the prototype vector centers in the feature space. The one-class classification criteria have been formally extended to the multi-class classification case. Let $\mathcal{K}$ be the set of layers on which the proposed objectives will be applied to, where $\mathbf{g}_k(\mathbf{x};\boldsymbol{\theta})$ is k-th layer representation of some input $\mathbf{x}$. This method aims to learn hypershserical prototypes in the k-th layer defined by the prototype matrices $\mathbf{A}^{(k)} \in \mathbb{R}^{C\times L_k}$, where L_k is the dimensionality of the k-th layer, and radii $\mathbf{R}^{|\mathcal{K}|\times C}$ that will act as one-class classifiers, verifying data sample activations belonging to the j-th class. To this end, the optimization problem for each sample $\mathbf{x}_i$ is the following:

$$\begin{aligned}\min_{\mathbf{R},\boldsymbol{\Xi},\boldsymbol{A}^{(k)}}: \quad & \sum_{k\in\mathcal{K}}\sum_{j=1}^{C} r_{kj}^2 + \sum_{k\in\mathcal{K}} c_k \sum_{i=1}^{N} \xi_{ki} \qquad (3)\\ \text{s.t.:} \quad & \sum_{k\in\mathcal{K}}\sum_{j=1}^{C}\left(-y_{ij}\left(r_{kj}^2 - \|\mathbf{g}_k(\mathbf{x}_i;\boldsymbol{\theta}) - \mathbf{a}_j^{(k)}\|^2\right) \le \xi_{ki}\right),\\ & \xi_{ki} \ge 0\end{aligned}$$

where $\mathbf{a}_j^{(k)}$ is the prototype center for class j, $y_{ij} = 1$ if sample $\mathbf{x}_i$ belongs to class j, or $y_{ij} = -1$, otherwise, ξ_{ki} are the slack variables and $c_k \ge 0$ is a hyperparameter that allows training error (i.e., soft margin formulation) relaxing the optimization constraints. The constraints of the above optimization problem can be optimized by applying the following hinge loss function in every layer selected in $\mathcal{K}$:

$$\mathcal{L}_M = \sum_{j}^{C} \max\left(c_k, -y_{ij}\left(r_{kj}^2 - \|\mathbf{g}_k(\mathbf{x}_i;\boldsymbol{\theta}) - \mathbf{a}_j^{(k)}\|^2\right)\right). \qquad (4)$$

In the deep learning case, both the feature vectors and the prototype vectors are trainable parameters, optimized by the corresponding hinge loss function, thus we employ a value of $c_k = 0$.

If $\|\mathbf{g}_k(\mathbf{x}_i;\boldsymbol{\theta}) - \mathbf{a}_j^{(k)}\|^2 < r_{kj}^2$, then the data representation $\mathbf{g}_k(\mathbf{x}_i,\boldsymbol{\theta})$ falls inside the j-th class hypersphere, while otherwise, the item lies outside the j-th hypersphere. The loss value is $\mathcal{L}_M > 0$ if and only if the one-class classifier decision function misclassifies $\mathbf{x}_i$, and it is equal to the distance of the data representations in the feature space from the closest hypersphere outer boundary. The compactness of the derived class representations is proportional to the learned value of the corresponding radius r_{kj}.

The above function does not produce loss values for marginal data items, i.e., items lying close to the hypersphere boundaries. The HCP optimization procedure as defined in [6] introduced geometrically inspired tricks to solve those issues. This work considers different optimization terms, inspired by well-established OCC methods. Specifically, we employ a contrastive loss term for items belonging to the same class. To this end, we consider a mini-batch of size N is randomly sampled and the contrastive prediction task is defined on pairs of data representations derived from the mini-batch, resulting in $2N$ data points. For a pair of data representations $\mathbf{z}_1 = \mathbf{g}_k(\mathbf{x}_1,\boldsymbol{\theta}) - \mathbf{a}_j^{(k)}$, $\mathbf{z}_2 = \mathbf{g}_k(\mathbf{x}_2-,\boldsymbol{\theta}) - \mathbf{a}_j^{(k)}$ belonging to the j-th same class, the loss function is defined as follows:

$$\mathcal{L}_C(\mathbf{z}_1,\mathbf{z}_2) = -\log\left(\frac{\exp(\mathbf{z}_1^{\mathrm{T}}\mathbf{z}_2/\mathrm{T})}{\exp(\mathbf{z}_1^{\mathrm{T}}\mathbf{z}_2/\mathrm{T}) + \sum_{\mathrm{i}=2}^{2\mathrm{N}}\exp(\mathbf{z}_1^{\mathrm{T}}\mathbf{z}_{\mathrm{i}}/\mathrm{T})}\right) \qquad (5)$$

where $\mathbf{z}_i$ are the remainder mini batch representations and T is the so-called temperature hyperparameter (a value of $T = 0.25$ was used in all our experiments). The introduction of the above loss term promotes the derivation of similar representations in the feature space, without minimizing their Euclidean distance. However, as pointed out in one-class classification tasks [9], the $\mathcal{L}_C$ might indirectly increase the Euclidean distance, especially if it is very small, which is something that is contradicting to adversarial robustness. Therefore, we follow the same practice and also employ an Angular loss term [9] to complement this contrastive loss:

$$\mathcal{L}_A(\mathbf{z}_1,\mathbf{z}_2) = \|\mathbf{z}_1^T\mathbf{z}_2\|^2. \qquad (6)$$

