Adversarial Attack on Facial Recognition System with Predefined Spatial Constraints

Md Abdullah Al Maruf
Virginia Tech
Blacksburg, USA
marufm@vt.edu

Sheik Murad Hassan Anik
Virginia Tech
Blacksburg, USA
murad@vt.edu

ABSTRACT

From cancer diagnosis to self-driving cars, machine learning is profoundly changing the world. Recent studies show that the state-of-the-art machine learning systems are vulnerable to adversarial examples resulting from small-magnitude perturbations added to the input. The broad use of machine learning systems makes it significant to understand the attacks on these systems where physical security and safety are at risk.

In this project, we focus on facial recognition systems, which are widely used in surveillance and access control. We develop and investigate resilient attacks that are physically realizable and inconspicuous, that allow an attacker to impersonate another individual. The investigation focuses on white-box attacks on the face-recognition systems. We develop an attack that will perturb only those facial regions that are normal to be changed for style, pose, fashion, etc. Our model automatically generates perturbations on a specific image for impersonation attack given the predefined spatial constraints. The attack evades the state-of-the-art face-recognition system with 100% successful impersonation attack considering those spatial constraints. We compare and evaluate the efficacy of our method with state-of-the-art adversarial attacks that do not consider any constraints. Consequently, we propose some suggestions for the possible defenses for these types of spatially constrained attacks.

CCS CONCEPTS

• **Security and privacy** → Graphical / visual passwords;

KEYWORDS

Image Classification, Face Recognition, Adversarial Attack, FGSM

ACM Reference format:
Md Abdullah Al Maruf and Sheik Murad Hassan Anik. 2019. Adversarial Attack on Facial Recognition System with Predefined Spatial Constraints. In *Proceedings of CS5984, Virginia Tech, May 2019 (Security Analytics),* 12 pages.
https://doi.org/10.1145/1122445.1122456

Security Analytics, May 2019, Virginia Tech

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Deep Neural Networks(DNNs) have been achieved the state-of-the-art performance in many vision applications. They are increasingly used as a part of many surveillance and access control systems. Recent works have demonstrated that DNNs are vulnerable to adversarial perturbations.[1][2][3][4] These optimally crafted perturbations to the input of DNNs can cause the systems misbehave unexpectedly and sometimes in dangerous ways.

The facial recognition systems are widely used for various sensitive purposes, including surveillance and access control. Thus, the attackers who mislead them can cause serious ramifications. But the attackers who aim to mislead facial bio-metric systems often do not have precise control over the systems' input. Rather, the attackers may be able to control only their physical appearance only. Converting the physical scenario into a digital one is not under the control of the attackers'. The input images are also affected by several factors like lighting conditions, pose, and distance. Consequently, it is harder for the attackers to craft adversarial input that can fool the classifier. Another difficulty that the attackers face while fooling the face recognition system is that, some perturbations might be visible to the outside world. If the attacker do excessive makeup to evade the facial recognition system, the bystanders can detect it and take necessary actions.

In the light of these challenges, our attack methods must be *physically realizable* and at the same time should be *inconspicuous*. The manipulation of the attacks must be such that the perturbations are sufficiently subtle and they are imperceptible to humans, or if perceptible, seem natural.

Though the attacker has full control over the facial area, she cannot perturb over the whole face. It'll be more visible and can cause *plausible deniability* if she tries to add perturbation on those facial regions that are not normal to be changed. All she can do is to add perturbation over the areas that are natural to be changed due to style, pose, fashion, etc. Those areas include hairs, facial hairs, eyebrows, moustache, etc. Our attack method ensures that the perturbations are only applied to those facial regions that are predefined by the attacker.

In this project, we did impersonation attack. In an impersonation attack, the adversary perturbs the input image in such a way that it is recognized as a specific other face. An adversary may try to disguise her face to be recognized as an authorized other users by the face recognizer.

In this paper we demonstrate the spatially constrained impersonation attack against facial recognition system. Our method successfully misclassify all the input test images(100%) to the target classes.

Our contributions in this paper are given below:

- We introduce the spatially constrained perturbations that constantly cause misclassification for the input images of a face recognition system.
- We evaluate our attack method with the classical unconstrained attack methods and showed that the performance is much similar to the best attack methods. Our attack method is more effective in real world attack. Because the perturbations are less visible and are on specific area of a face that are normal to be changed.
- We also suggest some possible defense mechanisms for this attack. Researchers can find these suggestions helpful for future defense proposals for these attacks.

2 RELATED WORKS

We survey the related work in generating adversarial examples. Different methods have been proposed to generate adversarial example in white box setting, where the adversary has full access to the internals of the classifier. We know that the adversarial attacks have transferability property, so black-box attacks are also possible. In [5], Goodfellow *et al.* explain that the primary cause of neural networks' vulnerability to adversarial perturbation is their linear nature. This work like many other was built on top of the paper by Szegedy *et al.* [6] where they discovered several models, including state-of-the-art neural networks are vulnerable to adversarial examples and misclassification.

Goodfellow proposed the FGSM(Fast Gradient Sign Descent) method, which applies a first-order approximation of the loss function to the construct adversarial examples[2]. Optimization based methods have been proposed to create adversarial perturbations for targeted attacks[1]. All of these methods perturb all over the images. By contrast, our attack method only perturb on the predefined facial regions.

One significant related work is Sharif *et al.* [7], they attacked a face recognition system by perturbing the frames of eyeglasses. Interestingly, their perturbation is much visible in the frames, and it requires much work to successfully print the perturbation on the frames. Their work demonstrated successful physical attacks in relatively stable physical conditions with little variation in pose, distance/angle from the camera, and lighting. As they constrained the variations, we designed our attack method to be effective in diverse physical world conditions.

Another work is Eykholt *et al.* [8], they propose a similar attack algorithm, Robust Physical Perturbation (RP2) that was robust under different physical conditions. Their algorithm was tested on real-world road sign classification. Though their proposed mechanism is effective in real world scenario, they did not propose or suggest any defenses to those attacks.

3 ADVERSARIAL ATTACK ON FACIAL RECOGNITION SYSTEM

Our goal is to investigate how it is possible to create an adversarial image from the input image so that the perturbations are only added to the predefined regions specified by the attacker. We first present a threat model for our attack, then present an algorithm to generate perturbations taking all the spatial constraints into account.

