Stress Testing Facial Recognition with Adversarial Examples

Emily Strong
CSYE Spring 2018
Northeastern University
Boston, MA
strong.e@husky.neu.edu

Abstract – Convolutional neural networks designed for facial recognition tasks are susceptible to adversarial examples similar to other types of CNNs. Here I stress test a facial recognition model with two types of adversarial examples – images augmented to decrease their quality, and images modified to trick the network into identifying the subject as a target person – to gain insights into what forms of adversarial examples are most and least effective. My results suggest that it depends largely on the image being used, though the skin tone of the person pictured has an effect. My method of targeting labels has proved to be less effective than those that use gradient feedback from the CNN, but is effective for evading correct identification.

I. INTRODUCTION

With the latest state-of-the-art facial recognition algorithms such as Facebook's DeepFace (Taigman, Yang, Ranzato and Wolf 2014) and Google's FaceNet FaceNet (Schroff, Kalenichenko and Philbin 2015), the ability of these models to recognize faces has met or surpassed human abilities. However, as with other convolutional neural networks (CNNs) used in image classification problems, facial recognition models are susceptible to adversarial examples. Adversarial examples are inputs designed to trick the model. CNNs measure changes in the values of adjacent pixels to detect edges, so slight changes to individual pixels can move where an edge is detected and cumulatively these changes result in the CNN detecting the pattern of a different class even though a person looking at the image can still correctly identify it.

Facial recognition algorithms encode distances of morphological features rather than global patterns. However, these distances are relative to detected edges of facial features so adversarial examples can still be designed using methods that change where the edges are detected. Sharif, Bhagavatula, Bauer and Reiter (2016) identified that facial recognition algorithms can be tricked using texture perturbation in multiple forms including glasses that can be printed and worn in the real world.

This study took two different approaches to adversarial examples. In Part A, a variety of methods of augmenting images such as noise addition and inversion were tested against the OpenFace model (Amos, Ludwiczuk, and Satyanarayanan 2016) to determine how they affect the accuracy of the model.

In Part B, images were modified specifically to have them be misclassified as a target person. The images were first optimized though a genetic algorithm to decrease the difference between the source and the target and then were further modified through a brute-force black box attack until either the target was reached or the image was transformed to such an extent that a face could no longer be detected. In this set of trials the stress test was using images of people with increasingly disparate morphological differences and against two models, one trained only on the test subjects and one trained on the FaceScrub data set (Ng and Winkler 2014) which features 530 people. Salah, Alyüz and Akarun (2008) found that 3D scans of faces clustered based on morphological differences divide on race and gender. The test cases for morphological differences thus assume that race and gender are likely to correspond to greater morphological differences, examining each of these separately as well as combined.

My results indicate that what strategies can be used to create adversarial examples depend highly on the individual image, though there also appears to be a relationship between skin tone and how effective certain of the image augmentation methods are. My method of modifying images for targeting particular labels had mixed results and is not an optimal method. The genetic algorithm yielded an average improvement of 0.23 ± 0.15, though it failed to improve three of the images, including the two that had a similarity score less than 1. These improvements however did not generally translate to success in the adversarial targeting. Of the ten subject pairs, only four achieved their target labels against the limited data set with either image, with only one of those pairs having reciprocal success, and only one image achieved success against the full FaceScrub data set. Surprisingly this was in the category that should have been the most challenging – different race and gender. The image that was successful was produced directly by the genetic algorithm with no additional modification, suggesting that the genetic algorithm might have eventually produced successful images for the other test cases if more generations had been run. Overall, however, due to its random nature and inconsistent results the method of modifying images used here is not recommended for targeted adversarial examples.

II. METHODS

A. Data Set

FaceScrub (Ng and Winkler 2014) is an image data set with 106,863 labeled images of 530 celebrities (265 men, 265 women). Due to the copyrights of the images, the data set consists of a list of URLs from which those images can be downloaded. The URLs are four years old so upon scraping them I found only 65,305 of the images are still available with a range of 31 to 197 images of each person.

To have a test case of people with very similar faces, I supplemented the data set with approximately 300 images each of Natalie Portman and Keira Knightley obtained through web scraping similar to the rest of the FaceScrub data set. For the remaining test subjects, I chose people with more than 100 pictures in the available subset of FaceScrub who met the general requirements for the test cases. FaceScrub and the supplemental images were used for training the face recognition model, while images of the subjects from Wikimedia Commons were used for testing.

