kNN-guided Adversarial Attacks (DISCUSSION PAPER)

Fabio Valerio Massoli[0000−0001−6447−1301], Fabrizio Falchi[0000−0001−6258−5313], and Giuseppe Amato[0000−0003−0171−4315]

ISTI-CNR, via G. Moruzzi 1, 56124 Pisa, Italy
{fabio.massoli, fabrizio.falchi, giuseppe.amato}@isti.cnr.it

Abstract In the last decade, we have witnessed a renaissance of Deep Learning models. Nowadays, they are widely used in industrial as well as scientific fields, and noticeably, these models reached super-human performances on specific tasks such as image classification. Unfortunately, despite their great success, it has been shown that they are vulnerable to adversarial attacks - images to which a specific amount of noise imperceptible to human eyes have been added to lead the model to a wrong decision. Typically, these malicious images are forged, pursuing a misclassification goal. However, when considering the task of Face Recognition (FR), this principle might not be enough to fool the system. Indeed, in the context FR, the deep models are generally used merely as features extractors while the final task of recognition is accomplished, for example, by similarity measurements. Thus, by crafting adversarials to fool the classifier, it might not be sufficient to fool the overall FR pipeline. Starting from this observation, we proposed to use a k-Nearest Neighbour algorithm as guidance to craft adversarial attacks against an FR system. In our study, we showed how this kind of attack could be more threatening for an FR system than misclassification-based ones considering both the targeted and untargeted attack strategies.

Keywords: Adversarial Attacks · Face Recognition · Convolutional Neural Networks.

1 Introduction

Since the publication of the AlexNet [6] in 2012, Deep Learning (DL) models started to play a central role in several scientific and industrial fields. Moreover, due to the extremely high computational power reached by the modern GPUs, the use of DL techniques has become state-of-the-art for solving problems such as: vision (e.g., image classification [6], object detection [4]), natural language processing [14], and sentiment analysis [9]. Despite this rebirth, there is a severe threat that poses a strong limit to the use of Deep Neural Networks (DNNs) in real-world scenarios, especially in sensitive contexts such as surveillance systems [11] for instance. It was recently shown that DNNs are vulnerable to adversarial samples [12,1] - images to which a precise amount of noise imperceptible to human eyes is added to fool a model. As shown in the next section of the paper, it is common for the adversaries to craft malicious samples following a misclassification principle, i.e., with the goal in mind of leading a model, such as a classifier, to output a wrong prediction with very high confidence. However, if we consider the case of Face Recognition (FR) systems, the game paradigm deeply changes. Indeed, differently from the classical classification tasks, in the context of FR, a DNN is usually used merely as a features extractor [8,13] while the final task of recognition is realized by performing, for example, similarity measurements among the extracted representations. For instance, we can consider the case of Face Identification (FI) in which the features extracted from a query image has to be compared with a database of known identities to identify a person. Thus, adversarials crafted employing a misclassification principle might not succeed in fooling a similarity-based scheme. Generally speaking, we can divide the attacks against FR systems into two categories, namely: impersonation and evasion attacks. In the former case, the goal of the attacker is to lead the system to recognize two faces as belonging to the same person when that it is not true, while in the latter case he wants the system not to recognize a person. In this context, starting from the idea of deep representation attacks [10], we proposed a variant based on the use of a k-Nearest Neighbour (k-NN) algorithm as guidance, to fool an FR system that relies on similarity measurement to assess queries identity. Moreover, we showed how such an attack could give rise to a greater threat concerning misclassification-based attacks. The rest of the paper is organized as follows: in Section 2, we briefly described some known algorithms to craft adversarial samples. In Section 3 and Section 4, we described our approach and presented the results of our study, respectively. Finally, in Section 5, we reported the conclusion and future perspectives of our work.

