Applying Adversarial Attack to

Multi-frame Anti-Spoofing Model

<https://github.com/maomao0819/Applying-Adversarial-Attack-to-Multi-frame-Anti-Spoofing-Model>

Members: 黃秉茂、朱偉綸

Guiding Mentor: 黃思皓教授

Department of Information Management and Finance

National Yang Ming Chiao Tung University

July, 2021

1. Abstract

Face anti-spoofing is critical to the security of face recognition systems. Depth supervised learning has been proven as one of the most effective methods for face anti-spoofing. For security, utilizing a multi-frame model instead of single frame detection, and considering more frames can reduce the probability to be beaten. Since the recognition with multiple frames can not only check the patterns in the frame but also ensure the time relative relationship between different frames, the multi-frame anti-spoofing model seems unbeatable. In this paper, **we study the** **vulnerability of anti-spoofing methods against adversarial perturbations**, **and try to defeat the multi-frame model**. To conduct the experiment, we adapt the basic adversarial attack method to attack on different dataset. Furthermore, we introduce the concept of image kernel filter to attempt increasing attack success rate and optimizing image quality.

2. Introduction

Face recognition technology has become mature, the detection can precisely tell who you are. Nonetheless, some people can use your photo to pretend to be you and use your identity to do something harmful to you. Thus, the face anti-spoofing technology is not less important than recognition. To optimize the detection system, many researchers consider more filters as the one of the best solutions. Nowadays, more and more face anti-spoofing methods on multiple frames have been designed, and there are many attack algorithms on images. However, there is no paper about how the performance is about attacking on the multi-frame face anti-spoofing model. We wonder if the multi-frame model is vulnerable to adversarial attack, and other researchers may improve the model by erasing the weakness. And to increase the performance of attacking, **we introduce the concept of the kernel image filters and SSIM to improve the image quality and the attack success rate** **on multi-frame face anti-spoofing model**.

3. Related Work

We will firstly train a model which can precisely classify the spoofing images and real images with **multiple frames**. We adapted the detection model [1] for our work, and the model considers 1) detailed discriminative clues between living and spoofing faces may be discarded through stacked vanilla convolutions, and 2) the dynamics of 3D moving faces provide important clues in detecting the spoofing faces. Furthermore, since the model is combined of Binary supervised Methods, Depth supervised Methods, and Temporal-based Methods, we believe it is powerful enough for our attacking experiment. And then, we try to beat the model with the basic adversarial attack method such as FGSM and iterative FGSM. They are both attack methods which consider the gradient of the error.

Several attack methods have been proposed to generate adversarial examples. Here we provide a brief introduction.

F**ast Gradient Sign Method (FGSM)**. FGSM [2] (Goodfellow et al., 2014) generates an adversarial example xadv by maximizing the loss function J(xadv, ytrue) with one-step update as:

xadv = x + eps \* sign(∇xJ(x, ytrue)),

where sign(·) function restricts the perturbation in the L∞ norm bound.

[3] **Iterative Fast Gradient Sign Method (I-FGSM)**. Kurakin et al. (2016) extend FGSM to an iterative version by applying FGSM with a very small step size α:

x0 = x, xt+1adv = Clipxeps{xtadv + α · sign(∇xJ(xtadv , ytrue))},

where Clipxeps (·) function restricts generated adversarial examples to be within the eps-ball of x.

4. Methodology:

In the beginning, we train a model which can precisely tell the spoofing images and real images with multi-frames. We adapted the detection model [1] for our work and we believe it is powerful enough for our attacking experiment. And then, we try to use adversarial attack like FGSM and iFGSM with kernel image filters to beat the model.

What with the attack success rate and what with the image quality, we introduce image kernel filters. **Since the model is temporal-based method, the filter concept may increase the fool rate.** To be more specific, the model considers the relationship between 5 frames, so, it is not wise to add the perturbation without considering the temporal relationship. Unfortunately, the adversarial attack may break the relationship severely. Since the attack will lead the pixel on different frames to be so much different, it may cause the image with the noise to remain being spoofing. Remember, the goal of the attack is to let the model misclassify the images. To alleviate the impact on temporality, we use the **filter to reduce the difference between the pixels on different frames**. Consequently, We try the basic and common filter on our experiment. Finally, we adopt the average filter [4] and Gaussian filter to improve the image quality and attack success rate. **Above all, we may be the first paper to propose the filter to increase the image quality and the attack success rate on multi-frame face anti-spoofing.**

Because we hope the images with noise are natural, and people shouldn’t get weird from the images with perturbation, we pursue images to be high quality. **For measuring the quality, we introduce the SSIM index** to demonstrate whether the image is good or bad. Besides, we attempt to add multiple filters and hope it will turn out to be a better result of attack success rate and image quality.