3.1 Threat Model

In this section, we will discuss our threat model. For our adversarial attack, the attacker is bound by some constraints. The threat model discussion is as follows:

- White-box attack scenario: We assume a white-box scenario where the attacker will know the internals of the detection model.
- Modification restriction: Even though the adversary has white-box access to the detection system, we will maintain the model modification restriction meaning the adversary will not be able to change the detection model or any of its parameters.
- Targeted attack: We only consider impersonation attack, where the attacker seeks to have his/her face misclassified as a specific other face.
- Training data poisoning: The adversary cannot poison the facial recognition system by altering training data.

3.2 Spatially Constrained Perturbation

Current attack algorithms focusing on digital images to add adversarial perturbations to all parts of the images, including the backgrounds. But, for real world scenario, the adversary have no or little control over the background imagery while doing the adversarial attack on the facial recognition system. Though the attacker has full control over the facial area, she cannot perturb over the whole face. It'll be more visible and can cause *plausible deniability* if she tries to add perturbation on those facial regions that are not normal to be changed. All she can do is to add perturbation over the areas that are natural to be changed due to style, pose, fashion, etc. Those areas include hairs, facial hairs, eyebrows, moustache, etc. So we need to ensure that the perturbations are only applied to those facial regions that are predefined by the attacker. For that we inherit the idea of masking from the *Eykholt et al.* paper [9].

Before going through the details of the masking, lets derive our algorithm starting with the optimization method that generates a perturbation for a single image x, without considering any spatial constraints. Then, we describe how to update the algorithm taking the spatial constraints into account.

The single-image optimization problem searches for perturbation δ to be added to the input x, such that the perturbed instance $x' = x + \delta$ is misclassified by the target classifier $f_\theta(.)$:

$$min \quad H(x+\delta, x), \quad s.t. \quad f_\theta(x+\delta) = y^*$$

where H is the $lp - norm$ of the distance function, and y^* is the target class. We are only considering targeted attack in this project. The above optimization problem is reformulated in the Lagrangian-relaxed form as:

$$\text{argmin}_\delta \quad \lambda||\delta||_p \quad + \quad J(f_\theta(x+\delta) = y^*) \tag{1}$$

Here, $J(.,.)$ is the Jacobian loss function, which measures the difference between the target label y^* and the model's prediction. λ controls the regularization and distance function H is specified here as $||\delta||_p$, denoting the l_p norm of δ.

To ensure that the perturbations are constrained in the predefined facial regions, we incorporate the idea of masking. Formally, the perturbation mask M_x, is a matrix whose size is same as the size

of the input image. The mask M_x contains ones(1) in the regions where the perturbation will be added, zeroes(0) where no perturbation will be added during optimization step. In our experiments, we use Dlib library [10] to find out the hairs, the facial hairs(by modifying the jaw detector), the eyebrows and the moustache. For women, we discard the facial hair and moustache regions for masking. The following Figure 1 shows the generated mask from an input image.

(a) Original Image. **(b) Generated mask.**

Figure 1: Masks generated using dlib.

Based on the above discussion, our final robust spatially constrained perturbation is thus optimized as:

$$\text{argmin}_{\delta} \quad \lambda||M_x * \delta||_p \quad + \quad J(f_\theta(x + M_x * \delta) = y^*) \tag{2}$$

Among the well-known attack methods we used a modified version of Fast-Gradient Sign Descent Method(FGSM) to attack the input images. The perturbation of FGSM attack is calculated as:

$$\delta = \epsilon * sign(\nabla_x J(\theta, x, y^*)) \tag{3}$$

Where the θ is the parameters of a model, x is the input to the model,y^* is the targets associated with$x$, ϵ is a small amount of perturbation, and $J(\theta, x, y^*)$ is the cost used to train the neural network. Our modified FGSM attack taking spatial constraints into account is:

$$\delta = M_x * \epsilon * sign(\nabla_x J(\theta, x, y^*)) \tag{4}$$

We used Foolbox [11] an attack toolbox to attack the facial recognition system. We modified the FGSM() method of foolbox to add these spatial constraints in our attack model. In the next section we will discuss the experiments and their results at length.

4 EXPERIMENT

In this section, we will discuss our approach for creating adversarial samples to fool facial recognition and classification system. We have categorized this section in three subsections, (i) Dataset and Classifier, (ii) Experimental Design, (iii) Result. We will begin with our dataset and classifier for facial recognition.

4.1 Dataset and Classifier

In this project we used the MCS2018 dataset [12] as the source of our facial images. The dataset contains 1 million images of 250x250 pixels resolution. We used the current state of the art image classification model architecture ResNet50 as our classifier. It was pretrained on the MCS2018 dataset. It can classify images of resolution 112x112 pixels. So, we re-sized the images from 250x250 pixels to 112x112 pixels in our code.

For evaluating our attack approach we randomly selected 100 source classes and corresponding 100 random target classes. In this report, for the limitation of space, we present evaluation of only 50 of the 100 pairs.

4.2 Experimental Design

Our goal is to generate adversarial samples for facial recognition and classification systems with predefined spatial constraints so that the classifier predicts the adversarial sample to different targeted class than the original.

One key challenge in our attack was the spatial constraint. Our approach uses facial mask to limit perturbation area on the source image. For generating facial mask, we used the DLib [10] library which provides facial landmarks on input image. The library provides complete and precise documentation for every class and function.It provides landmarks on specific facial regions like head, hair, eyes, node, nostrils, jawline etc. As our attack needs to perturb only on the facial hair regions, we chose head, eyebrows, mustache and jawline regions for our purpose. We had to enlarge some of these regions with manual tuning for complete coverage. Figure 10 shows generated masks for our source images.

We developed and evaluated our attack using Foolbox [11] which is a tool designed specifically for adversarial attacks. We tried several attack methods like Gradient Attach, L2 Iterative Attack, C&W Attack etc. Due to some limitations in GPU power, these approaches were too slow in our machine. So, we went with the Fast Gradient Sign Descent Method (FGSM) to generate adversarial samples for our attack.

The FGSM attack provided in Foolbox is the basic one. We modified this method to add the mask in the calculation. The mask is multiplied with perturbation to limit attack region. The process follows the equation 4.

