Robust Deep Face Recognition

Leo Mehr
`leomehr@stanford.edu`

Luca Schroeder
`lucschr@stanford.edu`

1 Introduction

Recent advances in facial recognition technology have revolutionized a plethora of applications, ranging from device authentication (e.g. Apple's Face ID) and e-commerce (e.g. Mastercard's 'selfie' payment technology) to public safety (e.g. police identifying criminals at large public events). And in the near future, facial recognition could be deployed in even higher-stake situations, such as targeting on military rifles [1] and management of pain medication for patients [2].

Thus, the risks of attacks on face identification technology and the consequences of misidentification grow ever larger—and so it is crucial to understand how brittle face identification currently is and how robust it can be made. These questions are particularly relevant in the wake of recent research on adversarial examples that has shown the vulnerability of DNN approaches to even small perturbations in natural images.

Our project seeks to: (i) evaluate the robustness of a state-of-the-art face recognition neural network by deploying a variety of modern white- and black-box attacks; and (ii) develop practical defenses that deep face recognition systems can use and assess the efficacy of these defenses against different classes of attacks.

Our code can be found on Github at `https://github.com/leomehr/cs230project`.

2 Dataset

We evaluate our attacks and defenses on the Labeled Faces in the Wild (LFW) dataset [3], widely recognized as the *de facto* face verification benchmark. The LFW dataset contains 13,233 images of 5,749 public figures, with 1,680 individuals represented by 2+ photos in the dataset. The LFW test set is a sequence of pairs of images which need to be classified as being photos of the same person or photos of different people. Figure 1 shows examples of both types of pairs. This face verification task is the basic building block of a face recognition system, which may compare a new face image with a number of images in its database to authenticate or identify users.

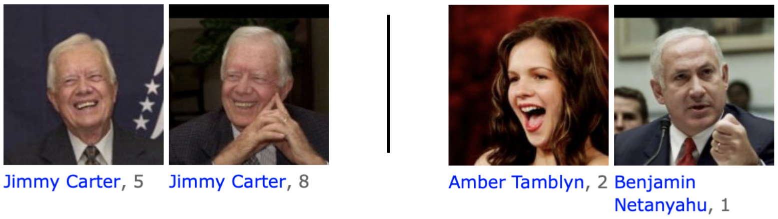


Figure 1: Sample LFW test pairs, to be classified as "same" (left) and "not same" (right)

LFW images are aligned with a Multi-task Cascaded Convolutional Network (MTCNN) [4] and scaled to 160×160. 0-1 RGB scale is used (i.e. input images are $\in [0,1]^{160 \times 160 \times 3}$).

,

3 Methods

The model we target with our attacks is a TensorFlow implementation of FaceNet [5], which gives a mapping between face images and a 128-dimensional embedding in a Euclidean space. This model uses the Inception-ResNet-v1 (GoogLeNet) architecture [6] and is pre-trained on the VGGFace2 dataset [7], which contains more than 3 million face images of 9,000+ individuals. The model achieves 99.65% accuracy on LFW.

To stage our attacks we use the Python package Foolbox [8], which contains reference implementations for many popular adversarial attack techniques. As Foolbox expects a (multiclass) classifier which takes one input image we wrap our FaceNet model as shown in Figure 2. For each LFW test pair (f_1, f_2), we fix the second face image f_2 and compute its embedding x_2. The adversary then feeds perturbed versions of f_1 into our model and our "classifier" outputs probabilities that the pairs of images are of the "same" class or of the "different" class. If the images were photos of the same person to begin with, the adversary's goal is to find perturbation Δ such that $d(\texttt{embedding}(f_1 + \Delta), x_2) > \texttt{threshold}$, i.e. such that FaceNet thinks the two faces are of different people. This set-up follows that in [9].

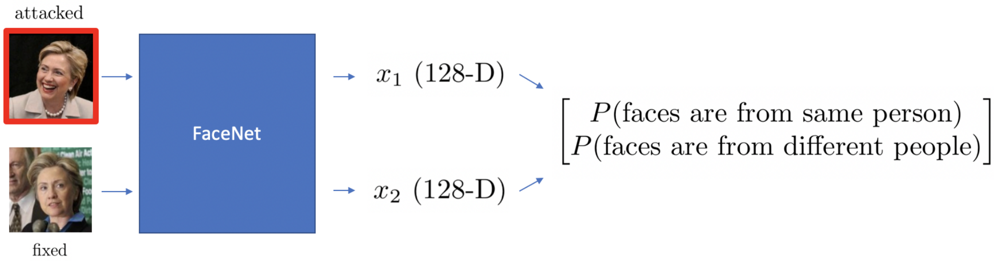

attacked
fixed
FaceNet
x_1 (128-D)
x_2 (128-D)
$\begin{bmatrix} P(\text{faces are from same person}) \\ P(\text{faces are from different people}) \end{bmatrix}$