B. Face Recognition Pipeline

The OpenFace model (Amos, Ludwiczuk, and Satyanarayanan 2016) uses the dlib Face Detector API which converts an image to a histogram of oriented gradients (HOG) and then uses a support vector machine (SVM) classier to detect the pattern of a face. If a face is found, the model then aligns the face using the outer corners of the eyes and the tip of the nose and performs an affine transformation to place them at standard positions within a bounding box around the face. The image is cropped to the bounding box and resized to 96 x 96 pixels. The resulting image is then inputted to a convolutional neural network with 39 layers that embed the face morphology, outputting 128 measures of distances between features. An SVM is then trained on those 128 measurements for the subjects in the training set. The CNN was built with Torch in Lua and has a Python API. The model runs natively in a Docker environment, which I adapted to use with Jupyter Notebook.

C. Image Augmentation

For the image augmentation stress testing, I treated images of Colin Firth and George Lopez to the following tests:

1. Inversion: images underwent total inversion and inversion by channel.
2. Add: images had their RGB channels incremented by 10 until either a different label was returned or a face could no longer be detected. This was done as four tests with all channels incremented and then each incremented separately.
3. Multiply: images had their RGB values doubled. This was done as four tests with all channels doubled and then each doubled separately.
4. Subtract: images had their RGB channels decremented by 10 until either a different label was returned or a face could no longer be detected. This was done as four tests with all channels decremented and then each decremented separately.
5. Gaussian blur: a Gaussian blur with sigma 5 was applied to the images.
6. Gaussian noise: Gaussian noise with a mean of 0 and a sigma of $\sqrt{(255)}$ was added to the images.
7. Salt and pepper noise: Salt and pepper noise was added to the images using thresholds of .05 and .95 when randomly generating floating point numbers in the range [0, 1].
8. Contrast: Contrast correction was applied using a correction factor of: $(259 * (C + 255)) / (255 * (259 - C))$ where C is 200 for high contrast and -200 for low contrast. The contrast adjustment is Factor * (RGB -128) + 128, applied as a point transformation.

D. Genetic Algorithm

Sharif et al. (2016) used gradients returned by their CNN to generate modifications to the images that were then optimized through a particle swarm until the adversarial target label was achieved. Unfortunately, the OpenFace API does not provide a method for accessing gradients. It does however return a matrix of embeddings, and matrices of different images can be used to assess the facial similarity of their subjects. This is done by subtracting one matrix from the other and calculating the dot product of the difference as the sum of the squares. I thus generated random noise and used a simple genetic algorithm to select the noise additions that moved a source image closer to the target, using the dot product as a fitness score. Figure 1 shows the pseudocode for the genetic algorithm. Noise was restricted to a range of ± 10 of the initial pixel values to prevent the image from changing so much that a face would no longer be detected.

The creators of OpenFace in testing the similarity scores of faces found that the average threshold for determining whether an image is of the same person is 0.99 (Amos, 2015), however the similarity score for the test images of Natalie Portman and Keira Knightley is 0.53, so I used 0.25 as a cutoff since the classifier correctly identifies them even with such similarity.

```
fittest image = source image
minimum score = dot product of source - target
population = [fittest image, minimum score]

while minimum score > cutoff and count < limit:
    20 times:
        generate version of fittest image with random noise added
        fitness score = dot product of image - target
        add image, fitness score to population
    sort population on fitness score
    update fittest image, minimum score
    population = [fittest image, minimum score]
```

Figure 1: Genetic Algorithm Psuedocode

E. Brute Force Attack

After optimizing the images through the genetic algorithm, I then used a brute force attack to achieve the target labels. The pairings for the adversarial attacks are listed in Table 3. The method of attach was similar to the random noise generation used for creating children in the genetic algorithm, however here with each try only one image was generated and the image was not constrained by a clipping range to accelerate the process. Each attack was run with terminating conditions of the target reached or a face is no longer detected, with a limit of 300 tries. Each of the test cases was run twice switching the source and target using a model trained only on the test subjects. If it successfully tricked the limited model it was then tested against a model trained on the full FaceScrub data set. For the non-human face and no face tasks, I used a modified version of the brute force attack in which the target was to trick the face detector model.