Here, we discuss about the evaluation metrices for our proposed adversarial attack on facial recognition systems. The attack success rate will be evaluated using the following equation [6]:

$$successRate = \frac{imagesMisclassifiedDueToAttack}{imagesClassifiedCorrectlyBeforeAttack}$$

We will also measure the SSIM as the similarity metric for the perturbed image and the actual image. DSSIM (Structural Dissimilarity) is a distance metric derived from SSIM (Structural SIMilarity). The basic form of SSIM compares three aspects of the two image samples, luminance (l), contrast (c), and structure (s). The SSIM score is then described in the following equation:

$$SSIM(x, y) = l(x, y) * c(x, y) * s(x, y)$$

4.3 Result

We evaluate the effectiveness of our approach by generating 100 adversarial examples for 100 randomly chosen source classes and 100 corresponding target classes (also chosen randomly) but for the limitation of space in this report we will discuss only first 50 pairs. We used Structural Similarity Index (SSIM) and L2 distance between source image and adversarial image for evaluation. The result is summarized in Table 1.

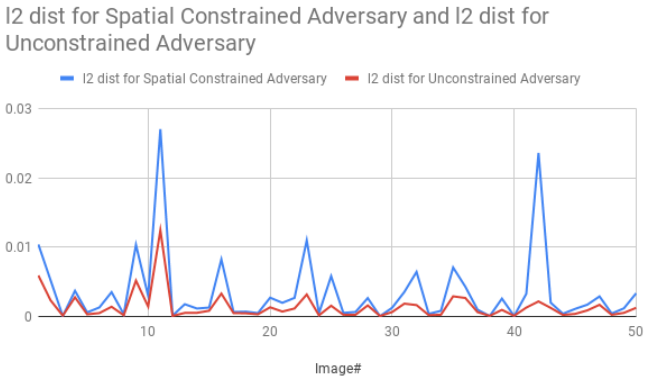

Figure 2: l2 distance between original and adversarial images.

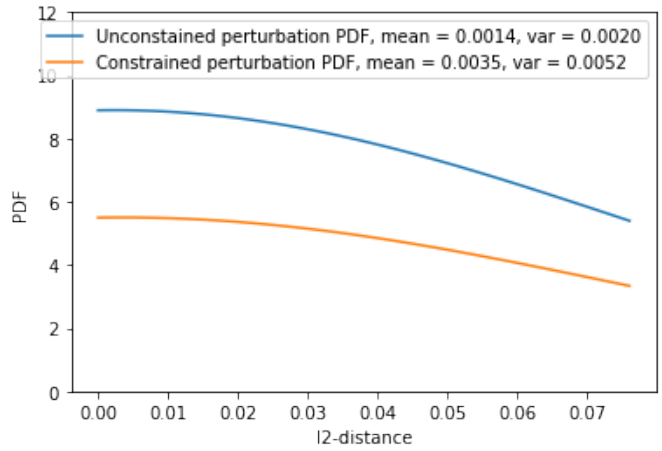

Figure 3: Probability Density Function for l2-distances.

We found the perturbation generated by our approach is not visible even when amplified 10 times. So, in this report, we attached the perturbation images amplified 100 times.

We performed our attack without using no mask (on the whole image) and using the generated mask for a comparison of perturbation, SSIM and L2 distance. The results of attack with mask on facial hair regions are shown in Figure 6 and 7. Here, from the left, the first column (a) represents the source image, the second column (b) represents the adversarial image, the thirst column (c) represents the generated perturbation and the last column (d) represents the target image.

The results of attack on the entire image are shown in Figure 8 and 8. For the comparison of masked and unmasked attack, we demonstrated the generated perturbations side by side in Figure 11. The (a) column denotes the masked perturbation and column (b) denotes perturbation on entire image. It is clear that the masked perturbations are much more visible than the unmasked ones. This is because the region is limited and to classify the algorithm has to perturb more in the defined region.

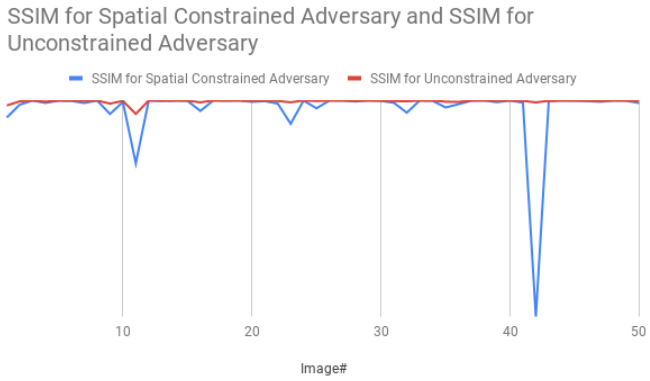

Figure 4: log-SSIMs between the original and adversarial images.

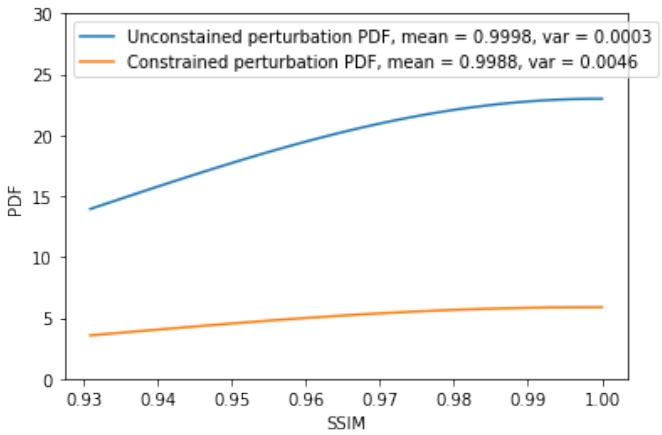

Figure 5: Probability Desity Functions for the SSIMs.

The Figure 2 shows the l2 distance among our 50 source images and generated adversarial images for both masked and non-masked attack variations. It can be seen that the l2 distance of the masked attack is higher for all source images than the non-masked samples. The Figure 3 shows the probability density function of the l2 distances across our source and adversarial images. Here we see that the mean and variance of masked attack is higher than the non-masked attack.

The Figure 4 shows the SSIM between the original and adversarial image across the 50 randomly chosen images for both masked (constrained) and non-masked (unconstrained) attack variations. It can be seen that the SSIM of the masked attack is lower for all source images compared to the corresponding non-masked samples. The Figure 5 shows the probability density function of the SSIM across selected source and adversarial images. It can be seen that although the mean is same in both type of attacks but the variance of masked attack is much higher than the non-masked attack.