Figure 2: Turning FaceNet into a classifier for Foolbox

We focus on four popular adversarial attack techniques:

- Fast Gradient Sign Method (FGSM) [10]: find smallest ϵ such that $f_1 + \Delta(\epsilon)$ is misclassified, where $\Delta = \epsilon \cdot \text{sign}(\nabla_f \mathcal{L}(f_1, \ell_0))$, where ℓ_0 is the true classification label. i.e. FGSM perturbs the image in the direction of the gradient, increasing loss and triggering misclassification;
- Deep Fool [11]: iteratively perturb f_1 in the direction of the gradient of the loss function, generating a sequence of perturbations $\epsilon_0, \epsilon_1, \ldots$ that terminates once the decision boundary is crossed, and yields the final perturbation $\epsilon = \sum_i \epsilon_i$. Deep Fool is roughly an iterative extension of FGSM and while it is more computationally expensive, it produces more effective adversarial examples and with much smaller perturbations;
- Additive Uniform Noise Attack (Uniform): i.i.d. Uniform noise is added to the image; the standard deviation of the noise is increased until misclassification is achieved;
- Additive Gaussian Noise Attack (Gaussian): same as Uniform but with i.i.d. Gaussian noise.

4 Preliminary Results

Since adversarial attacks can be time-consuming to run it is impractical to generate an entire adversarial dataset from LFW test pairs. Instead, we randomly chose 50 LFW test pairs and ran each attack on these. Similar to [9] we report the performance of the attacks on two conceptually distinct tasks:

1. *Anonymization.* If (f_1, f_2) are photos of the same person, the attacker tries to perturb f_1 such that $f_1' = f_1 + \Delta$ and f_2 are regarded as being faces of different people. An example of a real-world anonymization attack would be a person of interest trying to mask their presence from law enforcement face recognition software.

	Succ%, $\|\|\Delta\|\| < \infty$		Succ%, $\|\|\Delta\|\| < 5$		Succ%, $\|\|\Delta\|\| < 1$		Average $\|\|\Delta\|\|$	
	Anon.	Impers.	Anon.	Impers.	Anon.	Impers.	Anon.	Impers.
FGSM	100%	79.17%	100%	75%	50%	16.66%	1.23	2.41
DeepFool	100%	100%	100%	100%	92.31%	70.83 %	0.50	0.90
Uniform	100%	4.16%	7.69%	0%	3.84%	0%	33.89	11.42
Gaussian	100%	8.33%	7.69%	0%	3.84%	0%	33.36	22.47

Table 1: Success rates of attacks with constrained perturbation size $||\Delta||$, and average perturbation size $||\Delta||$ of successful attacks. Performance segmented into anonymization/impersonation tasks.

2. *Impersonation*. If (f_1, f_2) are photos of different people, the attacker tries to perturb f_1 such that $f_1' = f_1 + \Delta$ and f_2 are regarded as being faces of the same person. An example of a real-world impersonation attack would be an individual trying to gain access to someone else's device which uses face authentication.

Table 1 gives the success rate (percentage of test pairs for which a misclassification was achieved) for the 4 different attacks. It also shows how the success rate changes as the max perturbation size is constrained, and the average perturbation size (L2 norm of Δ) for successful misclassification attacks with no such size constraint. There are many interesting insights here that we will delve into in our full report, but in brief: 1) anonymization is a much easier task than impersonation, with 100% success rate for all attacks when perturbation size is unconstrained; 2) white-box attacks that require the gradient are successful much more often and with order-of-magnitude smaller perturbations than black-box additive noise attacks; 3) the best attack is DeepFool and this is able to essentially always succeed with essentially imperceptible changes to the source image. In Appendix A we have included representative sequences of images showing the first source image f_1, the perturbation Δ, the adversarial image $f_1 + \Delta$, and the second source image f_2 for both impersonation and anonymization attacks.

5 Future Work

Having demonstrated that there is indeed a robustness problem for face recognition systems our next step will be to develop and evaluate potential defenses. Since we have limited compute resources traditional defenses like training on adversarial examples or defensive distillation are unrealistic; therefore our focus will be on developing lightweight defenses which do not require retraining the FaceNet model.

