arXiv:1902.07776v3 [cs.CV] 23 Sep 2020

Perceptual quality-preserving black-box attack against deep learning image classifiers

Diego Gragnaniello[a,][**], Francesco Marra[a], Luisa Verdoliva[b], Giovanni Poggi[a]

[a]*Department of Electrical Engineering and Information Technology, University Federico II of Naples, via Claudio, 21, 80125 Naples, Italy*
[b]*Department of Industrial Engineering, University Federico II of Naples, via Claudio, 21, 80125 Naples, Italy*

ABSTRACT

Deep neural networks provide unprecedented performance in all image classification problems, including biometric recognition systems, key elements in all smart city environments. Recent studies, however, have shown their vulnerability to adversarial attacks, spawning an intense research effort in this field. With the aim of building better systems, new countermeasures and stronger attacks are proposed by the day. On the attacker's side, there is growing interest for the realistic black-box scenario, in which the user has no access to the neural network parameters. The problem is to design efficient attacks which mislead the neural network without compromising image quality. In this work, we propose to perform the black-box attack along a high-saliency and low-distortion path, so as to improve both the attack efficiency and the *perceptual* quality of the adversarial image. Numerical experiments on real-world systems prove the effectiveness of the proposed approach both on benchmark tasks and actual biometric applications.

© 2020

1. Introduction

Deep Neural Networks (DNNs) are by now widespread in industry and society as a whole, finding application in uncountable fields, from the movie industry, to autonomous driving, humanoid robots, video surveillance, and so on. Well trained DNNs largely outperform conventional systems, and can compete with human experts on a large variety of tasks. In particular, there has been a revolution in all vision-related tasks, which now rely almost exclusively on deep-learning solutions, starting from the 2012 seminal work of Krizhevsky *et al.* [1] where state-of-the-art image classification performance was achieved with a convolutional neural network (CNN).

Recent studies [2], however, have exposed some alarming weaknesses of DNNs. By injecting suitable *adversarial noise* on a given image, a malicious attacker can mislead a DNN into deciding for a wrong class, and even force it to output a *desired* wrong class, selected in advance by the attacker, a scenario described in Fig.1. What is worse, such attacks are extremely simple to perform. By exploiting backpropagation, one can compute the gradient of the loss with respect to the input

image, and build effective adversarial samples by gradient ascent/descent methods. Large loss variations can be induced by small changes in the image, ensuring that adversarial samples keep a good perceptual quality.
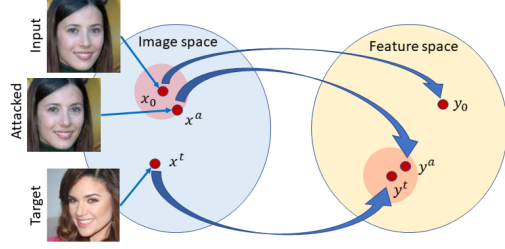
Following [2], several attacks have been proposed[1], mostly gradient-descent methods with the gradient estimated through backpropagation, from the early Fast Gradient Sign Method (FGSM) [3], and its iterative version (I-FGSM) [4], to more recent and sophisticated methods [5, 6, 7]. All such methods, however, require perfect knowledge of the network architecture and weights, a *white box* scenario which is hardly encountered in real-world applications. The focus is therefore shifting towards *black-box* attacks, where nothing is known in advance about the network structure, its weights, or the dataset used for training. In this scenario, the attacker can query the network at will and observe the outcome. This latter can be just a hard label, the distribution of probabilities across the classes (confidence levels), or even a feature vector.

There are many ways to perform black-box attacks to classifiers. A popular approach is to train a surrogate network to mimic the behavior of the target network [5, 8, 9]. The attack is performed on the surrogate and then transferred to the target.

[**]Corresponding author:
e-mail: `diego.gragnaniello@unina.it` (Diego Gragnaniello)

[1]As well as defenses, not mentioned here for brevity.

Fig. 1. Attack scenario. The attacker adds adversarial noise W to the original image x_0 to generate $x^a = x + W$. This attacked image should be similar to x_0 but such that the associated CNN output y^a is close to the target y^t.

However, this calls for full knowledge of the target training set, a precious information rarely available in practice. A more viable alternative, followed here, is to use again gradient-descent [10, 11, 12, 13, 14], with the gradient now estimated by means of suitable queries to the network.

Black-box adversarial attacks should satisfy three main requirements at the same time: being *i)* effective, *ii)* fast, and *iii)* inconspicuous. The first requirement needs no comment, as it is the primary goal of the attack. Efficiency (low number of queries) is also necessary to prevent the attack from becoming exceedingly slow. As for the third requirement, it is important in all systems with human-in-the-loop, such as semiautomatic biometric recognition systems. Unnatural, low quality images may be readily identified by dedicated statistical tests and sent to visual inspection, to be eventually easily detected.

To meet all these requirements, we propose a novel, perceptual quality-preserving (PQP), black-box attack, where quality and effectiveness are ensured in advance by a judicious choice of the perturbation path. Let us briefly summarize the major features and innovative contributions of our proposal: *i)* adversarial noise is injected only in "safe" regions where it has low impact on image quality and high impact on decisions; *ii)* safe regions are identified based on the local gradient of a perceptual quality measure, the structural similarity index (SSIM) [15]; *iii)* the SSIM gradient is computed with negligible cost by means of a simple dedicated CNN; *iv)* all perturbations and queries are 8-bit integer, to successfully attack systems that accept only popular integer formats, such as PNG or JPG.

We assess performance both on widespread benchmark datasets and on a realistic biometric application, including also several defensive strategies. In all cases, the proposed method outperforms reference techniques.

2. Related Work

Early query-based black-box attacks to deep neural networks appeared almost simultaneously in 2017. In [10] a zeroth order stochastic (ZOO) coordinate descent algorithm is proposed, based on the coordinate-wise ADAM optimizer or on coordinate-wise Newtons method. Both solutions modify dynamically the intensity of the attack, the former proving eventually more effective. Several strategies are proposed to reduce the number of queries: dimension reduction via bilinear interpolation, hierarchical attack and importance sampling.