1. Experiment

5.1. Attacked target

Since our goal is to fool the detecting recognition, we only attack the image whose label is spoofing and the model also correctly classifies it as a spoofing image.

5.2. Dataset

Four datasets - OULU-NPU, MSU-MFSD, CASIA-FASD, Replay-Attack are used in our experiment. OULU-NPU is a high-resolution database,consisting of 4950 real access and spoofing videos and containing four protocols to validate the generalization of models. MSU-MFSD contains 280 video recordings of genuine and attack faces. CASIA-FASD and Replay-Attack both contain low-resolution videos.

5.3. Performance Metrics

ASR: Attack Success Rate, which is the ratio represented by the spoofing images misclassified to be the real by the model.

SSIM: Structural Similarity Index Measure, which is the index that elaborates how two images are similar by considering structure, intensity, and contrast.

5.4. Comparison

5.4.1. model

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dataset | Accuracy | apcer | bpcer | acer |
| OULU-1 | 0.9625 | 0.042 | 0.0363 | 0.0392 |
| OULU-2 | 0.9430 | 0.0279 | 0.0715 | 0.04973 |
| OULU-3 | 0.9274 | 0.0826 | 0.0339 | 0.0583 |
| OULU-4 | 0.9815 | 0 | 0.0526 | 0.0263 |
| CASIA | 0.9175 | 0.3016 | 0.0185 | 0.1601 |
| Replay Attack | 0.9479 | 0.1625 | 0.03 | 0.09625 |
| MSU | 0.9329 | 0.0727 | 0.0513 | 0.062 |

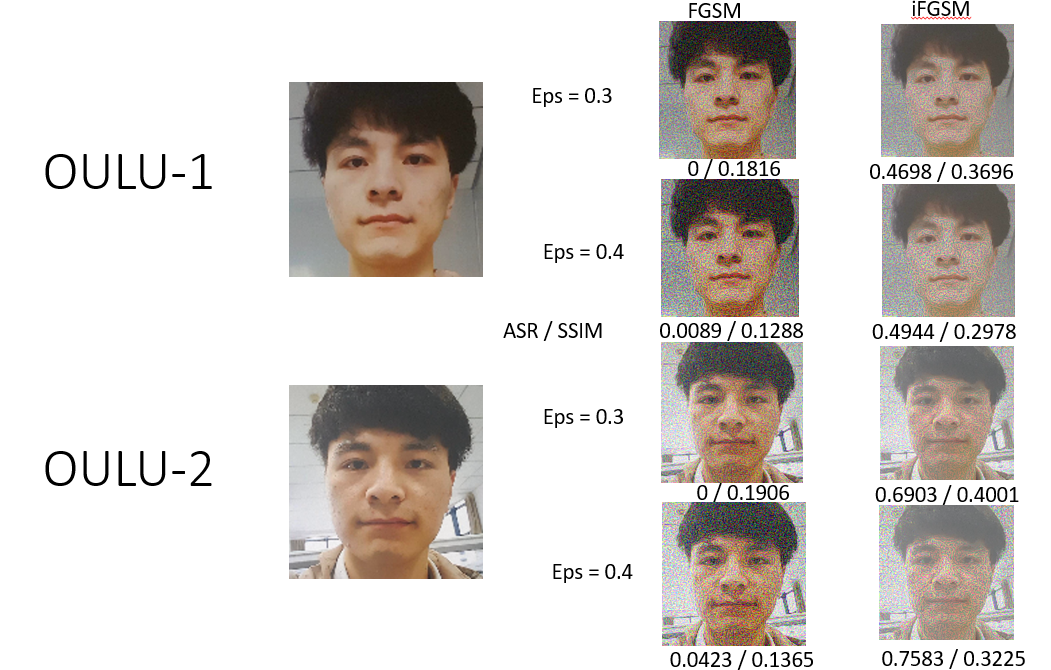
The model is strong enough to classify real or spoofing,and it is worth being attacked for our work.