Finally, we formulate the proposed learning procedure called Robust One-class Classification (ROCC) loss function as the combination of the constraints of the abovementioned optimization terms, as follows:

$$\mathcal{L}_{ROCC} = \mathcal{L}_M + \mathcal{L}_C + \mathcal{L}_A, \qquad (7)$$

where relevant weighting hyperparameters can be considered as well i.e., $\mathcal{L}_{ROCC} = \mu_1\mathcal{L}_M + \mu_2\mathcal{L}_P + \mu_3\mathcal{L}_{NP}$, for adjusting the contribution of each term to the overall loss. In our experiments, weighting parameters were not employed (i.e., $\mu_1 = \mu_2 = \mu_3 = 1$), since it was found that the relevant loss terms produce values that allow smooth optimization.

The proposed optimization terms are employed together with standard Cross entropy loss in the final layer of a neural network, and are advised to be separately implemented in intermediate layers. Determining where is the optimal place to introduce the intermediate supervision constraints is an open problem. Our selection is described in the experimental results. Nevertheless, it should be pointed out that a trade-off between optimal classification accuracy and adversarial robustness should be considered; i.e., the closer to the input the intermediate supervision step is employed with the proposed optimization options, the more the adversarial robustness, while the closer to the output, the better the classification accuracy of the model.

Table 1: Classification accuracy of the competing methods.

Method/Dataset	**CIFAR-10**	**CIFAR-100**	**SVHN**
Vanilla [21]	93.36	**74.04**	96.23
CL [19]	93.77	69.75	95.90
PCL [5]	92.30	68.19	95.37
HCP [6]	93.31	72.83	95.85
ROCC	**94.46**	73.62	**96.31**

4. EXPERIMENTS

This section describes the experiments conducted for evaluating the performance of the proposed optimization scheme. ResNet-101 [21] architecture was employed as the baseline architecture, which is typically employed in image classification problems and produces close to state-of-the-art results. In terms of datasets, we have employed the publicly available CIFAR-10, CIFAR-100 [22] and SVHN [23] datasets which contain 10, 100 and 10 classes, respectively. The classification models were pretrained for 200 epochs using softmax-only and fine-tuned for an additional 400 epochs using the loss function proposed by the different adversarial defense methods. Along with the proposed method (ROCC), we have also employed the Hyperspherical Class Prototype method (HCP), the PCL adversarial defense [5] (PCL) and the closely related center loss function [19] (CL). Hereafter, we refer to the competing methods with their respective acronyms. The loss functions for the proposed ROCC method, the HCP, PCL and CL and were applied in the same ResNet layers (i.e., the 256-dimensional layer-3 and 1024-dimensional final layer). All experiments were implemented in Pytorch 1.6.0.

In our first set of experiments, we compare the classification performance of the competing methods in the employed datasets. Since all datasets are well balanced in terms of contained classes and contain many test samples, we compare the competing methods in terms of classification accuracy. Table 1 reports the obtained classification accuracy in the respective datasets. As can be observed, the proposed method outperforms all other adversarial robustness methods in every case while it even outperformed the vanilla softmax optimization function in two cases. This can be attributed to the fact that the proposed optimization functions only consider how to obtain better representations for each class, thus being compatible with any standard classification loss function.

Table 2: Robustness (classification accuracy) in PGD black-box attack, by using the Vanilla ResNet architecture as attack model.

Method/Dataset	**CIFAR-10**	**CIFAR-100**	**SVHN**
CL [19]	57.60	40.40	86.59
PCL [5]	61.61	42.55	84.94
HCP [6]	60.67	**46.92**	86.50
ROCC	**65.09**	44.97	**86.92**

In our second set of experiments, we evaluate the Robustness of the competing methods to the iterative projected gradient descent (PGD) [3] attack, with a corresponding parameter $e = 0.1$. To this end, we employed the Vanilla ResNet architecture for generating adversarial samples, and inferred their labels by the respective robust models trained using the competing methods. Here it should be noted that this attack is the strongest form of transferability attacks, since the only difference between the attack and target architecture are the network parameters. The results are reported in Table 2. As can be observed, in the 10-class datasets (CIFAR-10, SVHN) the proposed ROCC method outperformed the competition, except the CIFAR-100 case.

Table 3: Cross-method black-box PGD attacks in CIFAR-10.