In [11] a greedy local search is carried out aimed at perturbing only the patches that impact most on decision. The adversarial samples, however, exhibit odd patches, which are easily spotted and are sort of a trademark of the attack. Moreover, a very large number of queries is necessary to complete the attack, as observed also in [12]. Along the same line, [16] uses single-pixel adversarial perturbations, extended to groups of five pixels in [17]. The most suitable pixels to attack are selected through a simple genetic algorithm. However, the success rate is rather low, below 80%, and attacks are easily spotted and very fragile to any form of image processing.

[12] uses a gradient estimation strategy followed by the classical FGSM attack, with random grouping to reduce the number of queries. The method compares favourably with ZOO [10], and proves effective also with real-world systems. A further variation of ZOO, called AutoZOOM (autoencoder-based zeroth order optimization method) [13], uses an adaptive random gradient estimation strategy to balance query counts and distortion. Complexity is reduced through subsampling, obtained by using an autoencoder trained off-line with unlabeled data or through bilinear interpolation. Also [14] pursues the main goal of limiting the number of queries. To this end, it makes use of natural evolution strategies (NES) to estimate the black-box output gradient with respect to the attacked input image, and generate suitable adversarial samples. NES is also adopted in [18] with the aim of estimating the adversarial perturbation distribution of the network under attack, so as to draw samples from such a distribution which are likely to fool the classifier.

It is worth underlining that several elegant solutions, *e.g.,* [10, 13, 14], rely on real-valued (that is, non-integer) perturbations. The attack is iteratively refined based on closed-form optimization and estimation methods, and the changes are often quite subtle. These small changes, however, are readily washed out at each query by any form of image compression or even just rounding, with a performance impairment, as demonstrated experimentally in [19]. This is exactly what happens in all systems that accept queries only in integer-valued format, an intrinsic form of defense [20]. Hence, practical attacks should be designed and tested to prove robustness to such scenarios [21].

3. Background

A CNN used for image classification computes a function, $y = f(x; \theta)$, which maps an input image, $x \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, into an output vector, $y \in \mathbb{R}^L$. In the following, with no loss of generality, we will consider color images, hence $n_3 = 3$. The specific function implemented by the CNN depends on the model parameters, θ, namely, the CNN weights which are learned during the training phase to optimize a suitable loss function.

Here, we consider two scenarios of practical interest. In the first one, the classes are known in advance, and the system is asked to decide which class the query belongs to, and how reliable the decision is. Accordingly, it provides in output a vector of probabilities

$$y(i) = \Pr(x \in C_i), \quad i = 1, \ldots, L \qquad (1)$$

also called confidence levels, with C_i the i-th class, and L equal to the number of classes. The decision is made in favor of the

maximum-probability class. In the second scenario, a scalable system is considered, where the classes are not known in advance, and their number grows with time. This applies, for example, to biometric identification systems, where new users keep being enrolled all the time. In this case the CNN is used as a feature extractor. For each input image, x, the CNN generates a discriminative vector of features, y, with length L unrelated with the number of classes, which is then used to perform the actual classification, for example, with a minimum distance rule. In both cases, we assume the CNN to be already trained, with parameters θ defined once and for all. Therefore, in the following we will neglect dependencies on θ.

CNN-based systems are vulnerable to adversarial attacks. To formalize the problem, we define: x_0, the original image, $y_0 = f(x_0)$, the output vector associated with it, $x^a = x_0 + W$, the modified image, with W the additive adversarial noise, $y^a = f(x^a)$, the output vector associated with it, and y^t, the target output vector. Moreover, we introduce $\mathcal{L}(y_1, y_2)$, a suitable loss function measuring vector mismatch in the output domain, and $D(x_1, x_2)$, a suitable measure of the image-domain distortion. The attacker's aim is to generate a new image, x^a, that is close to the original in the source domain, hence small $D(x^a, x_0)$, but whose associated vector is close to the target vector in the output domain, small $\mathcal{L}(y^a, y^t)$, as described pictorially in Fig.1.

We cast the problem as a constrained optimization, setting a limit on the acceptable image distortion, $D_{\max}$. Accordingly, the attacker looks for the image x^a defined by

$$x^a = \arg\min_x \mathcal{L}(f(x), y^t), \quad \text{s.t. } D(x, x_0) \le D_{\max} \quad (2)$$

For typical classification problems, the loss of choice is the cross-entropy. Instead, when the CNN is used for feature extraction, we will consider the Euclidean distance between the extracted feature vectors.

As for the attack strategies, starting from a given image, x, the small perturbation ΔW that maximizes the loss decrease is, by definition, proportional to the gradient of the loss itself with respect to x

$$\Delta W \sim \nabla_x \mathcal{L}(f(x), y^t) \quad (3)$$

If the loss is differentiable and the CNN is perfectly known with its parameters, namely, in the white box scenario, the gradient can be computed through backpropagation, allowing very effective attack strategies. In a black-box (BB) scenario, instead, the attacker has no information about the model architecture, its parameters, θ, or the training set used to learn them. However, the attacker can query the system at will, and use the corresponding outputs to estimate the gradient. For any unit-norm direction of interest, ϕ, the attacker collects the outputs, y^+ and y^-, corresponding to the opposite queries $x^+ = x + \epsilon\phi$ and $x^- = x - \epsilon\phi$, with ϵ suitably small, and estimates the derivative of $\mathcal{L}$ along ϕ as,

$$\frac{\partial \mathcal{L}}{\partial \phi} \simeq \frac{\mathcal{L}(y^+, y^t) - \mathcal{L}(y^-, y^t)}{2\epsilon} \quad (4)$$