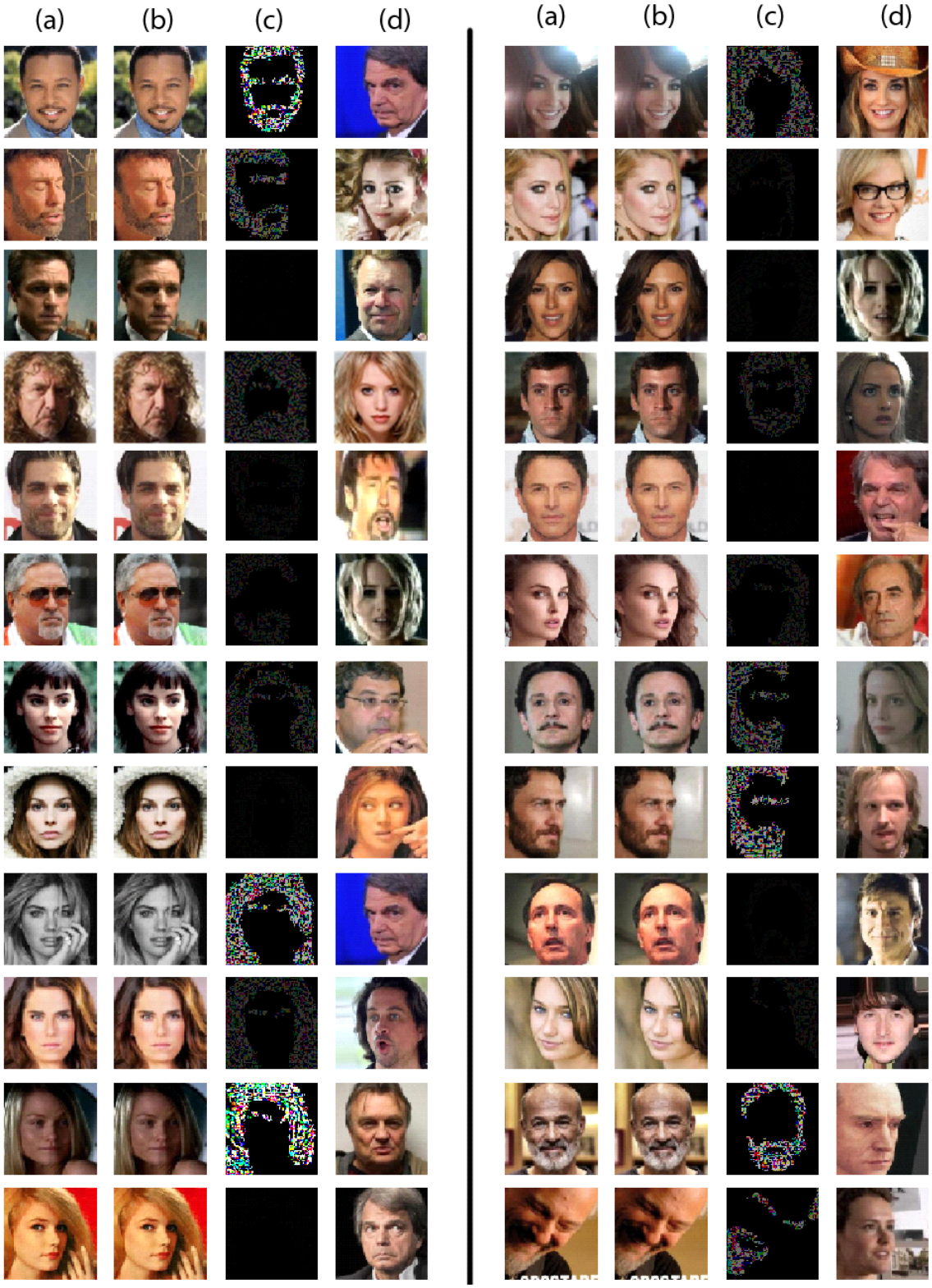

Figure 6: (a) Original input image, (b) Adversarial image, (c) The perturbation x 100, (d) The target image. Attack results with spatial constraints (part 1).