5.4.2. FGSM vs iFGSM

The table show the attack success rate (%) with different epsilon.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset | attack method | 0.3 | 0.4 | 0.5 | 0.6 |
| OULU-1 | FGSM | 0 | 0.89 | 7.61 | 14.09 |
| **iFGSM** | **46.98** | **49.44** | **51.68** | **53.69** |
| OULU-2 | FGSM | 0 | 4.23 | 0 | 32.78 |
| **iFGSM** | **69.03** | **75.83** | **80.82** | **82.78** |
| OULU-3 | FGSM | 0 | 0.47 | 0.47 | 1.89 |
| **iFGSM** | **60.19** | **65.4** | **66.35** | **70.14** |
| OULU-4 | FGSM | 0 | 5.71 | 11.42 | 11.42 |
| **iFGSM** | **51.43** | **57.14** | **57.14** | **57.14** |
| CASIA | **FGSM** | **6.6** | **45.75** | **86.79** | **93.4** |
| iFGSM | 4.25 | 12.26 | 29.72 | 54.72 |
| RA | **FGSM** | **17.22** | **22.22** | **29.17** | **30.28** |
| iFGSM | 15.56 | 16.39 | 28.06 | 25.56 |
| MSU | **FGSM** | **36.27** | **38.25** | **36.27** | **38.25** |
| iFGSM | 28.43 | 29.41 | 33.33 | 47.06 |

The bottom-left corner value is the attack success rate / SSIM.



For the dataset OULU-1, the result of the iFGSM is better than FGSM. However, for the others datasets, the preformance of the iFGSM is worse than FGSM.

5.4.3. the filter, the dimension ,and the application place

We attempt to apply the concept of the image filter and try to find where is the best place to apply the kernel filter and which filter is hitting the point. To do the appropriate comparison, we have made some experiment. The table show that which filter is applied, applied on 2-dimension or 3-dimention to the perturbation and where does the filter apply. For the filter type, we only apply Gaussian filter and Uniform filter. Furthermore,‘noise’means we apply the filter to the perturbation,‘image’means we only apply the filter to the image after adding perturbation, and ‘all’means we apply the filter to the perturbation and the image after adding perturbation.

The table will show the attack success rate (%) with the result of the different eps and auxiliary filters based on the same dataset and the same adversarial attack method.

The dataset is **OULU-1**, and the attack method is **FGSM**.

|  |  |  |
| --- | --- | --- |
| **Filter \ Epsilon** | **0.3** | **0.4** |
| No filter | 0 | 0.89 |
| 3D Uniform noise | 31.10 | 11.86 |
| 3D Uniform image | 1.34 | 0.44 |
| 3D Uniform all | 1.34 | 2.91 |
| **3D Gaussian(sigma=1) noise** | **60.85** | **56.38** |
| 3D Gaussian(sigma=1) image | 12.08 | 20.36 |
| 3D Gaussian(sigma=1) all | 8.50 | 9.17 |
| 3D Gaussian(sigma=0.5) noise | 0 | 0 |
| 3D Gaussian(sigma=2) noise | 37.81 | 41.16 |
| 3D Gaussian(sigma=2) image | 12.75 | 16.33 |
| 2D Uniform noise | 0.22 | 0 |
| 2D Uniform image | 0 | 0 |
| 2D Uniform all | 0.45 | 0 |
| 2D Gaussian(sigma=1) noise | 0.22 | 0 |
| 2D Gaussian(sigma=1) image | 0 | 0 |
| 2D Gaussian(sigma=1) all | 1.34 | 0.45 |

The dataset is **OULU-1**, and the attack method is **iFGSM**.

|  |  |  |
| --- | --- | --- |
| **Filter \ Epsilon** | **0.3** | **0.4** |
| **No filter** | **46.98** | **49.44** |
| 3D Uniform noise | 6.26 | 11.86 |
| 3D Uniform image | 0 | 0 |
| 3D Uniform all | 2.91 | 2.91 |
| 3D Gaussian(sigma=1) noise | 6.26 | 13.42 |
| 3D Gaussian(sigma=1) image | 2.01 | 2.24 |
| 3D Gaussian(sigma=1) all | 2.91 | 4.03 |
| 2D Uniform noise | 16.78 | 22.37 |
| 2D Uniform image | 11.19 | 18.12 |
| 2D Uniform all | 4.25 | 17.80 |
| 2D Gaussian(sigma=1) noise | 11.41 | 17.67 |
| 2D Gaussian(sigma=1) image | 4.7 | 16.55 |
| 2D Gaussian(sigma=1) all | 0.89 | 8.05 |