If the selected directions are the individual pixels, the whole gradient can be computed, but at the cost of a large number of queries. Hence, practical algorithms approximate gradient-based BB attacks through more efficient strategies.
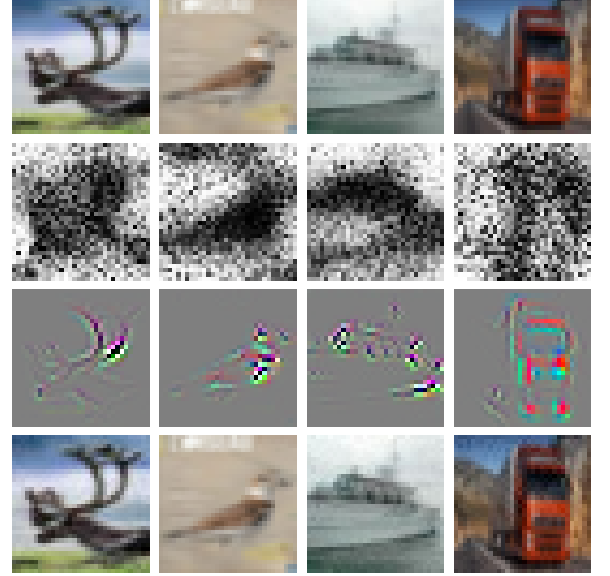

Fig. 2. From top to bottom: original CIFAR10 image, SSIM gradient, saliency map, attacked image. PQP works on low-SSIM gradient pixels, often image edges and keypoints, characterized by high saliency. The attacked image is very similar to the original and does not show unnatural artifacts.

4. Proposed method

We have the twofold goal to drive the classifier towards a desired decision *and* preserve a good image quality. Now, in white-box systems, the gradient of the loss is known, so perturbations are typically taken along the steepest descent path and the effects on distortion are taken into account only a posteriori, by verifying the quality constraint and rejecting unsuitable perturbations. In black-box systems, we cannot compute the gradient of the loss, if not by means of an inordinate number of queries. However, we *can* compute the gradient of the distortion. So, we follow the dual approach, and take perturbations along the path that increases distortion the least, verifying the loss reduction afterwards.

With Fig.2 we explain the rationale of our approach. The first row shows some 32×32-pixel images of the CIFAR10 dataset. On the second row, the corresponding SSIM gradient images are shown. Note that, contrary to what happens with L_p distortion laws, perturbations cause a smaller or larger increase in distortion depending on the local context. In active areas of the image, with keypoints, boundaries and textures, errors are less visible and, accordingly, the SSIM gradient is smaller, while it is larger in flat areas, where errors stand out. Therefore, working in low-SSIM gradient areas, we ensure that attacks have low visibility. On the other hand, it is exactly these features, not background areas, that the CNN takes most into account for classification. This appears clearly in the saliency maps shown on the third row, obtained for the VGG16 net, where bright colors highlight pixels that contribute most to the final decision. Therefore, the very same areas that ensure low quality gradient are also the most important for classification, and SSIM-guided perturbations are not only inconspicuous but also very effective. The fourth row shows the final (successful) adversarial samples

Algorithm 1 PQP

Require: $x_0, y^t, D_{\max}, \mathcal{L}_{\min}, Q, N, \delta, k_{\max}$
Ensure: x^a
1:   $x = x_0$
2:   **while** $D(x, x_0) < D_{\max}$ **and** $\mathcal{L}(f(x), y^t) > \mathcal{L}_{\min}$ **do**
3:      $\mathcal{G} = \nabla_x \text{SSIM}(x, x_0)$
4:      $M_{\text{low}} = \text{Segment}(\mathcal{G}, Q)$ ▷ select low-gradient pixels
5:      $k = 0$
6:      $\Delta\mathcal{L} = 0$
7:      **while** $k < k_{\max}$ **and** $\Delta\mathcal{L} \geq 0$ **do**
8:         $\Delta W = \text{Perturbation}(M_{\text{low}}, N, \delta)$ ▷ generate ΔW
9:         $\mathcal{L}^+ = \mathcal{L}(f(x + \Delta W), y^t)$ ▷ BB query
10:       $\mathcal{L}^- = \mathcal{L}(f(x - \Delta W), y^t)$ ▷ BB query
11:       $\Delta\mathcal{L} = \min(\mathcal{L}^+, \mathcal{L}^-) - \mathcal{L}(f(x), y^t)$
12:       $k = k + 1$ ▷ if $k=k_{\max}$ accept anyway
13:      **end while**
14:      **if** $\mathcal{L}^+ < \mathcal{L}^-$ **then**
15:         $x = x + \Delta W$
16:      **else**
17:         $x = x - \Delta W$
18:      **end if**
19:   **end while**
20:   $x^a = x$;

produced by PQP, barely distinguishable from the originals and free from unnatural artifacts. We note, in passing, that SSIM was already used in the literature [22], but only for white-box attacks, and only for checking the effect of perturbations, not for driving them.

Before moving to describe the proposed PQP method in detail, let us consider a naive algorithm which embodies its concepts in the simplest form. At each step, this algorithm computes the gradient of the SSIM, $\mathcal{G} = \nabla_x \text{SSIM}(x, x_0)$, with respect to the current attack, x, and selects the component where a perturbation causes the least increase of the SSIM

$$(s_1, c_1) = \arg\min_{s,c} |\mathcal{G}(s, c)| \qquad (5)$$

with s the spatial coordinates and c the color band. Then, two new images $x^+ = x + 1_{s_1,c_1}$ and $x^- = x - 1_{s_1,c_1}$ are generated which differ from x by 1 level only at the selected component. The system is probed with these queries, and the one that most reduces the loss becomes the new attacked image. These steps are then repeated until convergence. Note that the SSIM gradient can be computed with negligible cost both in closed form [23], and by means of a simple convolutional network, as we do here (see appendix).

This naive algorithm, however, suffers from two practical problems: *i)* by modifying only one pixel at a time, and by only one level, it becomes exceedingly slow, and *ii)* by choosing deterministically the pixel to modify, it is easily trapped in local minima. The actual PQP method, described by the pseudo-code of Procedure 1, overcomes both these problems. In particular, the procedure of line 8 generates a perturbation, ΔW, which modifies N pixels at once and by $\pm\delta$ levels. Suitable choices of N and δ allow one to speed up the attack considerably with respect to pixel-wise perturbations, without a significant quality impairment. Like in the naive algorithm, the net is queried twice, with $x + \Delta W$ and $x - \Delta W$, and the query which most reduces the loss is accepted. If neither reduces the loss, a new random perturbation is generated, with the same rules. After

$k_{\max}$ unsuccessful attempts, the image is modified anyway to escape local minima.