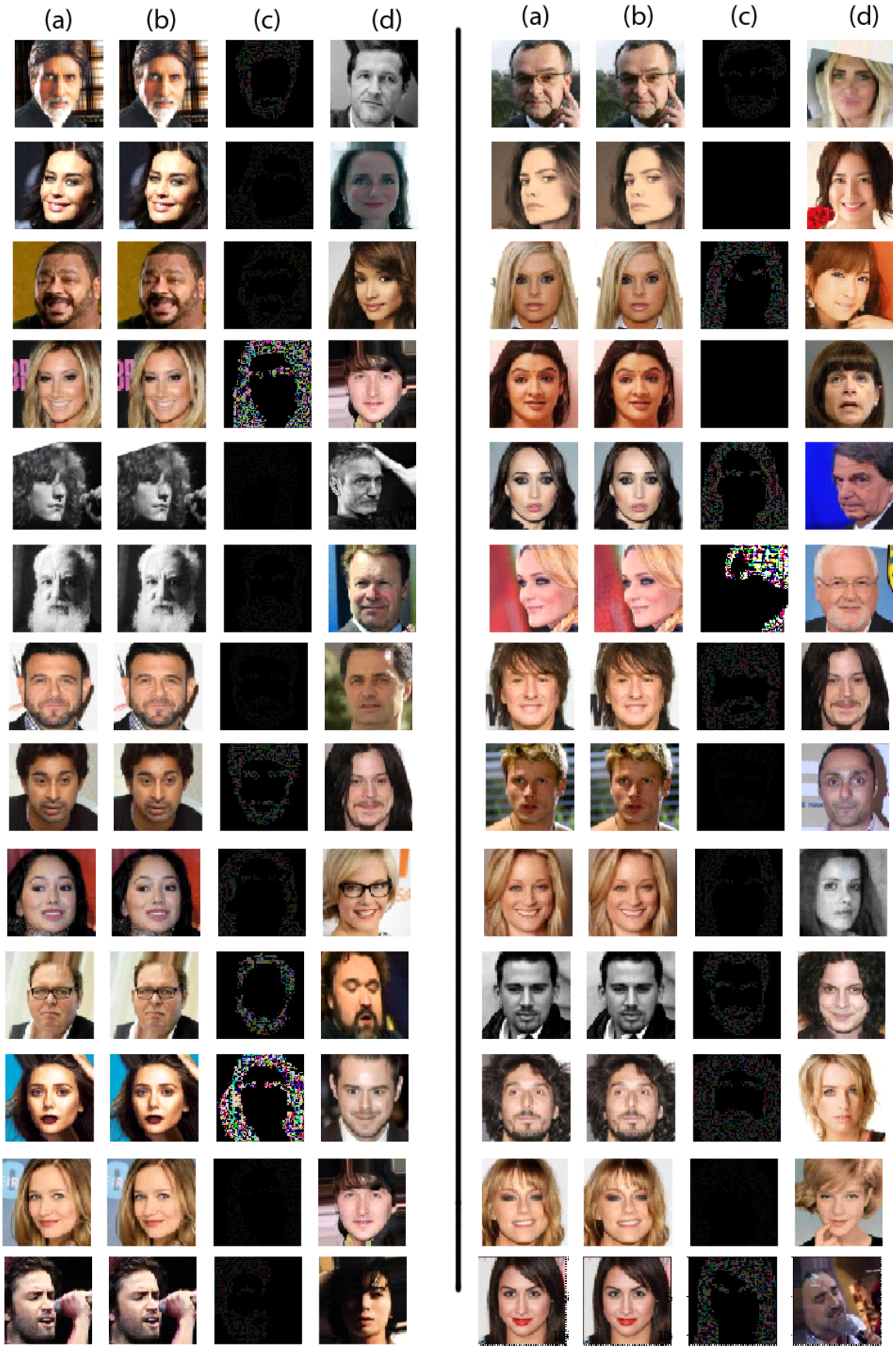

Figure 7: (a) Original input image, (b) Adversarial image, (c) The perturbation x 100, (d) The target image. Attack results with spatial constraints (part 2).

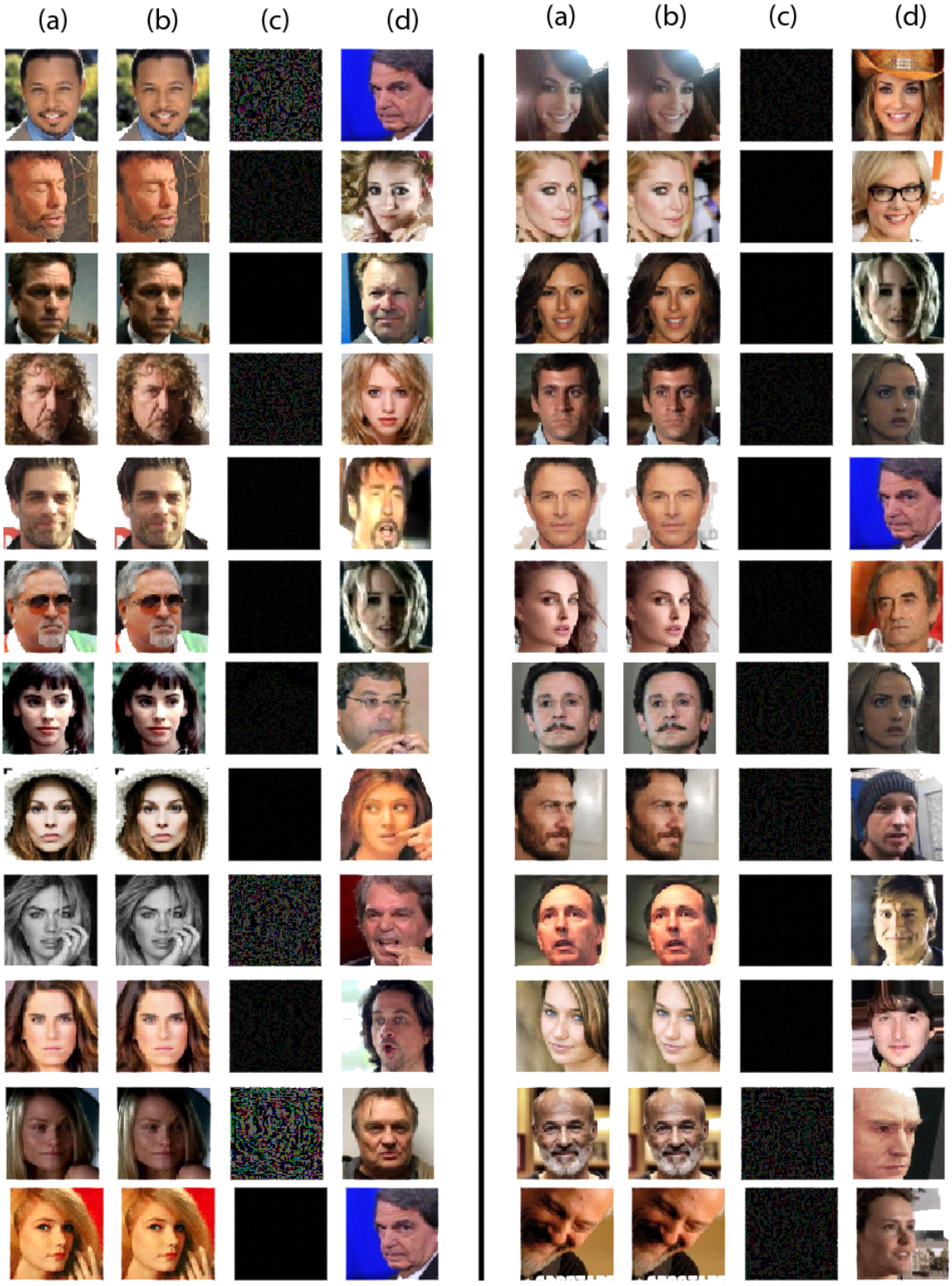

Figure 8: (a) Original input image, (b) Adversarial image, (c) The perturbation x 100, (d) The target image. Attack results without spatial constraints (part 1).