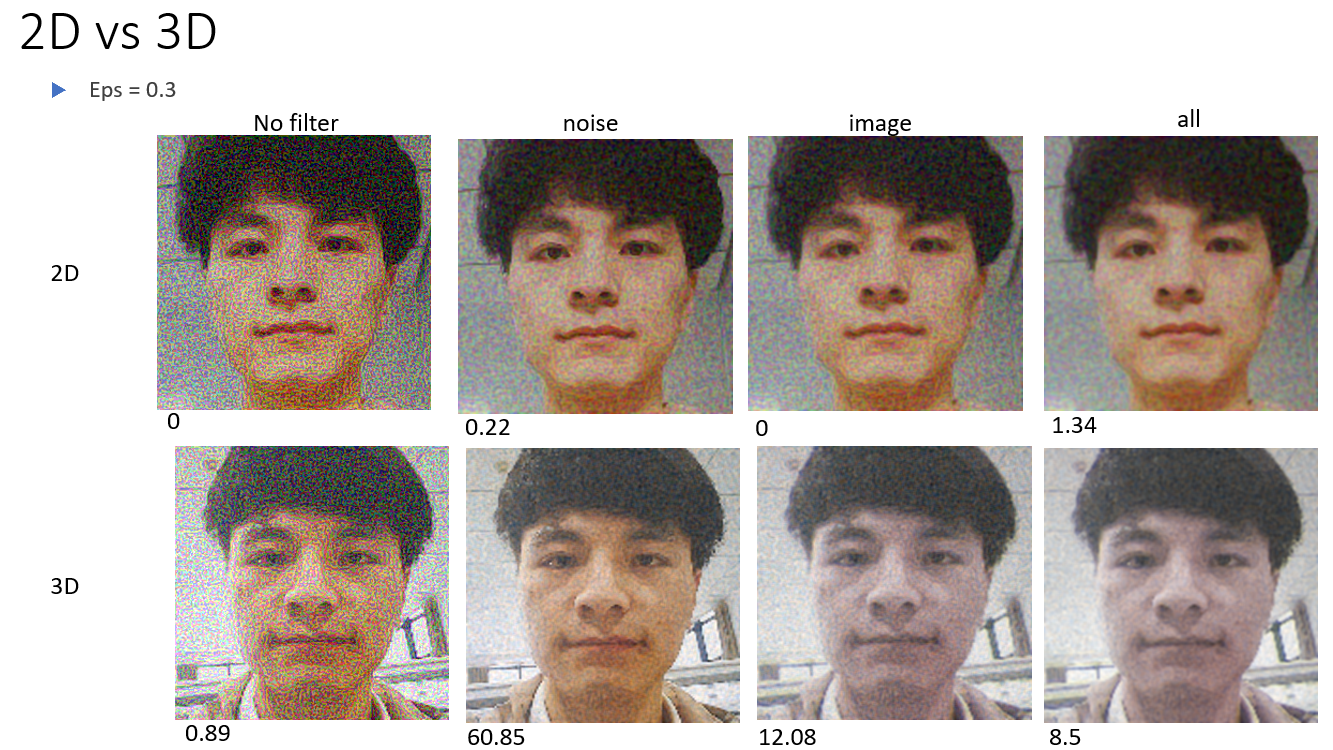
The dataset is **OULU-2**, and the attack method is **FGSM**.