To avoid local minima, instead, the N pixels to modify are chosen randomly, with the only constraint to belong to the set M_{low} comprising the $Q\%$ pixels with the lowest SSIM gradient. Choosing Q large enough, *e.g.* 50% of the image, ensures high attack variety while avoiding any sharp quality degradation.

A third improvement consists in using *color-coherent* perturbations, such that the sign of perturbations is the same for all color components of a pixel. This choice prevents abrupt, highly visible, variations of the hue, and proved experimentally to improve the attack.

5. Experimental analysis

In this Section, we analyze the performance of the proposed black-box attack both for general-purpose object recognition, with the popular CIFAR dataset, and for a biometric face recognition task, using the MCS2018 dataset. As performance metrics we will consider the success rate of the attack, SR, the number of black-box queries necessary to complete the attack, NQ, and the average distortion of successfully attacked images. To measure distortion we consider not only SSIM, used itself to guide the attack, but also two more full-reference measures, PSNR (peak signal-to-noise ratio), and VIF (visual information fidelity) [24], and two no-reference ones, BRISQUE (blind/referenceless image spatial quality evaluator) [25], and PIQE (perception-based image quality evaluator) [26]. Note that number of queries and image quality are computed only on successfully attacked images.

We will compare results with several baselines and state-of-the-art references: IFD (iterative finite differences) [12] is the main baseline, with pixel-wise perturbations, slow but characterized by high success rate; IGE-QR-RG (iterative gradient estimation with query reduction by random grouping) is a fast method proposed in [12] based on group-wise perturbations; Local Search Attack (LSA) [11] is a method that only perturbs the most salient pixels in the image; AutoZOOM-B is the version of AutoZOOM [13] which uses bilinear interpolation to gain efficiency with no need of prior information; NES-LQ is the limited-query method proposed in [14] based on a natural evolution strategy (NES); $\mathcal{N}$Attack [18] is another recently proposed NES-based approach that draws adversarial perturbations from their estimated distribution. In addition, we provide results also for the white-box I-FGSM (iterative fast gradient sign method) attack [4], as a sort of upper bound for the performance of black-box methods. We use the code published by the authors online [27, 28, 29], with the parameters suggested in the original papers, to which the reader is referred for any further detail. For the proposed method, based on some preliminary experiments, analyzed in Tab.4, we set $Q=66\%$ $N=20$, $\delta=1$, and $k_{\max}=20$. Our own code is published online [30]. Unless otherwise specified, we consider a 8-bit integer setting, with images rounded if necessary after each iteration.

5.1. Object recognition with the CIFAR10 dataset

The popular CIFAR10 object recognition dataset comprises 60000 32×32-pixel RGB images, 50000 for training and 10000
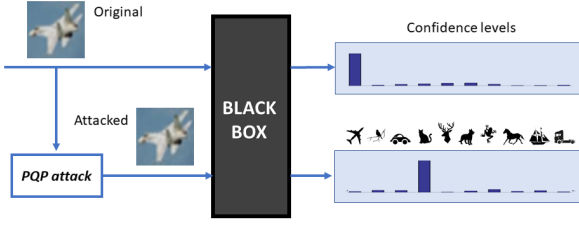
Fig. 3. The CIFAR10 attack scenario. The attacker injects some adversarial noise to increase the confidence level of the target class.

Table 1. Attacking ResNet32 on CIFAR10.

Attack	SR↑	NQ↓	SSIM↑	PSNR↑	VIF↑	BRI.↓	PIQE↓
I-FGSM (WB)	98.69	9	0.991	42.65	10.51	39.09	38.92
IFD	99.01	26639	0.991	42.67	10.53	39.10	39.08
IGE-QR-RG	95.84	5467	0.985	40.84	9.54	39.00	38.92
LSA	0.00	–	–	–	–	–	–
AutoZOOM-B	0.00	–	–	–	–	–	–
NES-LQ	8.74	–	–	–	–	–	–
$\mathcal{N}$Attack	92.97	11459	0.986	37.30	7.67	39.13	42.04
PQP	98.61	1593	0.989	39.77	8.52	39.23	35.42

Table 2. Attacking ResNet32 with no rounding nor SSIM constraint.

Attack	SR↑	NQ↓	SSIM↑	PSNR↑	VIF↑	BRI.↓	PIQE↓
LSA	20.44	–	–	–	–	–	–
AutoZOOM-B	20.54	–	–	–	–	–	–
NES-LQ	100.00	1495	0.880	27.56	4.36	41.16	37.73
PQP	98.61	1585	0.990	39.82	8.45	39.24	35.37

for testing, equally distributed among 10 classes. In the experiments, we consider the most challenging task and target the class with the lowest confidence score. The attack stops either when the confidence of the selected target class exceeds 0.9, in which case we label the attack as successful, or when the SSIM goes below 0.95, a case of unsuccessful attack. Fig.3 depicts the CIFAR10 attack scenario.

Tab.1 shows results for the ResNet32 [31] classifier but similar results have been observed with other deep networks, which are not reported here for brevity. To allow for an easier inspection of results, near each indicator an arrow indicates whether good performances correspond to large ↑ or small ↓ numbers. First of all, we observe a huge gap between the success rates of IFD, IGE-QR-RG, and PQP, all very close to 100%, and those of LSA, AutoZOOM-B, and NES-LQ, much lower. This is not surprising. While the former are designed or can be adapted to work in our integer-valued setting, the latter are intrinsically real-valued, with perturbations that are largely washed out by rounding, leading to mostly ineffective attacks, despite all attempts to optimize parameters for this scenario. With such low SRs, of course, it makes no sense to compute the other performance indicators. All integer-based methods attack quite easily this classifier, ensuring also a very low distortion, with a minimum SSIM of 0.985. With such high success rates and good quality indicators, comparable to those of the white-box attack, the truly discriminative metric is the number of queries. Under this point of view, the proposed method is about 3 times faster than IGE-QR-RG, and 15 times faster than the pixelwise baseline (a comparison with the white-box attack makes no sense). As for $\mathcal{N}$Attack, with over 10000 queries on the average, it turns out to be rather inefficient, contrary to what observed in [18]. This is very likely due to our more challenging scenario, with worst-case targeted attacks, which renders impractical to estimate the distribution of adversarial noise. In summary, PQP ensures successful attacks with a small number of queries. Moreover, no quality loss is observed, not only in terms of SSIM but of all quality measures, even with a significant improvement in terms of the no-reference PIQE.