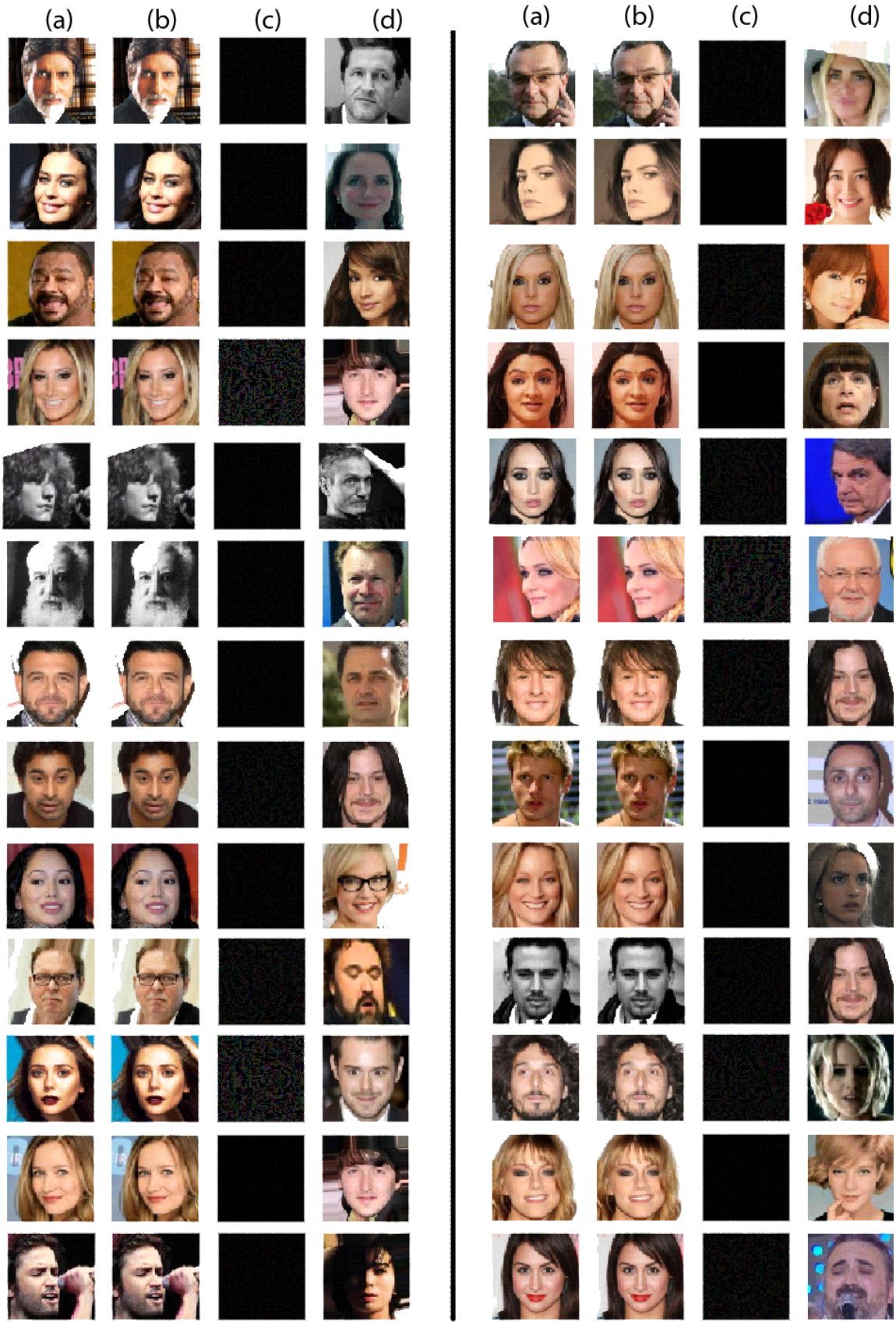

Figure 9: (a) Original input image, (b) Adversarial image, (c) The perturbation x 100, (d) The target image. Attack results without spatial constraints (part 2).

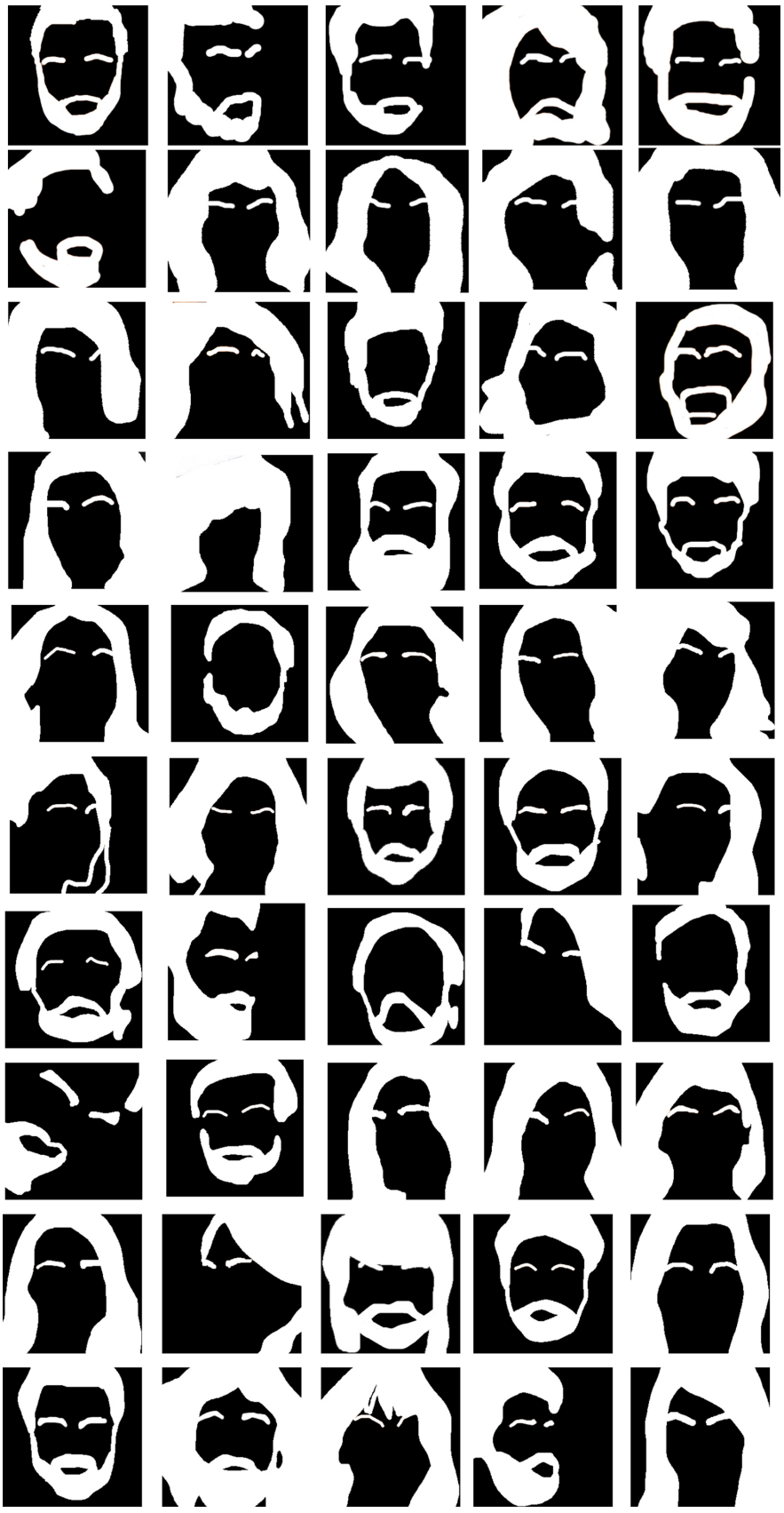

Figure 10: Generated and Modified Facial Masks.