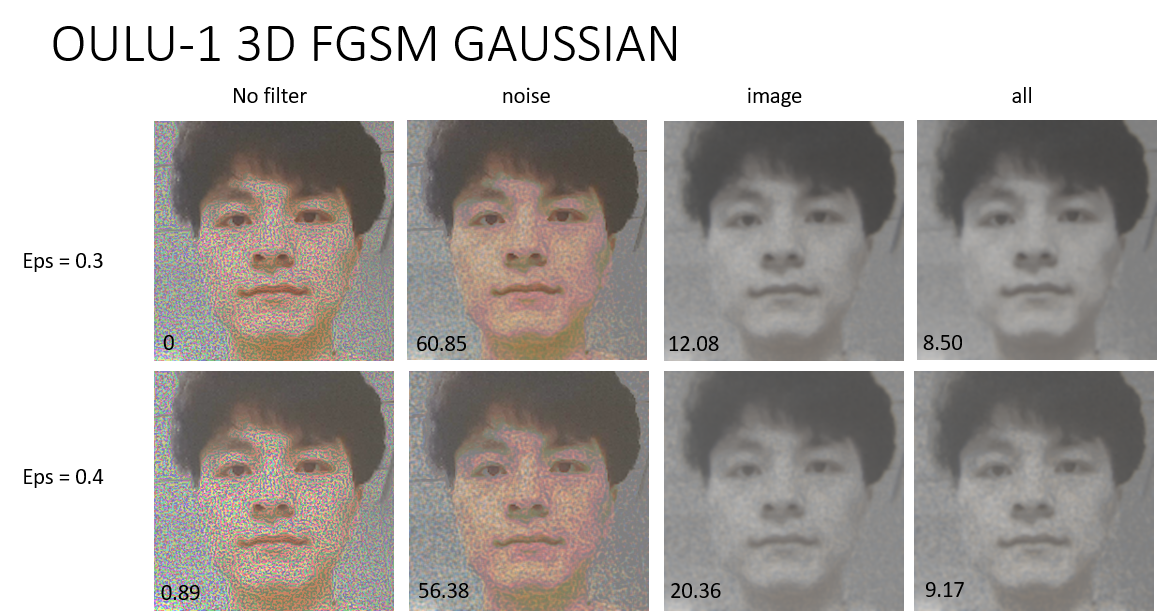
|  |  |  |
| --- | --- | --- |
| Filter \ Epsilon | 0.3 | 0.4 |
| No filter | 0 | 4.23 |
| 3D Uniform noise | 11.03 | 4.83 |
| 3D Uniform image | 4.23 | 4.08 |
| 3D Uniform all | 1.21 | 4.83 |
| **3D Gaussian(sigma=1) noise** | **9.97** | **19.94** |
| 3D Gaussian(sigma=1) image | 2.57 | 6.34 |
| 3D Gaussian(sigma=1) all | 3.32 | 7.85 |
| 2D Uniform noise | 0 | 0 |
| 2D Uniform image | 0.15 | 0.45 |
| 2D Uniform all | 0.15 | 0.15 |
| 2D Gaussian(sigma=1) noise | 0 | 0 |
| 2D Gaussian(sigma=1) image | 0.15 | 0.3 |
| 2D Gaussian(sigma=1) all | 0.15 | 0 |

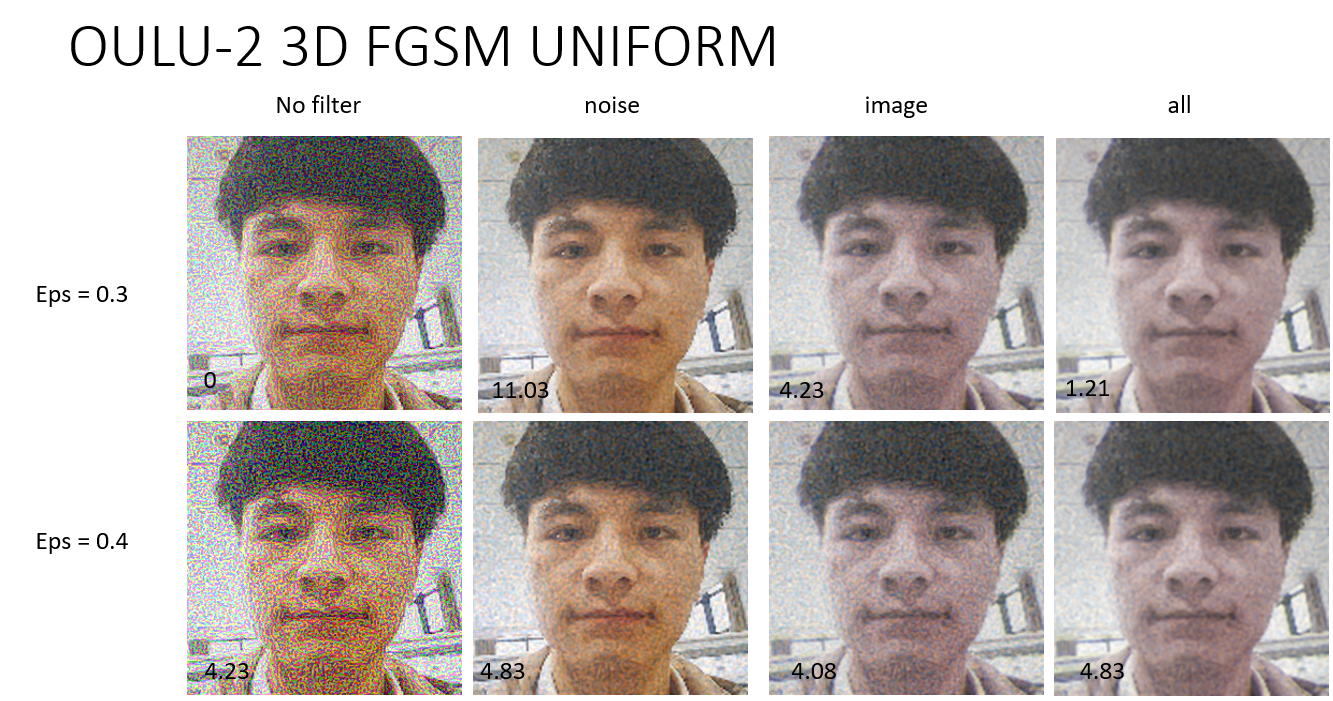
The dataset is **OULU-1**, and the attack method is **iFGSM**.