The above experiment shows that some inherently "floating-point" attacks, LSA, AutoZOOM, and NES-LQ, do not fit well our integer-valued scenario, and hence will not be considered anymore in our analysis. However, to establish a fair comparison with these state-of-the-art methods, we carry out a further experiment on the CIFAR10 dataset, relaxing both the 8-bit integer constraint and the SSIM contraint. Instead, we keep targeting the worst class and requiring a confidence level beyond

0.9 to declare a success. Results are reported in Tab.2. Even in this favourable scenario, AutoZOOM-B and LSA keep providing poor results, with a success rate around 20%. NES-LQ, instead, ensures an excellent success rate, but the image quality shows a serious impairment uniformly for all measures, with a PSNR loss of about 10 dB, and a SSIM decrease of about 0.1 with respect to PQP. Overall, these results suggest that the proposed algorithm works much better than reference methods not only in our challenging "real-world" scenario, but also in a more abstract scenario in which all constraints on data format are removed.

To complete our realistic analysis on the CIFAR10 dataset, we consider the presence of countermeasures. Indeed, with the diffusion of adversarial attacks, some defense strategies are becoming widespread, like the so-called NN (nearest neighbor) defense. For each test image, x, the net extracts a feature vector, called F here, computes its distance from the centroid of all classes, $\overline{F}_1, \ldots, \overline{F}_{10}$, and decides in favor of the closest one. The NN classifier is somewhat inferior to the standard one but more robust to adversarial attacks, and fully sensible for biometric authentication systems. Here, we consider NN defense with the worst-case attack, where the target class is $t = \arg\max_c \|F - \overline{F}_c\|_2$. The attack is successful if the feature vector of the attacked image, F^a, becomes closer to $\overline{F}_t$ than to all other centroids, and the distance $\|F^a - \overline{F}_t\|_2$ goes below a given threshold γ. We set $\gamma = 0.2$, which is the average distance between the features of a class and the corresponding centroid, while the average distance between the test images and the centroid of their target-class (distance before attack) is 0.81. Results reported in Tab.3 show that the new classifier is more robust to attacks. Success rates reduce significantly, even for the white-box reference, going down to 81% for IGE-QR-RG and to 84% for $\mathcal{N}$Attack. PQP, instead, keeps ensuring a good success rate, the smallest number of queries among black-box methods, and a very good image quality.

Table 3. Attacking ResNet32 on CIFAR10 with NN defense.

Attack	SR↑	NQ↓	SSIM↑	PSNR↑	VIF↑	BRI.↓	PIQE↓
I-FGSM (WB)	92.03	23	0.984	37.07	7.73	39.18	35.99
IFD	92.37	71420	0.984	37.09	7.73	39.23	35.99
IGE-QR-RG	81.37	11278	0.980	36.04	7.31	39.49	36.79
$\mathcal{N}$Attack	84.00	38419	0.977	34.65	6.91	39.88	39.99
PQP	95.81	3991	0.987	36.07	6.61	39.16	34.57

Table 4. Choosing parameters on the MCS2018 dataset ($\gamma = 1.0$).

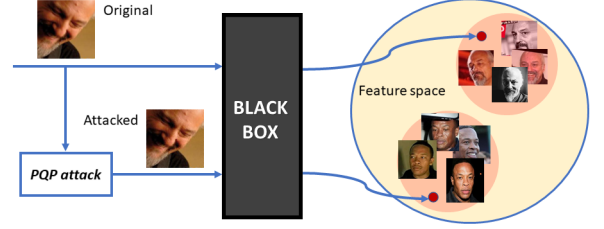| Q	N	δ	SR↑	NQ↓	SSIM↑	PSNR↑
66%	20	1	94.20	13265	0.976	38.62
33%	–	–	98.00	18348	0.981	37.74
100%	–	–	72.00	*10636	*0.968	*39.41
–	10	–	97.60	20093	0.981	39.55
–	40	–	85.00	8507	0.972	37.69
–	–	2	63.20	*5163	*0.966	*36.91
–	–	3	27.80	–	–	–

Results marked with "*" may be overly optimistic.

5.2. Face recognition with the MCS2018 dataset

Evaluating the robustness of a face recognition system is of primary importance to improve its performance in an adversarial setting. This task was originally proposed in the *"Adversarial Attacks on Black Box Face Recognition"* competition, in the context of the MachinesCanSee conference (MCS2018) held in Moscow in June 2018. The reference scenario is depicted in Fig.4. A black-box face recognition system is provided, which extracts a 512-component unit-norm feature vector from each submitted 112×112 RGB image. This vector is then fed to a database to single out the best matching identity. In the competition, the attacker was asked to imperceptibly modify an input image associated with a given identity, with the aim of tricking the system into recognizing a different specific identity. 1000 pairs of source-target identities (S_i, T_i) were provided, with 5 images for each identity: $S_i \rightarrow \{x_{i1}^s, \dots, x_{i5}^s\}$, and $T_i \rightarrow \{x_{i1}^t, \dots, x_{i5}^t\}$. The goal of the attacker was to move the feature vector of the attacked image F_{ij}^a close to the centroid, $\overline{G}_i$, of the target identity features, subject to $\text{SSIM}(x_{ij}^s, x_{ij}^{s,a}) \geq 0.95$. In the competition, we (the GRIP team) obtained eventually the lowest average distance [32], D=0.928, using a preliminary version of the proposed algorithm.