III. RESULTS

A. Part A – Image Augmentation

The results of the image augmentation are in Table 1 with example images generated shown in Figure 2. The difference in results between image 1 and image 2 demonstrate that the efficacy of each strategy is highly dependent on the particular image used, though skin tone likely also affects the strategies related to color modifications. For example, each image had a face detection error on a test that the other one did not, and similarly some color changes caused one image to be incorrectly labeled but not the other.

Table 1. Effects of Image Augmentation on Face Classification

Modification	**Image 1: Colin Firth**		**Image 2: George Lopez**	
	Final parameter	***Result***	***Final parameter***	***Result***
Inversion				
Complete		No face detected		Hayden Christensen
Red		Jonathan Sadowski		Peggy Lipton
Green		Ben Kingsley		George Clooney
Blue		Colin Firth		Burt Reynolds
Add				
Total	+60	Bruce Greenwood	+60	Tom Hanks
Red	+60	Jack Nicholson	+60	Tom Hanks
Green	+70	Sean Bean	+70	Robert Downey Jr
Blue	+100	Colin Firth	+100	George Lopez
Multiply	X2		X2	
Total		No face detected		Dustin Hoffman
Red		Colin Firth		George Lopez
Green		Colin Firth		George Lopez
Blue		Colin Firth		George Lopez
Subtract				
Total	-30	Alan Alda	-30	Josh Brolin
Red	-80	Ben Kingsley	-100	Robert Downey Jr
Green	-50	Neal McDonough	-40	Mel Gibson
Blue	-90	Aaron Eckhart	-60	No face detected
Gaussian Blur		Colin Firth		George Lopez
Gaussian Noise		Colin Firth		George Lopez
Salt and Pepper Noise		Jack Nicholson		Jackie Chan
Contrast				
High	+200	Colin Firth	+200	Kal Penn
Low	-200	No face detected	-200	No face detected

Figure 2: Example Outputs for George Lopez. A. Total Inversion. B. Green Inversion. C. Multiply. D. Add Blue. E. Subtract Red. F. Gaussian Blur. G. Gaussian Noise. H. Salt and Pepper Noise. I. High Contrast.

B. Part B – Adversarial Targeting

The results of the genetic algorithm are shown in Table 2 and the results of the brute force attacks are shown in Table 3. The genetic algorithm failed to generate a single image that improved the similarity of the images of Natalie Portman and Keira Knightley despite 10,000 random modifications being generated over the course of the two runs. This also happened with Tatyana M. Ali targeting Ken Watanabe. Furthermore, there was an issue with the constraint being insufficient for Samuel L. Jackson – every time his image was used after 1-3 modifications his face could no longer be detected, likely due to a combination of his dark skin tone and his hat and glasses in the picture. For the other test subjects, however, it did achieve improvement beginning in the first or second generation though none achieved the 0.26 cutoff. With one image, Kristin Chenoweth targeting George Lopez, the genetic algorithm successfully produced an image that achieved the target label. The average improvement for the runs that completed all 250 generations was 0.23 ± 0.15.

The method of adding random noise appears to be an effective method for avoiding correct labeling as most test cases have achieved a different label within a few tries, however it does not successfully achieve target labels for all test cases with the limited data set and only one succeed when tested against the full FaceScrub data set. In addition, the test case of Colin Firth targeting Matthew Perry did succeed on the first run, but on a subsequent re-run it failed. Table 4 summarizes the fitness score trends across the test cases and the results against the limited data set. Interestingly, this same method was successful for causing a bicycle to be detected as a face, but the test case with a non-human face did not succeed. Figures 3 and 4 show example outputs and demonstrate the amount of noise added to each image to achieve these results.