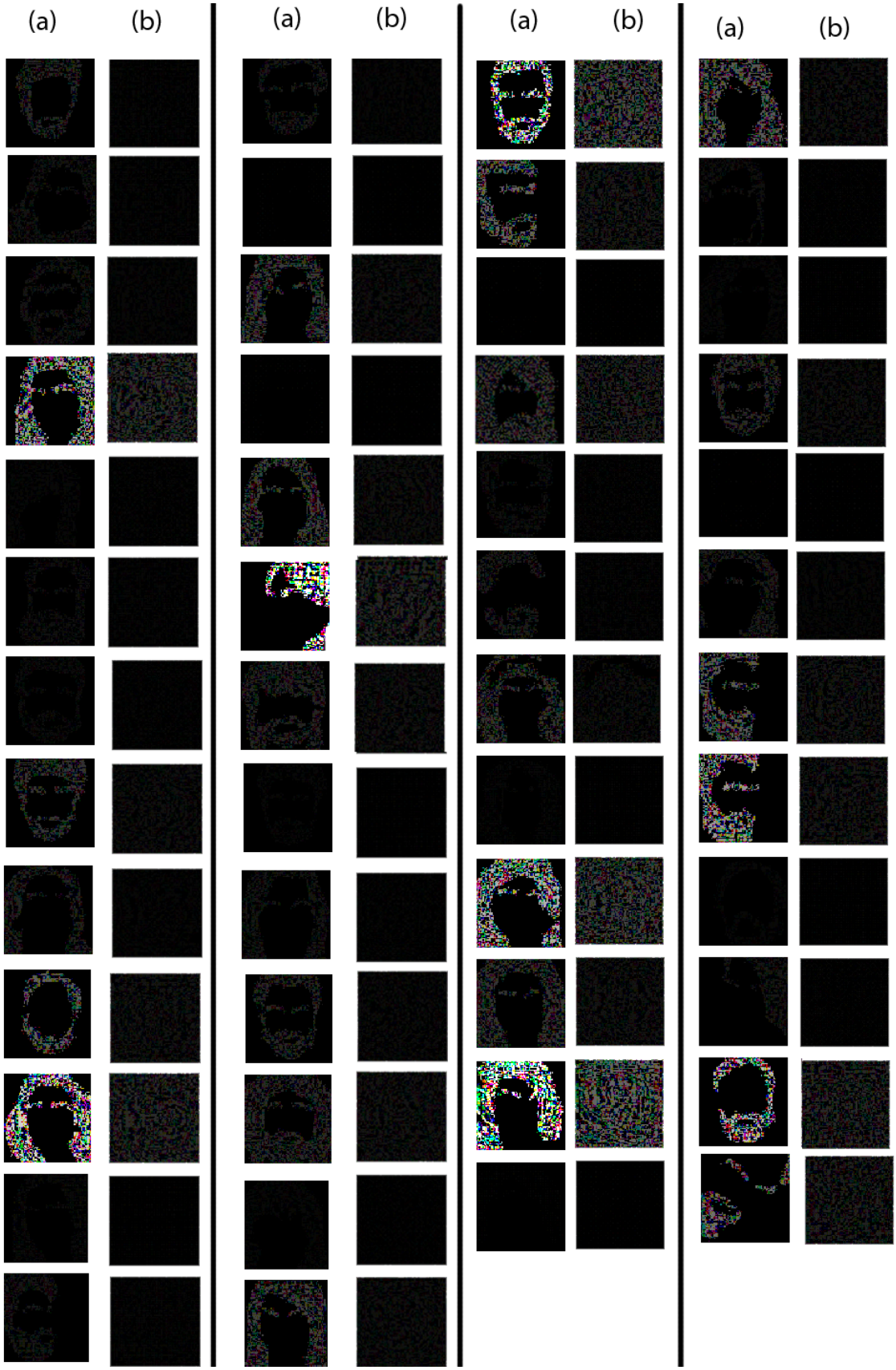

Figure 11: (a) Perturbation with Spatial Constraints, (b) Perturbation without Spatial Constraints. Perturbation comparison between Constrained and Unconstrained attack for an input image.