Here, we modify slightly the attacker's goal, in order to define a success rate and allow for a meaningful comparison with the other tasks. We declare a success for image x_{ij}^s when $\|F_{ij}^a - \overline{G}_i\| < \gamma$, with γ a suitably chosen threshold, a failure when the SSIM goes below 0.95. Since the feature vectors have unitary norm, unrelated vectors are nearly orthogonal, with a distance close to $\sqrt{2} \simeq 1.414$. Features associated to the same identity are much closer, with average intra-class distance 0.903. Therefore, we consider two cases, $\gamma = 1.0$ and $\gamma = 0.9$, the latter being the most challenging.

To select the PQP's parameters used in all experiments, Q=66%, N=20 and δ=1, we carried out some preliminary tests on the MCS2018 dataset. Tab.4 shows, for γ=1, the effect of

Fig. 4. The MCS2018 attack scenario (implicit NN defense). The injected adversarial noise brings the output feature vector close to the feature vectors associated with the sample images of the target subject.

perturbing these parameters with respect to their default values. Note that statistics are computed only on the successful attacks, hence they are overly optimistic when SR ≪ 100%, in which case we mark results with an asterisk. These tests prove clearly the importance of working in the low-SSIM gradient region, as blind attacks (Q=100%) cause a sharp decrease of both success rate and quality. Moreover, they speak against increasing δ to speed up the attack, while increasing N is less detrimental to quality.

Tab.5 shows results for attacks to the MCS2018 face recognition system (to carry out experiments in a reasonable time, we implemented a replica of the competition system based on the organizer's description and software, and testing strict compliance of results). Given the stringent MCS2018 rules, attacks are much more difficult than in the CIFAR10 case. Even with the larger distance threshold, γ=1.0, a level which does not always ensure reliable face identification, the IGE-QR-RG attack fails most of the times, while $\mathcal{N}$Attack only reaches 64% success rate. At the same level, however, the proposed PQP has a success rate beyond 95%, even better than I-FGSM and IFD. With γ=0.9 all performance metrics worsen, of course, and only PQP keeps ensuring a success rate over 80%. It is also worth underlining the large improvement with respect to PQP-α, the preliminary version which ranked first in the MCS2018 competition, called "grip" in that context. As for image quality, it is guaranteed in all cases by the 0.95 constraint on the SSIM, and PQP provides always the largest SSIM. Nonetheless, turning to the other quality measures, results are more heterogeneous. The $\mathcal{N}$Attack, in particular, seems to ensure a better image quality than PQP according to the VIF and BRISQUE measures, but much worse in terms of PIQE. These contrasting indications are in part due to the fact that results are computed on a limited and biased sample (only successful attacks). Another warning concerns no-reference quality estimators, which are typically trained on large patches drawn from high-resolution images, whose statistics largely differ from those used in our experiments. All these facts suggest to take these data with a grain of salt, and refer ultimately to visual inspection of attacked images, as in Fig.6.

In order to gain a better insight into the performance of all methods, we consider a "lighter" version of the MCS2018 system, which simulates a less watchful human visual inspection. Accordingly, the constraint on the SSIM is replaced by a much weaker constraint on PSNR, required only to exceed 30 dB. With these rules, success rates are always very high, never less

Table 5. Attacking the MCS2018 face recognition system.

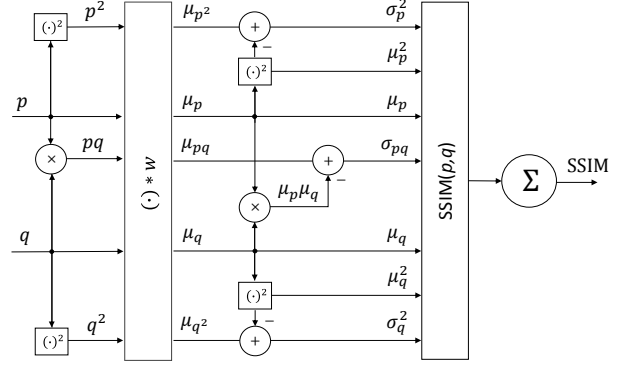| γ	Attack	SR↑	NQ↓	SSIM↑	PSNR↑	VIF↑	BRI.↓	PIQE↓
	I-FGSM (WB)	84.62	7	0.975	40.40	5.65	30.94	23.69
1.0	IFD	81.74	454743	0.971	39.84	7.14	30.80	23.41
	IGE-QR-RG	16.44	–	–	–	–	–	–
	$\mathcal{N}$Attack	64.08	*17798	*0.970	*38.23	*8.28	*31.73	*27.57
	PQP-α (grip)	66.68	*11883	*0.965	*34.46	*6.30	*29.53	*13.36
	PQP	95.80	13400	0.978	37.13	6.34	34.66	14.69
	I-FGSM (WB)	62.76	*9	*0.970	*39.25	*5.80	*31.46	*24.20
0.9	IFD	56.86	*536919	*0.967	*38.93	*7.58	*31.48	*25.22
	IGE-QR-RG	4.66	–	–	–	–	–	–
	$\mathcal{N}$Attack	35.20	*24066	*0.966	*37.36	*7.92	*32.09	*25.78
	PQP-α (grip)	39.88	*14550	*0.962	*33.87	*6.44	*30.39	*13.44
	PQP	83.52	18244	0.973	35.90	6.61	34.11	15.24

Results marked with "*" may be overly optimistic.

Table 6. Attacking a "lighter" MCS2018 face recognition system.