Table 2: Genetic Algorithm Transformations

			Person 1 → Person 2				**Person 2 → Person 1**			
Person 1	**Person 2**	**Initial D**	**Final D**	**Generations**	**1st Change**	**Fittest Gen.**	**Final D**	**Generations**	**1st Change**	**Fittest Gen.**
Keira Knightley	Natalie Portman	0.53	0.53	250	N/A	N/A	0.53	250	N/A	N/A
Kristin Chenoweth	Keira Knightley	1.83	1.48	250	0	195	1.49	250	0	110
Colin Firth	Matthew Perry	1.33	0.96	250	0	169	1.22	250	0	178
Kristin Chenoweth	Colin Firth	2.07	1.73	250	0	169	1.57	250	0	8
Tatyana M Ali	Samuel L Jackson	1.75	1.60	250	18	116	N/A	1	0	N/A
Colin Firth	George Lopez	1.49	1.30	250	1	86	1.06	250	0	248
Samuel L Jackson	Ken Watanabe	2.24	N/A	1	0	N/A	1.92	250	0	185
Tatyana M Ali	Kristin Chenoweth	1.89	1.57	250	0	44	1.71	250	1	150
Kristin Chenoweth	George Lopez	1.82	1.50	25	0	30	1.68	250	0	177
Tatyana M Ali	Ken Watanabe	1.42	1.42	250	N/A	N/A	1.33	250	0	193

Table 3: Adversarial Classification

Number of tries to get to a different label, total number of tries and whether the target was ever reached. An asterisk (*) indicates that the image was modified to such an extent that a face could no longer be detected. A dash (-) indicates that no different label was achieved. N/A indicates full model was not tested due to failure with limited.

Test Case	Source	Target	Model	Tries to Different	Total Tries	Succeeded
People with similar faces	Keira Knightley	Natalie Portman	Limited	8	8	Yes
			Full	3	291*	No
	Natalie Portman	Keira Knightley	Limited	9	11	Yes
			Full	3	20*	No
People with same race, same gender	Kristin Chenoweth	Keira Knightley	Limited	-	300	No
			Full	-	300	No
	Keira Knightley	Kristin Chenoweth	Limited	33	67	Yes
			Full	0	300	No
	Colin Firth	Matthew Perry	Limited	212	212	Yes
			Full	56	300	No
	Matthew Perry	Colin Firth	Limited	-	300	No
			Full	64	69*	No
People with different genders	Kristin Chenoweth	Colin Firth	Limited	-	300	No
			Full	N/A	N/A	
	Colin Firth	Kristin Chenoweth	Limited	-	300	No
			Full	N/A	N/A	
	Tatyana M Ali	Samuel L Jackson	Limited	-	300	No
			Full	N/A	N/A	
	Samuel L Jackson	Tatyana M Ali	Limited	-	2*	No
			Full	N/A	N/A	
People of different races	Colin Firth	George Lopez	Limited	167	300	No
			Full	N/A	N/A	
	George Lopez	Colin Firth	Limited	-	300	No
			Full	N/A	N/A	
	Samuel L Jackson	Ken Watanabe	Limited	-	3*	No
			Full	N/A	N/A	
	Ken Watanabe	Samuel L Jackson	Limited	-	300	No
			Full	N/A	N/A	
	Kristin Chenoweth	Tatyana M Ali	Limited	162	268*	No
			Full	N/A	N/A	
	Tatyana M Ali	Kristin Chenoweth	Limited	-	300	No
			Full	N/A	N/A	
People of different races and genders	Kristin Chenoweth	George Lopez	Limited	0	0	Yes
			Full	0	0	Yes
	George Lopez	Kristin Chenoweth	Limited	-	300	No
			Full	N/A	N/A	
	Tatyana M Ali	Ken Watanabe	Limited	-	300	No
			Full	N/A	N/A	
	Ken Watanabe	Tatyana M Ali	Limited	-	300	No
			Full	N/A	N/A	
Non-Human Face	cat	human face	Detector		200	No
No Face	bicycle	human face	Detector		15	Yes

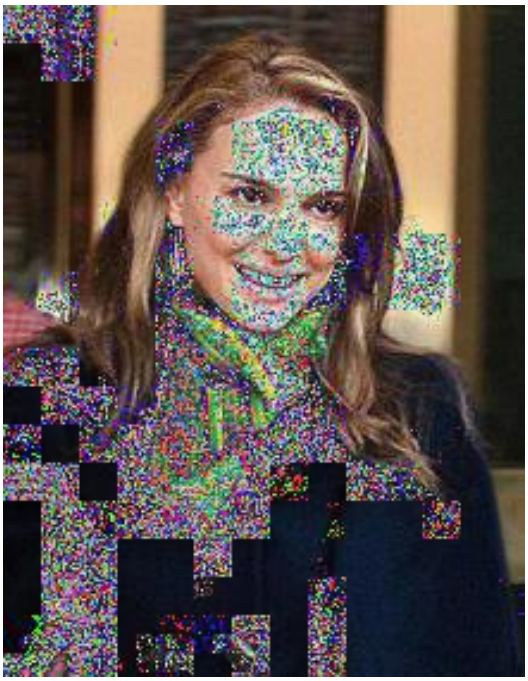
A  B 

Figure 3: Example Outputs of Genetic Algorithm and Adversarial Classification. A. Colin Firth optimized to match George Lopez. B. Natalie Portman successfully brute force modified to match Keira Knightley.