Attack Method/Robust Method	CL [19]	PCL [5]	HCP [6]	**ROCC**
CL [19]	-	73.46	75.51	**80.53**
PCL [5]	69.83	-	75.21	**78.90**
HCP [6]	78.17	79.47	-	**83.34**
ROCC	**64.01**	**65.16**	**67.23**	-

Finally, in our third set of experiments, we employed the competing architectures to attack each other, as ”host” and target architectures. We again used the PGD attack with $e = 0.1$. Here, it should be expected that the most robust architectures are supposed to a) remain robust in transferability attacks and b) create strong adversarial samples that are able to fool the other defenses. As can be observed, the proposed ROCC method produces the strongest transferability attacks among the competition (red), while at the same time, it remains the most robust in the opposite scenario (bold).

5. CONCLUSION

This work described a method for increasing the robustness of deep neural network-based image classification systems to adversarial attacks, by exploiting and re-formulating one-class classification inspired optimization criteria. Experimental results denoted that the proposed optimization scheme increases adversarial robustness in black-box adversarial attacks without negative effects in classification accuracy. As this work found an interesting link between one-class classification and adversarial robustness, future work could include studying the opposite direction; i.e., adapting adversarial robustness methods for training one-class classification problems. In addition, the proposed criteria should also be studied in other forms of computer vision problems, e.g., regression-based problems such as object detection/tracking.

6. REFERENCES

[1] X. Yuan, P. He, Q. Zhu, and X. Li, “Adversarial examples: Attacks and defenses for deep learning,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 9, pp. 2805–2824, 2019.

[2] Kui Ren, Tianhang Zheng, Zhan Qin, and Xue Liu, "Adversarial attacks and defenses in deep learning," *Engineering*, vol. 6, no. 3, pp. 346–360, 2020.

[3] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[4] 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.

[5] Aamir Mustafa, Salman H Khan, Munawar Hayat, Roland Goecke, Jianbing Shen, and Ling Shao, "Deeply supervised discriminative learning for adversarial defense," *IEEE transactions on pattern analysis and machine intelligence*, 2020.

[6] Vasileios Mygdalis and Ioannis Pitas, "Hyperspherical class prototypes for adversarial robustness," *Pattern Recognition*, p. 108527, 2022.

[7] Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft, "Deep one-class classification," in *International conference on machine learning*. PMLR, 2018, pp. 4393–4402.

[8] Tal Reiss, Niv Cohen, Liron Bergman, and Yedid Hoshen, "Panda: Adapting pretrained features for anomaly detection and segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 2806–2814.

[9] Tal Reiss and Yedid Hoshen, "Mean-shifted contrastive loss for anomaly detection," *arXiv preprint arXiv:2106.03844*, 2021.

[10] Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian Goodfellow, Aleksander Madry, and Alexey Kurakin, "On evaluating adversarial robustness," *arXiv preprint arXiv:1902.06705*, 2019.

[11] Cihang Xie, Yuxin Wu, Laurens van der Maaten, Alan L Yuille, and Kaiming He, "Feature denoising for improving adversarial robustness," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 501–509.

[12] Akhil Goel, Akshay Agarwal, Mayank Vatsa, Richa Singh, and Nalini K Ratha, "Dndnet: Reconfiguring cnn for adversarial robustness," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 22–23.

[13] Xiaoqin Zhang, Jinxin Wang, Tao Wang, Runhua Jiang, Jiawei Xu, and Li Zhao, "Robust feature learning for adversarial defense via hierarchical feature alignment," *Information Sciences*, vol. 560, pp. 256–270, 2021.

[14] Yunpeng Chen, Marcus Rohrbach, Zhicheng Yan, Yan Shuicheng, Jiashi Feng, and Yannis Kalantidis, "Graph-based global reasoning networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 433–442.

[15] Marco Cuturi, "Sinkhorn distances: Lightspeed computation of optimal transport," *Advances in neural information processing systems*, vol. 26, pp. 2292–2300, 2013.

[16] Nikola Jovanović, Mislav Balunović, Maximilian Baader, and Martin Vechev, "Certified defenses: Why tighter relaxations may hurt training?," *arXiv preprint arXiv:2102.06700*, 2021.

[17] Thomas Mensink, Jakob Verbeek, Florent Perronnin, and Gabriela Csurka, "Distance-based image classification: Generalizing to new classes at near-zero cost," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 11, pp. 2624–2637, 2013.

[18] Florian Schroff, Dmitry Kalenichenko, and James Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815–823.

[19] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao, "A discriminative feature learning approach for deep face recognition," in *European conference on computer vision*. Springer, 2016, pp. 499–515.

[20] Vasileios Mygdalis, Anastasios Tefas, and Ioannis Pitas, "K-anonymity inspired adversarial attack and multiple one-class classification defense," *Neural Networks*, vol. 124, pp. 296–307, 2020.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[22] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," *Master's thesis, Department of Computer Science, University of Toronto*, 2009.

[23] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng, "Reading digits in natural images with unsupervised feature learning," *In: Neural Information Processing Systems (NIPS) Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.