2 Adversarial Attacks

Usually, the guiding principle followed while designing adversarial attack algorithms is to lead the model to assign a wrong label with high probability to an image. One of the first studies in this direction was proposed by [12] that designed an optimization procedure based on the L-BFGS algorithm. Specifically, the authors solved the following optimization problem:

$$\min_{\delta} \mathcal{L}(x+\delta,\ y_{gt}) + \lambda \cdot \parallel \delta \parallel_2; \quad \text{with} \ \ x+\delta \in [l,u]^m \,, \tag{1}$$

where $\mathcal{L}$ is the loss function, λ is a parameter found by linear search, y_{gt} is the ground truth label for the input x, δ is the adversarial perturbation, l and u represent the lower and upper bound for the pixel values respectively, and $x+\delta$

is the current adversarial sample crafted from the given input x. A faster way to produce malicious images was proposed by [5], in which the authors proposed to linearize the loss function around the current value of the parameters, thus crafting the adversarial as:

$$x_{adv} = x + \epsilon \cdot \text{sign}(\nabla_x J(\theta, x, y_{true}))\,, \tag{2}$$

The method shown in Equation 2 is known as Fast Gradient Sign Method (FGSM) and allows to craft adversarial samples by only employing a single gradient step. In Equation 2, $\nabla_x J(\theta, x, y)$ is the gradient of the loss function J evaluated around the current neural network status θ, x is the input, y_{true} is its label, and ϵ is the maximum distortion allowed on the input such that $\parallel x - x_{adv} \parallel_\infty < \epsilon$. In [7], the authors proposed a unified view on adversarial attacks and defenses. Specifically, they reformulated the adversarial training for deep models as a "saddle-point" optimization problem

$$\min_\theta \rho(\theta) = \min_\theta \mathbb{E}_{(x,y)\sim\mathcal{D}}\big[\max_{\delta\in\mathcal{S}} L(\theta, x+\delta, y)\big] \tag{3}$$

in which the inner maximization aims at finding an adversarial of a given input x while the outer minimization problem has the goal of reducing the loss after the attack. Finally, they proposed an iterative procedure named Projected Gradient Descent (PGD):

$$x_{N+1}^{adv} = \prod_{x+\mathcal{S}} (x_N^{adv} + \alpha \cdot \text{sign}(\nabla_x L(\theta, x, y)))\,. \tag{4}$$

An improved version of the iterative attacks was proposed in [3] named Momentum-Iterative FGSM (MI-FGSM). The authors integrated a momentum term that helped to escape from poor local maxima during, thus yielding stronger attacks. The main idea is first to evaluate the velocity vector in the gradient direction and then use it to craft the adversarial. The velocity vector is given by

$$g_{N+1} = \mu \cdot g_N + \frac{J(x_N^{adv}, y)}{\parallel \nabla_x J(x_N^{adv}, y) \parallel_1}\,, \tag{5}$$

where $x_0^{adv} = x$, $g_0 = 0$, μ is the decay factor of the running average, and y is the ground truth label. Finally, the adversarial example in the ϵ-vicinity measured by L_2 distance is given by

$$x_{N+1}^{adv} = x_N^{adv} + \alpha \cdot \frac{g_{N+1}}{\parallel g_{N+1} \parallel_2}\,, \tag{6}$$

where $\alpha = \epsilon/T$ with T being the total number of iterations. All the attacks we discussed hitherto focused their attention on fooling a model by letting it predict a wrong class, with high confidence, for the given input. Nevertheless, the situation changes in the context of FR systems. Indeed, in such cases, the neural network is usually not used as a classifier, but as features extractor. For each query image, the model extracts a deep representation, which in turn is compared with a database of known identities. Thus, in those cases, the previous attack techniques might not be effective as one could expect. Following this concern, interesting suggestions on how to design more suited attacks for FR systems came from [10]. In [10], the authors formulated the attack against a DNN by looking at the internal representation of a model, in other words, they focused their attack on making the deep features of an adversarial as close as possible to the one of a target query. More formally, given a doublet $(I_s,\, I_t)$ where the former is the source image and the latter the target one, the goal of the crafting process is to find a new image, I_α, such that its internal representation at a specific layer l of the neural network under attack, $\phi_l(I_\alpha)$, is as close as possible to the one of the target image, $\phi_l(I_t)$. Meanwhile, I_α has to be as close as possible to the source image, I_s, in the input space. The optimization problem can be formulated as follows:

$$I_\alpha = \arg\min_{I} \parallel \phi_l(I) - \phi_l(I_t) \parallel_2^2, \quad \text{subject to} \;\; \parallel I - I_s \parallel_\infty < \delta \,, \tag{7}$$

where δ is the maximum perturbation allowed on each pixel.