Figure 4: Image of bicycle successfully detected as a face.

Table 5: Summary of Test Results with Limited Data Set

Test Case	Mean Difference	Mean Final Fitness Score	Success	Failure
1: Similar	0.53	0.53	2	0
2: Dissimilar	1.74	1.29	2	2
3: Same Gender	1.91	1.57	0	4
4: Same Race	1.87	1.67	0	6
5: Different Race and Gender	1.62	1.48	1	3
6: Non-Human Face	-	-	0	1
7: No Face	-	-	1	0

IV. DISCUSSION

This study examined a variety of methods of generating adversarial examples, from simple image augmentation to using a genetic algorithm to optimize the differences in embedding measurements. Of the various methods tested, none would pass the litmus test of being undetectable to a person. Salt and pepper noise comes the closest, but an expert on working with convolutional networks would likely be suspicious.

Collectively the results highlight the fact that adversarial examples work by changing where a convolutional neural network detects edges, with the skin tone of the person pictured appearing to be related to which strategies are effective and at what parameter level. This finding is particularly exemplified by the test cases with Samuel L. Jackson. After 1-3 iterations of adding random noise, his face could no longer be detected even with the genetic algorithm where the modifications were constrained.

One of the most interesting results is that my hypothesis that face morphology differences would increase across the test cases was incorrect. Indeed, prior to being run through the genetic algorithm, the images of people of different races but the same gender had the greatest morphological difference as measured by the face embeddings. I was however correct that similar morphology would correspond to better performance. Of the test cases that succeeded against the limited data set, the average final fitness score was 1.00 versus 1.57 for those that failed.

The most significant factor in the results of the targeted examples appears to be the random nature of the modification used in the targeting. Only during the running of the genetic algorithm was there feedback to optimize where the noise was added, but even with feedback only a single image produced by that method was successful without additional modification. This image was also the only one that succeeded against the full FaceScrub data set. By comparison, the two images with the greatest similarity were not improved at all by the genetic algorithm. The brute force method was however successful for them, suggesting that of the 10,000 children generated during the genetic algorithm, some proportion of them would have been able to trick the face classifier if that had been added as a terminating condition. Similarly, with the people with dissimilar faces, with one pair after the genetic algorithm there was only a difference of 0.01 between their fitness scores but one image was successful and the other was not, and with the other pair one image was successful on the first try but not on a subsequent re-run. This method of modifying images for targeted attacks is thus not recommended as a general strategy. In a

future study it would appropriate to use a different facial recognition model that provides gradient feedback so that adversarial examples could be generated in a controlled manner.

REFERENCES

Amos, Brandon. "Demo 2: Comparing two images." *OpenFace.* CMUSatyaLab, 13 October 2015. Web. 19 April 2018. http://cmusatyalab.github.io/openface/demo-2-comparison/

Amos, Brandon, Bartosz Ludwiczuk, and Mahadev Satyanarayanan. "Openface: A general-purpose face recognition library with mobile applications." *CMU School of Computer Science* (2016).

Ng, Hong-Wei, and Stefan Winkler. "A data-driven approach to cleaning large face datasets." *Image Processing (ICIP), 2014 IEEE International Conference on*. IEEE, 2014.

Salah, Albert A., Nese Alyüz, and Lale Akarun. "Registration of three-dimensional face scans with average face models." *Journal of Electronic Imaging* 17.1 (2008): 011006.

Schroff, Florian, Dmitry Kalenichenko, and James Philbin. "Facenet: A unified embedding for face recognition and clustering." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.

Sharif, Mahmood, Sruti Bhagavatula, Lujo Bauer, and Michael K. Reiter. "Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition." *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016.

Taigman, Yaniv, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. "Deepface: Closing the gap to human-level performance in face verification." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014.