|  |  |  |
| --- | --- | --- |
| Filter\Epsilon | 0.3 | 0.4 |
| **No filter** | **69.03** | **75.83** |
| 3D Uniform noise | 14.65 | 21.60 |
| 3D Uniform image | 8.61 | 12.81 |
| 3D Uniform all | 12.24 | 18.43 |
| 3D Gaussian(sigma=1) noise | 14.80 | 21.60 |
| 3D Gaussian(sigma=1) image | 18.28 | 21.60 |
| 3D Gaussian(sigma=1) all | 19.64 | 30.06 |
| 2D Uniform noise | 25.53 | 40.48 |
| 2D Uniform image | 24.17 | 31.18 |
| 2D Uniform all | 17.22 | 29.61 |
| 2D Gaussian(sigma=1) noise | 20.24 | 31.27 |
| 2D Gaussian(sigma=1) image | 16.62 | 29.15 |
| 2D Gaussian(sigma=1) all | 6.95 | 23.26 |

The bottom-left corner value is the attack success rate.







The result on other dataset is close. In conclusion, **For FGSM, the fittest method is applying 3D Gaussian filter to the perturbation of FGSM**, and we can significantly fool the multi-frame anti-spoofing model. However, **for iFGSM, the auxiliary filters doesn’t brings any advantage** to the attack, and they do nothing but reduce the attack success rate. Thus, the auxiliary image filters only work on FGSM.

We also conduct other experiments with PIL smooth and smooth\_more filters. Unfortunately, the performance of those isn’t good enough.

* + 1. SSIM, Sigma and Multiple filter on 3D Gaussian and FGSM

Since the filter doesn’t work on iFGSM and the Gaussian filter is better than the uniform filter, we only apply 3D Gaussian filter to the perturbation of FGSM to the rest of the experiment. For the sake of not changing the image so severely, we introduce the SSIM to indicate the image texture. Moreover, we conduct more experiment on the filter. We test different parameters (sigma) of the Gaussian filter and try to apply more layer of the filter.

The table show the attack success rate (%) and the SSIM with the result of the different eps, sigma, the number of the layers.

The dataset is **OULU-1**, and the attack method is **FGSM**.

|  |  |  |
| --- | --- | --- |
|  | **0.3** | **0.4** |
| Sigma | ASR | SSIM | ASR | SSIM |
| no | 0 | 0.1816 | 0.89 | 0.1288 |
| 0.5 | 0 | 0.8660 | 0 | 0.8011 |
| 0.8 | 38.93 | 0.6526 | 15.88 | 0.5500 |
| 0.8 \* 2 | 37.88 | 0.7195 | 62.64 | 0.7692 |
| 1 | 60.85 | 0.7833 | 56.38 | 0.6961 |
| 1 \* 2 | 59.28 | 0.9179 | 59.51 | 0.8739 |
| **1.2** | **62.64 | 0.8660** | **62.86 | 0.8011** |
| 1.4 | 60.18 | 0.9153 | 59.96 | 0.8701 |
| 2 | 37.81 | 0.9717 | 41.70 | 0.9566 |