3 Proposed Approach

In our experiments, we considered adversarial samples crafted utilizing two guiding principles. On the one hand, we used a typical misclassification-based approach in which the goal of the attacks was to fool a DNN acting as a classifier. On the other hand, we proposed an attack in which we explicitly took into account a more realistic FR setup in which the DNN was used as a features extractor, and the final task of the FR was accomplished by employing similarity measurements among deep representations. In this case, we used our proposed approach where we considered a k-NN algorithm as guidance to craft adversarial samples. Finally, we compared the effectiveness of both approaches to fool an FR system. Regarding the threatened model, we considered the state-of-the-art SeNet-50 [2] from which we extracted the deep features at the penultimate layer. As source dataset, we used the test set of VGGFace2 [2], which comprises 500 identities, that we split into training and experimental sets. To be able to craft misclassification-based attacks, we attached a fully connected (FC) layer on top of the features extractor, and to train it, we used the training split cited above, the SGD optimizer with a learning rate of $1.e^{-3}$, a momentum of 0.9 and a batch size equal to 128. Then, we used the PGD [7] and the MI-FGSM [3] algorithms to craft misclassification-based adversarials.

Concerning our proposed approach, instead, we started by training the k-NN using the class centroids of the VGGFace2 [2] training split images, and then we set k=1. Subsequently, we cast the adversarial crafting procedure as an optimization problem, and we used the L-BFGS algorithm to solve it. Specifically, the main parameters to provide to the algorithm were the following: a starting point from which to begin the optimization procedure, a target, the function to minimize, and a set of parameters that were directly passed to it. The starting point was always a query image from which we wanted to craft an adversarial, while the target was the centroid of the class we wanted to move the adversarial features close to. As a minimization criterion, we used the regression loss. Specifically, the loss was evaluated among the target centroid and the current state of the adversarial sample deep features. At each step of the optimization procedure, we used the k-NN to assess if the deep representation of the adversarials were able to fool the FR system, i.e., if they were close enough to the target centroid deep representation. If so, we stopped the attack; otherwise, we continued the optimization. Moreover, to be sure that the adversarial remained closed enough to the query image so that a human eye could not distinguish between them, we empirically bounded their distance, in the pixel space, to be: $\delta <= 7$ (Equation 7). A schematic view of the overall algorithm is given in Figure 1.

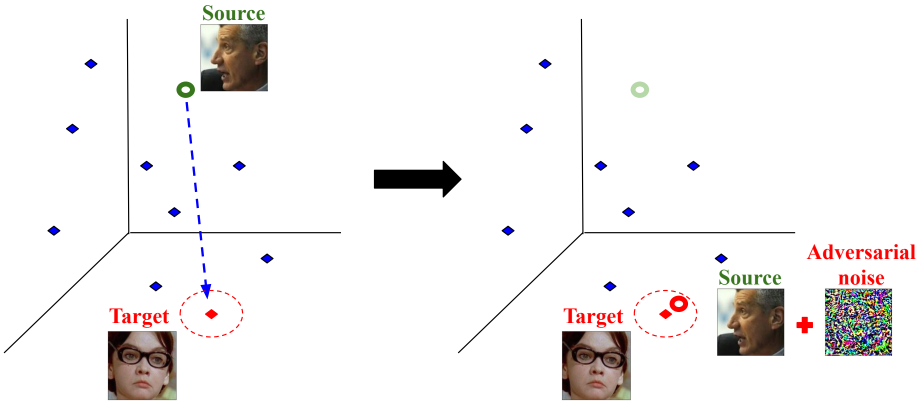

Figure 1. Schematic view of the adversarials generation process. Left: initial features representation of the source image (green circle) and target class centroid (red rhombus). Right: features of the crafted adversarial sample (red solid circle). The blue rhombus represent the centroids of other classes.