γ	Attack	SR↑	NQ↓	SSIM↑	PSNR↑	VIF↑	BRI.↓	PIQE↓
	I-FGSM (WB)	100.00	9	0.968	39.71	5.45	30.21	23.16
1.0	IFD	100.00	544140	0.964	39.18	6.67	30.28	22.34
	IGE-QR-RG	99.98	17498	0.912	35.02	5.81	29.66	16.97
	$\mathcal{N}$Attack	99.12	25503	0.954	37.16	7.54	30.52	25.32
	PQP-α (grip)	99.86	21286	0.954	33.90	5.97	27.76	13.36
	PQP	100.00	14620	0.977	36.98	6.19	34.97	14.71
	I-FGSM (WB)	98.98	15	0.949	37.63	5.45	30.17	23.18
0.9	IFD	99.16	893743	0.944	37.31	6.75	30.28	22.39
	IGE-QR-RG	98.36	27410	0.878	33.56	5.79	29.70	17.16
	$\mathcal{N}$Attack	90.90	39728	0.940	35.74	6.70	30.40	22.48
	PQP-α (grip)	97.20	59440	0.939	33.05	5.97	27.70	13.44
	PQP	99.60	30967	0.965	35.39	6.19	34.98	14.79

than 90%, allowing for the collection of significant performance data. In these conditions, PQP is about as fast as IGE-QR-RG, but ensures a much better quality according to all measures except BRISQUE. Also the $\mathcal{N}$Attack exhibits better VIF and BRISQUE numbers but much worse SSIM and PIQE numbers and, more important, it requires a much larger number of queries. Turning to the most interesting case of γ=0.9, PQP keeps providing an average SSIM above 0.95, while for IGE-QR-RG this drops well below 0.9, which entails very likely visible distortions. The $\mathcal{N}$Attack keeps exhibiting contrasting image quality results, but now with a much smaller success rate. In general, we believe SSIM and PSNR to be the most reliable indicators of quality, at least for our scenario characterized by very small images subject to adversarial attacks.

This is confirmed by Fig.6, showing visual results for the various types of attacks on two MCS2018 images. Apparently, the "bearded man" image is relatively simple to attack, and all methods introduce only limited distortion. Nonetheless, for γ=0.9, weird geometrical patterns and color distortions appear in the flat areas of images attacked by all methods except PQP. The "girl" image is obviously more difficult to attack, and visible distortions arise even for γ=1. Also in this case, however, the PQP images appear more natural than the others, showing a lower quality than the original but without heavy patterns and color distortions.

Fig. 5. Computing SSIM gradient by back-propagation on a *ad hoc* CNN.

6. Conclusions

We have proposed a new black-box method for attacking deep learning-based classifiers. The image under attack is iteratively modified until the desired classification and confidence level are achieved. However, changes are constrained to the low-SSIM gradient part of the image, where perturbations have a limited impact on perceptual quality and, often, high impact on classification. As a result, effective adversarial samples with high perceptual quality are rapidly generated. This makes the proposed approach suitable to attack semi-automatic biometric authentication systems, usually implemented in surveillance systems of smart cities.

Experiments carried out in two quite different scenarios testify on the potential of the proposed approach. Results are always very good, both with 8-bit integer and real-valued images, and do not impair much when some defense strategies are enacted. The perceptual quality is always good, visible distortions appear only in very small images, while adversarial noise introduced in regular-size images is always inconspicuous.

Appendix: Computing SSIM gradient by an *ad hoc* CNN

The SSIM between two homologous image patches p and q reads as

$$\text{SSIM}(p,q) = \frac{(2\mu_p\mu_q + \epsilon_1)(2\sigma_{pq} + \epsilon_2)}{(\mu_p^2 + \mu_q^2 + \epsilon_1)(\sigma_p^2 + \sigma_q^2 + \epsilon_2)} \qquad (6)$$

Moments are computed with a suitable weighting window w as in [15]

$$\begin{aligned}
\mu_q &= \sum_{s \in \Omega} w(s)q(s) \\
\sigma_q^2 &= \sum_{s \in \Omega} w(s)(q(s) - \mu_q)^2 \\
\sigma_{pq} &= \sum_{s \in \Omega} w(s)(p(s) - \mu_p)(q(s) - \mu_q)
\end{aligned} \qquad (7)$$

with s spanning the patch coordinates Ω. Similar formulas hold for μ_p, σ_p^2.

To compute such weighted moments we use the simple CNN shown in Fig.5, comprising a single convolutional layer and some further block for algebraic operations. Hence, the gradient of the SSIM with respect to the whole image is then obtained by the usual back-propagation of the loss.
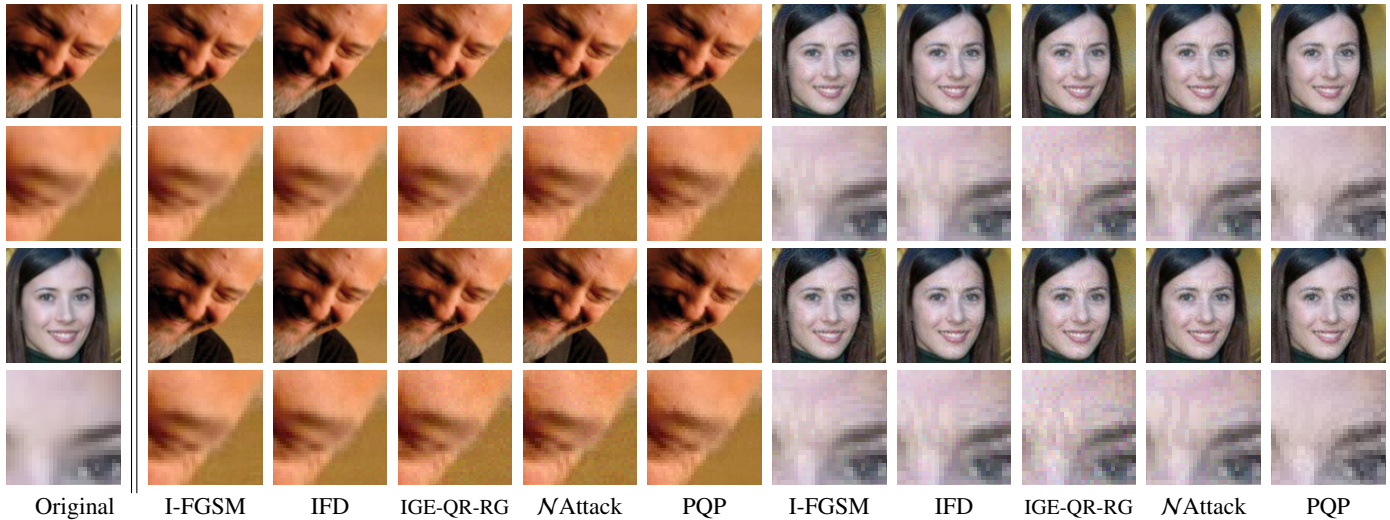
Fig. 6. Example attacks on two MCS2018 images with $\gamma = 1.0$ (top) and $\gamma = 0.9$ (bottom). Originals are shown in the left column. For the "bearded man", a light adversarial noise is sufficient, and attacked images show little signs of distortion, more visible in the flat areas except for PQP. A heavier adversarial noise is necessary to attack the "girl", with visible distortions and many "weird" patters. PQP images are also distorted, but keep a natural appearance.