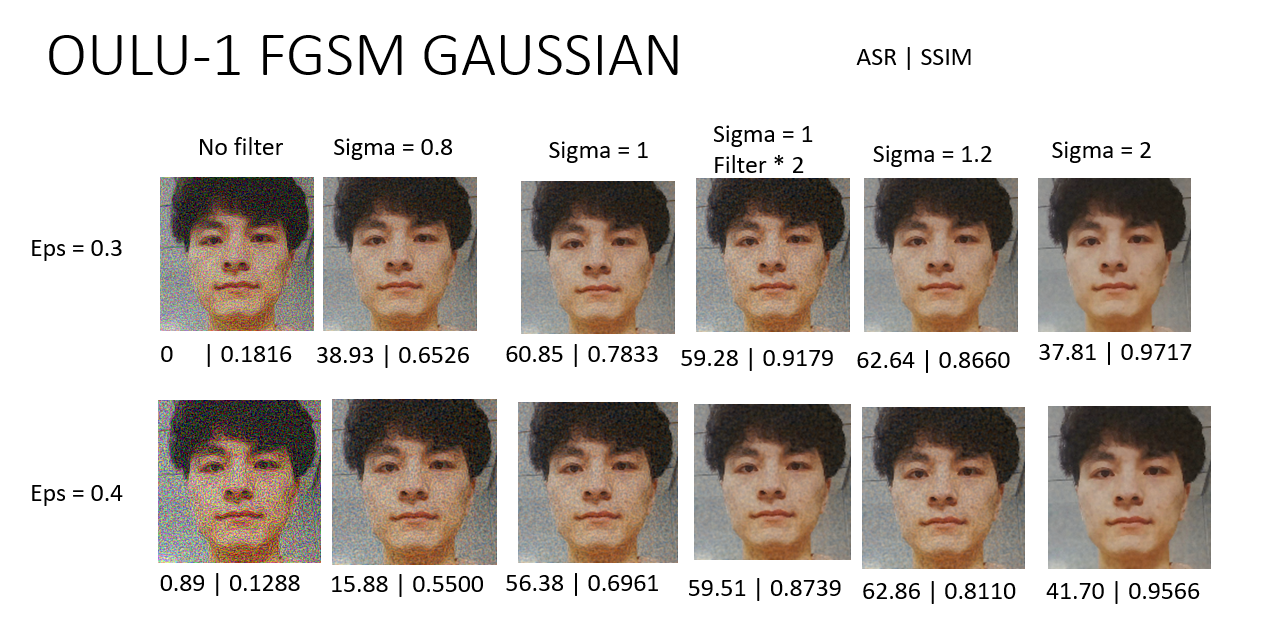
The dataset is **OULU-2**, and the attack method is **FGSM**.

|  |  |  |
| --- | --- | --- |
|  | **0.3** | **0.4** |
| Sigma | ASR | SSIM | ASR | SSIM |
| no | 0 | 0.1906 | 4.23 | 0.1365 |
| 0.8 | 2.27 | 0.6630 | 2.11 | 0.5615 |
| 0.8 \* 2 | 0 | 0.9658 | 0 | 0.9460 |
| **1** | **2.57 | 0.9541** | **6.34 | 0.9267** |
| 1 \* 2 | 0 | 0.9803 | 0 | 0.9691 |
| 1.2 | 0.15 | 0.8696 | 2.11 | 0.8067 |
| 1.2 \* 2 | 0 | 0.9873 | 0 | 0.9804 |
| 2 | 0 | 0.9716 | 0 | 0.9567 |

The sigma is fixed on 1 and the bottom-left corner value is the attack success rate / SSIM, and it tell the different images of different numbers of the Gaussian layers.



The bottom-left corner value is the attack success rate | SSIM.