Table 1: Result summary

Image #	Image Label	Target Label	SSIM	L2-distance	Unmasked SSIM	Unmasked L2-distance
1	402	73	0.9974589628	0.010413501	0.999261296	0.005930241
2	337	5	0.999373273	0.005244055	0.9998725691	0.0023396933
3	168	454	0.9999998913	7.44E-05	0.9999996976	0.00012813989
4	472	154	0.9996434846	0.0037152076	0.9998359635	0.0027752158
5	219	430	0.9999734674	0.00059546606	0.9999955188	0.00031720795
6	472	510	0.9999454228	0.0013095279	0.9999930825	0.00049691205
7	193	341	0.9996267199	0.0035243984	0.9999540663	0.0014121939
8	447	424	0.9999973421	0.0003307324	0.9999992101	0.0001828034
9	402	219	0.9979918286	0.010365272	0.999535511	0.0051879724
10	180	113	0.9997777483	0.0029819908	0.9999469789	0.0014415347
11	213	363	0.9906370343	0.027087025	0.9979982131	0.012470216
12	402	457	0.9999994801	0.00012214346	0.9999997091	9.12E-05
13	129	402	0.9999322638	0.0017789219	0.9999947266	0.0005203335
14	16	193	0.9999670692	0.0011550139	0.9999932491	0.0005328522
15	338	341	0.9999593571	0.0012939681	0.9999841655	0.0008336142
16	331	377	0.9984448548	0.008255266	0.9997282001	0.003313337
17	17	506	0.9999931333	0.0006545648	0.999995732	0.00049747044
18	168	322	0.9999782994	0.00070662634	0.9999918306	0.00046588885
19	310	364	0.9999922562	0.00049823214	0.9999961347	0.00032051286
20	112	89	0.9998032021	0.0027236978	0.9999522288	0.0013346148
21	330	111	0.9999084581	0.001981562	0.9999888074	0.0007172512
22	36	399	0.9995346575	0.0026945402	0.9999272841	0.001160728
23	9	425	0.9965237377	0.011005732	0.9997318699	0.003186235
24	331	18	0.9999918301	0.0004764049	0.9999989774	0.00017560266
25	292	331	0.9988377039	0.0058352514	0.9999205318	0.0015599675
26	330	219	0.9999891341	0.0005276187	0.9999976426	0.00025626575
27	472	114	0.999990548	0.0006567561	0.999998705	0.00025359527
28	89	433	0.9998573968	0.0026633856	0.9999533531	0.0016048664
29	402	303	0.9999998613	4.70E-05	0.9999996812	7.74E-05
30	109	213	0.9999432857	0.0012656685	0.9999858213	0.0006451972
31	89	184	0.9996545377	0.0035470212	0.9999133668	0.0018588462
32	418	361	0.9981842177	0.0064504864	0.9998823014	0.0016522438
33	285	402	0.9999947851	0.00038694064	0.9999987652	0.00020054157
34	331	112	0.999964763	0.0008183232	0.9999972817	0.000258078
35	50	282	0.9989658685	0.007084377	0.9998296599	0.0029146217
36	425	133	0.9994460702	0.004310608	0.9997889077	0.0026803215
37	509	37	0.9999679851	0.000989254	0.9999875311	0.000637349
38	436	219	0.9999998493	6.52E-05	0.9999996544	9.70E-05
39	43	402	0.9997513418	0.0025901082	0.9999667666	0.00096087414
40	37	33	0.9999998431	7.01E-05	0.9999997109	0.00010106935
41	402	321	0.9996826075	0.0032913834	0.9999540673	0.0012955012
42	206	302	0.9682045338	0.023642939	0.9997219543	0.0021821926
43	112	307	0.9998782554	0.0019964904	0.9999565593	0.0012454166
44	139	17	0.9999966853	0.000401299	0.9999991125	0.00022975143
45	89	282	0.9999674474	0.0011028155	0.9999961664	0.00038001727
46	112	43	0.9999175097	0.0017096956	0.9999796604	0.00088544784
47	472	361	0.9998149151	0.0029057262	0.9999389298	0.001695947
48	405	364	0.9999934036	0.00043009172	0.999997919	0.00024792392
49	7	398	0.9999596485	0.0011853324	0.9999925926	0.00052992994
50	115	45	0.9996318311	0.0033515692	0.9999505536	0.001258083

5 DISCUSSION & LIMITATIONS

From the result section, we can say that adding spatial constraints requires more perturbations than adversarial attacks without any constraints. That is why the l2-distance between the adversarial examples and the original image is much higher for the spatial constrained attack. The structural similarity between the original images and the adversarial examples are also higher for the unconstrained adversarial attack. For this comparison, we fixed the target classes of each images.

Possible Defenses To defend this real world spatial constrained attack, we are proposing three defense mechanisms. (i) One incorporates adversarial training. We can synthetically generate such many adversarial samples and train our detector to classify them as fake. Putting a detector before the classifier will help in this regard. (ii) Also, we can assume that the adversary has different facial structure than the target person. If we can train the classifier with some extra structural information over the faces(like 3D structure of the face), it will be more difficult for the attacker to generate effective perturbation without being observed to the outside world. (iii) It is obvious that there are some specific areas like hairs, eyes, facial hairs are more prone to these real world attacks. Moreover, the attacker needs to transform the source face to the target face considering many different angles and lighting conditions. As the possible transformations are obvious, the facial recognizer can reverse those transformations to generate a temporary image. And then check the structural similarity of that image to the representative target class image. As the perturbed faces normally have different structure than the target face, the detector can detect the modified face. These are the proposed defense mechanisms. We will find the efficacy of these mechanisms broadly in our project.

Limitations Though our attack is 100% successful for this test dataset, one important thing is that our dataset is small. And some hyper-parameters needed to tune a lot for some of the input images to converge. We could not do fully automation for our attack method. Each image required different type of attention to make the perturbation converge in our permissible range. For some of the images, we needed manual masking to correctly find out our desired masks. But from the evaluation and the comparison with the unconstrained attack, we can establish that our attack formulation is correct.

In addition, the notion of inconspicuousness is subjective, and the only way to quantify it adequately requires to incorporate human-subject studies. We did SSIM(Structure Similarity Index) for that measure. In our future work, we plan to work on make this attack more robust and more compatible to the real-world. So that we can perturb different angles of a image and can generate an uniform perturbation for that image.

6 CONCLUSION

In this project, we devised an approach to create adversarial samples with spatial constraints to fool facial recognition and classification models. Using our approach we crafted adversarial samples of 100 randomly chosen source classes. All of the generated adversarial samples were successful in fooling the classification model. Our work shows that, it possible to generate highly efficient adversarial samples with predefined constraint like facial hair to fool available facial recognition and classification systems. We also show that the crafted adversarial images are indistinguishable from the corresponding original images. To stop security vulnerability and identity threat, new defense mechanisms should be devised to defend against such adversarial attacks.

REFERENCES

[1] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2017, pp. 39–57.

[2] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[3] J. Kos, I. Fischer, and D. Song, "Adversarial examples for generative models," in *2018 IEEE Security and Privacy Workshops (SPW)*. IEEE, 2018, pp. 36–42.

[4] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, "Universal adversarial perturbations," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1765–1773.

[5] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *CoRR*, vol. abs/1412.6572, 2015.

[6] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. J. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *CoRR*, vol. abs/1312.6199, 2013. [Online]. Available: http://arxiv.org/abs/1312.6199

[7] M. Sharif, S. Bhagavatula, L. Bauer, and M. K. Reiter, "Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016, pp. 1528–1540.

[8] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. X. Song, "Robust physical-world attacks on deep learning visual classification," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1625–1634, 2018.

[9] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. Song, "Robust physical-world attacks on deep learning models," *arXiv preprint arXiv:1707.08945*, 2017.

[10] D. E. King, "Dlib-ml: A machine learning toolkit," *Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009.

[11] J. Rauber, W. Brendel, and M. Bethge, "Foolbox: A python toolbox to benchmark the robustness of machine learning models," *arXiv preprint arXiv:1707.04131*, 2017. [Online]. Available: http://arxiv.org/abs/1707.04131

[12] D. Gragnaniello, F. Marra, G. Poggi, and L. Verdoliva, "Perceptual quality-preserving black-box attack against deep learning image classifiers," *CoRR*, vol. abs/1902.07776, 2019.