We propose two classes of defenses that we plan to implement: (1) denoising and (2) attack detection. (1) Denoising an image effectively averages the values of nearby pixels. Recent research has illustrated the success of denoising at the feature level inside networks [12], yet this is an expensive approach that requires modifying the architecture and retraining. Our suggested approach is more lightweight by only modifying the input image, and we hypothesize it can hinder the efficacy of black-box random noise attacks, although we are not confident about its performance on the gradient-based white box attacks of FGSM and Deep Fool. Another potential direction here is to add an additional component that applies traditional probabilistic image restoration techniques to input images to remove perturbation noise before feeding them to the FaceNet CNN, leveraging the continuity of natural images. The second class of defense is (2) attack detection, in which we would attempt to classify whether an image is attacked before running our identification network. If an attack is predicted, then we can discard the image and effectively cause the identification task to automatically fail. This strategy is inspired by Safety Net [13], and we believe that a traditional CNN architecture may work well for this binary classification task. Note that in the real world attack detection would be an effective defense against impersonation attacks (since the face recognition system fails and denies access) but not against anonymization attacks (since the face recognition system fails and does not identify the individual).

Additionally, on the attack side we may try to gain insight into the transferability of attacks on face recognition systems by passing our generated adversarial examples to the Azure Face API [14] and seeing whether the adversarial pairs are correctly classified by this unseen model.

References

[1] Matthew Cox. Army's next infantry weapon could have facial-recognition technology. *Military.com*, 2019. [Online; accessed 10-October-2019].

[2] Clarice Smith. Facial recognition enters into healthcare. *Journal of AHIMA*, 2018. [Online; accessed 10-October-2019].

[3] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[4] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 2016.

[5] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. *CVPR*, 2015.

[6] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[7] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 67–74. IEEE, 2018.

[8] Jonas Rauber, Wieland Brendel, and Matthias Bethge. Foolbox v0.8.0: A python toolbox to benchmark the robustness of machine learning models. *CoRR*, 2017.

[9] Bruno López Garcia. Crafting adversarial faces. *brunolopezgarcia.github.io*, 2018. [Online; accessed 7-November-2019].

[10] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *ICLR*, 2015.

[11] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2574–2582, 2016.

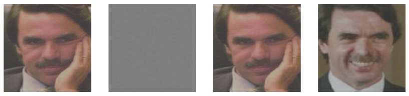
[12] Cihang Xie, Yuxin Wu, Laurens van der Maaten, Alan L Yuille, and Kaiming He. Feature denoising for improving adversarial robustness. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 501–509, 2019.

[13] Jiajun Lu, Theerasit Issaranon, and David Forsyth. Safetynet: Detecting and rejecting adversarial examples robustly. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 446–454, 2017.

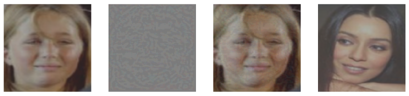
[14] Face api - face recognition services. *Microsoft Azure*, 2019. [Online; accessed 7-November-2019].

A Examples of attacks

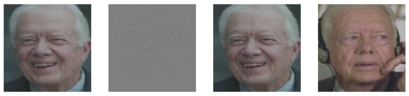
fgsm: Same Person -> Different
Embedding dist 0.594 -> 0.996
Norm change 0.544



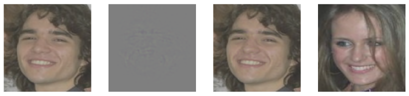
fgsm: Different Person -> Same
Embedding dist 2.345 -> 0.975
Norm change 4.079



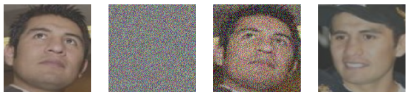
deep_fool: Same Person -> Different
Embedding dist 0.213 -> 0.991
Norm change 0.912



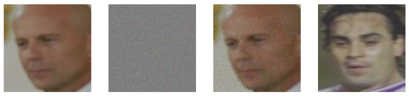
deep_fool: Different Person -> Same
Embedding dist 1.854 -> 0.990
Norm change 0.900



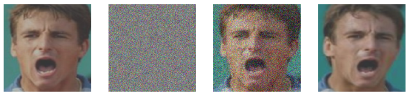
uniform_noise: Same Person -> Different
Embedding dist 0.555 -> 1.003
Norm change 38.168



uniform_noise: Different Person -> Same
Embedding dist 1.231 -> 0.986
Norm change 11.750



guassian_noise: Same Person -> Different
Embedding dist 0.357 -> 0.999
Norm change 28.380



guassian_noise: Different Person -> Same
Embedding dist 1.231 -> 0.964
Norm change 12.789