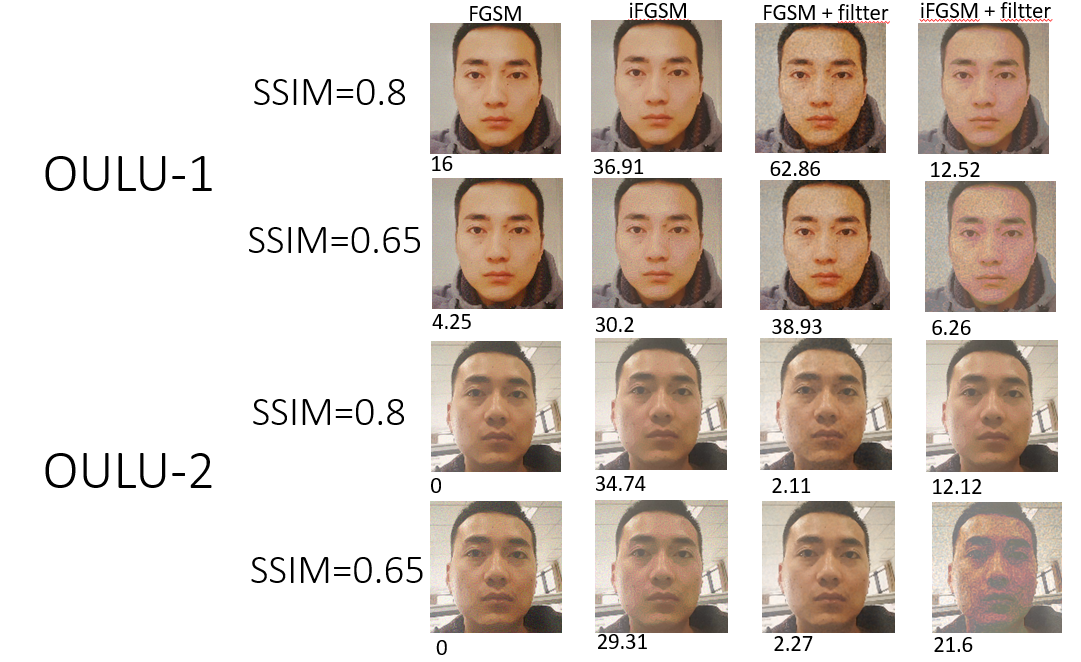
In conclusion, **the most proper sigma is around 1**. In addition, although 2 layers of the filter can increase the image quality, it losts lots of attack success rate. After considering the trade-off, we regard the multiple layer method as a not good Auxiliary tools.

5.4.5. Comprehension Comparison

The ASR of the nearly image quality on two protocols of OULU-NPU

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset | SSIM | FGSM | iFGSM | FGSM + filter | iFGSM + filter |
| OULU-1 | SSIM=0.8 | 16 | 36.91 | **62.86** | 12.52 |
| OULU-1 | SSIM=0.65 | 4.25 | 30.2 | **38.93** | 6.26 |
| OULU-2 | SSIM=0.8 | 0 | **34.74** | 2.11 | 12.12 |
| OULU-2 | SSIM=0.65 | 0 | **29.31** | 2.27 | 21.6 |

The bottom-left corner value is the attack success rate.



**For dataset OULU-1, the performance of FGSM + filter is the best, and for dataset OULU-2, the result of iFGSM is most significant** among all. Furthermore, **filter can improve FGSM performance**.

1. Demo System:

Step 1. Film a video with cellphone(about 1 min)

Step 2. Refilm the video with the camera of laptop

Step 3. Cut the video in step 2. into frames

Step 4. Crop face to every frames

Step 5. Generate depth map

Step 6. Go through our multi-frame model to check whether the input frames are spoofing to model

Step 7. Attack by FGSM (+filter) and Iterative FGSM (+filter)

1. Conclusion:

In conclusion, it is more difficult to fool the multi-frame model than the single frame model. If we follow the previous Fast Gradient Sign Method, although we can make each frame closer to the real image respectively, it may cause the difference between frames to become larger, and cause the model to still judge it as a spoof image. In our experiment, we proposed a novel method. We add a smoothing filter to the perturbation generated by Fast Gradient Sign Method, which can make the mask we want to add on our frame become more uniform. This allows us to reduce the difference between frames as much as possible while bringing them closer to the real image. Finally, making it easier for the model to judge it as a real image. Extensive experiments demonstrate the superiority of our method.

And, the followings show the brief conclusion of the experiment result, and the attack is on multi-frame model.

* **filter will enhance the effect of FGSM.**
* **filter will worsen the performance of iFGSM**
* **Filter Performance：Gaussian > Uniform**
* **SSIM: Gaussian > Uniform > no filter**
* **Multiple filter increase image quality**
* **Best filter’s sigma is around 1**
* **The best number of the layers is only 1.**
* **Trade-off between ASR and SSIM is important**
* **Best attack: iFGSM or FGSM with Filter**

8. Reference:

[1] WANG, Zezheng, et al. Deep spatial gradient and temporal depth learning for face anti-spoofing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020. p. 5042-5051.

[2] GOODFELLOW, Ian J.; SHLENS, Jonathon; SZEGEDY, Christian. Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572, 2014.

[3] KURAKIN, Alexey, et al. Adversarial examples in the physical world. 2016.

[4] FEIBUSH, Eliot A.; LEVOY, Marc; COOK, Robert L. Synthetic texturing using digital filters. In: Proceedings of the 7th annual conference on Computer graphics and interactive techniques. 1980. p. 294-301.

[5] BERGHOLM, Fredrik. Edge focusing. *IEEE transactions on pattern analysis and machine intelligence*, 1987, 6: 726-741.