In Figure 1, we reported a face image next to the target centroid (red rhombus) only to show an example of the identity represented by the target point.

4 Experimental Results

In this section, we compared the adversarials obtained through our method with the ones obtained by misclassification ones. Specifically, we compared our k-NN guided attack against PGD [7] and MI-FGSM [3] attacks. In Figure 2, we reported a few examples of the manipulated images produced by our algorithm.

In Figure 2, we can notice how similar the images in the first and third columns appear to a human eye. Thus, even though we surfed the features space to craft the adversarials, we have been able to keep the malicious inputs very

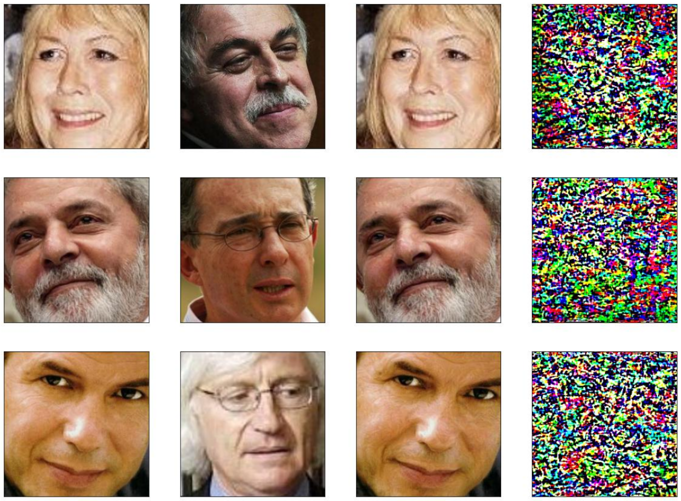

Figure 2. Examples of adversarial attacks crafted by using a k-NN as guidance. From left to right, each column corresponds to: source image (I_s), target class image (I_t), adversarial sample (I_α), adversarial noise, respectively.

close to the original images in the pixel space. Subsequently, we compared the adversarials we generated with the k-NN with the ones crafted employing other attacks. Specifically, we considered attacks focused on fooling the classification task. Since our goal was to show that misclassification-based attacks might not be suitable to produce samples able to fool an FR system, a fundamental property to analyze was the euclidean distance among the adversarials deep features and the centroids of the respective adversarial class. Such a scenario emulates the case in with the features extracted from a query image have to be compared against a known database of known identities. The results are shown in Figure 3 for both targeted and untargeted attacks.

As we can see from Figure 3, the samples generated using the k-NN guided algorithm have an expected value of the euclidean distance from the (adversarial) class centroids lower than the ones generated with other attacks. Such a result holds for targeted as well as untargeted attacks.

As a *gedankenexperiment*, we shall suppose that the dotted-dashed black line in Figure 3 represented the threshold applied by the FR system when it had to decide on a FI task, for instance. In this case, to identify a person, the FR system evaluated the distance of the deep features of the query image from the centroids of the various available identities in the database. Thus, if

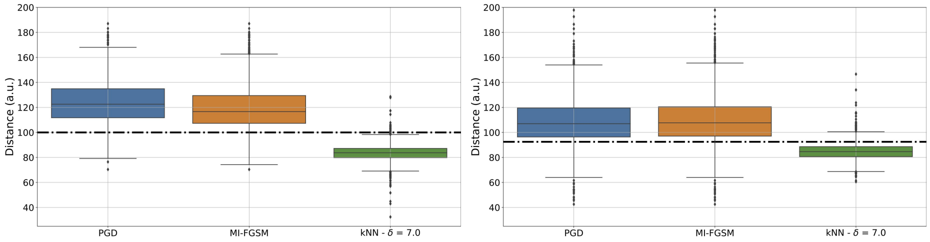

Figure 3. Euclidean distance of the adversarial features from the centroids of the adversarial class. Left: targeted attacks. Right: untargeted attacks. The dashed-dotted line represents a hypothetical threshold used by a FR system to assess the similarity of deep features.

we considered the threshold as shown in Figure 3, we observed that the attacks' success rate for misclassification-based attacks dramatically decreased compared to what happened for the k-NN guided ones. The results are reported in Table 1

Table 1. Attack success rates after applying the distance threshold as shown in Figure 3. The 100% reference value is given by the total number of crafted adversarial.