References

[1] A. Krizhevsky, I. Sutskever, G. Hinton, Imagenet classification with deep convolutional neural networks, in: Neural Information Processing Systems (NIPS), 2012, pp. 1097–1105.

[2] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, R. Fergus, Intriguing properties of neural networks, in: International Conference on Learning Representations, 2014.

[3] I. Goodfellow, J. Shlens, C. Szegedy, Explaining and harnessing adversarial examples, in: International Conference on Learning Representations, 2015.

[4] A. Kurakin, I. Goodfellow, S. Bengio, Adversarial examples in the physical world, arXiv preprint arXiv:1607.02533 (2016).

[5] N. Carlini, D. Wagner, Towards evaluating the robustness of neural networks, in: IEEE Symposium on Security and Privacy, 2017, pp. 39–57.

[6] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, A. Vladu, Towards deep learning models resistant to adversarial attacks, arXiv:1706.06083 (2017).

[7] M. Cisse, Y. Adi, N. Neverova, J. Keshet, Houdini: Fooling deep structured prediction models, arXiv preprint arXiv:1707.05373 (2017).

[8] N. Papernot, P. McDaniel, I. Goodfellow, Transferability in machine learning: from phenomena to black-box attacks using adversarial samples, https://arxiv.org/abs/1605.07277 (2016).

[9] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, A. Swami, Practical black-box attacks against machine learning, in: Proceedings of the 2017 ACM on Asia conference on computer and communications security, 2017, pp. 506–519.

[10] P.-Y. Chen, H. Zhang, Y. Sharma, J. Yi, C.-J. Hsieh, Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models, in: Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, 2017, pp. 15–26.

[11] N. Narodytska, S. Kasiviswanathan, Simple black-box adversarial perturbations for deep networks, in: CVPR Workshops, 2017.

[12] A. Bhagoji, W. He, B. Li, D. Song, Exploring the space of black-box attacks on deep neural networks, arXiv preprint arXiv:1712.09491 (2017).

[13] C.-C. Tu, P. Ting, P.-Y. Chen, S. Liu, H. Zhang, J. Yi, C.-J. Hsieh, S.-M. Cheng, Autozoom: Autoencoder-based zeroth order optimization method for attacking black-box neural networks, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 33, 2019, pp. 742–749.

[14] A. Ilyas, L. Engstrom, A. Athalye, J. Lin, Black-box adversarial attacks with limited queries and information, in: International Conference on Machine Learning, 2018, pp. 2137–2146.

[15] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Transactions on Image Processing 13 (2004) 600–612.

[16] J. Su, D. V. Vargas, K. Sakurai, One pixel attack for fooling deep neural networks, IEEE Transactions on Evolutionary Computation 23 (2019) 828–841.

[17] J. Su, D. Vargas, K. Sakurai, Attacking convolutional neural network using differential evolution, IPSJ Transactions on Computer Vision and Applications (2019).

[18] Y. Li, L. Li, L. Wang, T. Zhang, B. Gong, Nattack: Learning the distributions of adversarial examples for an improved black-box attack on deep neural networks, in: International Conference on Machine Learning, 2019, pp. 3866–3876.

[19] Y. Duan, Z. Zhao, L. Bu, F. Song, Taking care of the discretization problem: A black-box adversarial image attack in discrete integer domain, arXiv preprint arXiv:1905.07672v4 (2019).

[20] N. Das, M. Shanbhogue, S. Chen, F. Hohman, S. Li, L. Chen, M. Kounavis, D. Chau, SHIELD: Fast, practical defense and vaccination for deep learning using JPEG compression, in: ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018.

[21] F. Marra, D. Gragnaniello, L. Verdoliva, On the vulnerability of deep learning to adversarial attacks for camera model identification, Signal Processing: Image Communication 65 (2018) 240–248.

[22] A. Rozsa, E. M. Rudd, T. E. Boult, Adversarial diversity and hard positive generation, in: CVPR Workshops, 2016, pp. 25–32.

[23] A. Avanaki, Exact histogram specification optimized for structural similarity, arXiv preprint arXiv:0901.0065 (2008).

[24] H. R. Sheikh, A. C. Bovik, Image information and visual quality, IEEE Transactions on image processing 15 (2006) 430–444.

[25] A. Mittal, A. K. Moorthy, A. C. Bovik, No-reference image quality assessment in the spatial domain, IEEE Transactions on image processing 21 (2012) 4695–4708.

[26] N. Venkatanath, D. Praneeth, M. C. Bh, S. S. Channappayya, S. S. Medasani, Blind image quality evaluation using perception based features, in: 2015 Twenty First National Conference on Communications (NCC), IEEE, 2015, pp. 1–6.

[27] Ifg github repository, https://github.com/sunblaze-ucb/blackbox-attacks, 2017.

[28] AutoZOOM github repository, https://github.com/IBM/Autozoom-Attack, 2018.

[29] Query limited black-box attack github repository, https://github.com/labsix/limited-blackbox-attacks, 2018.

[30] PQP github repository, https://github.com/dgragnaniello/PQP, 2019.

[31] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[32] Results of the CODALAB MCS2018 competition, https://competitions.codalab.org/competitions/19090#results, 2018.