Attack Algorithm	Attack Success rate with thr (%)	
	Targeted	Untargeted
PGD	2.7	16.1
MI-FGSM	4.5	17.2
k-NN	**97.3**	**88.7**

To sum up, we can assess that, in the attempt of attacking an FR system in which a DNN is used as a backbone features extractor, the use of misclassification-based attacks might not be successful as in classification contexts to fool the recognition system. This happens because, in the FR scenario, the final output does not usually rely on the output of a deep classifier instead of a similarity measurement among deep features. Thus, it is mandatory to consider such aspects while designing attacks.

5 Conclusion and Future Perspectives

The threat of adversarial attacks poses severe limits on the use of deep learning techniques in sensitive real-world applications such as surveillance systems. Several algorithms have been proposed to fool a DNN to lead it to output a wrong prediction with high confidence. As shown, even though a classical misclassification-based attack can fool a neural network used as a classifier, the very same attack might not be strong enough to fool an FR system in which the output is based on similarity measurements carried upon deep features. Thus, different strategies have to be considered. We demonstrated that it is possible to fool an FR system through a deep features attack in which a k-NN is used as a guide in finding the proper perturbation to apply to the input image. Thus, it is possible to bring an adversarial closer to the aimed identity features compared to a misclassification-based. These results hold for the targeted and untargeted attacks. There are several potential extensions of this work. Different techniques to craft adversarials against an FR system might be considered, for example, with k > 1 or based on other guiding principles. On the other hand, it would be interesting trying to detect those kinds of attacks in which the adversary tries to emulate the deep representation of natural images within an attack.

References

1. Biggio, B., Corona, I., Maiorca, D., Nelson, B., Šrndić, N., Laskov, P., Giacinto, G., Roli, F.: Evasion attacks against machine learning at test time. In: Joint European conference on machine learning and knowledge discovery in databases. pp. 387–402. Springer (2013)
2. Cao, Q., Shen, L., Xie, W., Parkhi, O.M., Zisserman, A.: Vggface2: A dataset for recognising faces across pose and age. In: 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018). pp. 67–74. IEEE (2018)
3. Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., Li, J.: Boosting adversarial attacks with momentum. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 9185–9193 (2018)
4. Girshick, R.: Fast r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 1440–1448 (2015)
5. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572 (2014)
6. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012)
7. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083 (2017)
8. Massoli, F.V., Amato, G., Falchi, F.: Cross-resolution learning for face recognition. arXiv preprint arXiv:1912.02851 (2019)
9. Ortis, A., Farinella, G.M., Battiato, S.: An overview on image sentiment analysis: Methods datasets and current challenges. In: Proc. 16th Int. Joint Conf. E-Bus. Telecommun. vol. 1, pp. 290–300 (2019)
10. Sabour, S., Cao, Y., Faghri, F., Fleet, D.J.: Adversarial manipulation of deep representations. arXiv preprint arXiv:1511.05122 (2015)
11. Sreenu, G., Durai, M.S.: Intelligent video surveillance: a review through deep learning techniques for crowd analysis. Journal of Big Data **6**(1), 48 (2019)
12. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R.: Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199 (2013)
13. Wang, F., Cheng, J., Liu, W., Liu, H.: Additive margin softmax for face verification. IEEE Signal Processing Letters **25**(7), 926–930 (2018)
14. Young, T., Hazarika, D., Poria, S., Cambria, E.: Recent trends in deep learning based natural language processing. ieee Computational intelligenCe magazine **13**(3), 55–75 (2018)