# Towards Face Encryption by Generating Adversarial Identity Masks CVPR Proceedings

Wenhan Liu
University of Waterloo
Waterloo, ON, Canada
w288liu@uwaterloo.ca

Zhiqi Bei
University of Waterloo
Waterloo, ON, Canada
zbei@uwaterloo.ca

## Abstract

*Nowadays, the development of social media and networks has brought a huge amount of personal data shared publicity, and the Face Recognition System has increased the potential risks for privacy leakage of personal information. To users, this would cause privacy problems that they don't like. Hence, it's necessary to have a method to protect their privacy information from different unfriendly systems, such as unauthorized face recognition systems. However, users' experience with the use of social media should not be affected, and as a result, the Targeted Identity-Protection Iterative Method (TIP-IM) is introduced.*

## 1. Glossary

TIM-IM: Targeted Identity-Protection Iterative Method, which is the proposed method of generating the adversarial identity mask to probe images.

Real image/Probe image: the user's original image.

Protected image: The image after adding the adversarial mask.

Gallery set: A gallery set that unauthorized face recognition systems use to identify the target's identity in an image.

Target Set: A set of images containing ten images of different identities, and the TIP-IM would fool the unauthorized face recognition system to detect faces in probe image to one of the identities of faces within the target set.

## 2. Introduction

Since the development of social media has brought a huge impact on people's daily life and caused influenced their privacy information. Also, the fast development of Artificial Intelligence (AI) and Machine Learning (ML) has derived a lot of methods to acquire users' personal information from their social media pages, and unauthorized systems may access and make use of users' privacy information. With the information, unauthorized third parties could associate users with their preferences, visited places, social interactions, etc. However, with the use of the Targeted Identity-Protection Iterative Method, an adversarial identity mask can be added to user's face in the images they post, and with the use of this adversarial mask, an unauthorized face recognition system will not be able to detect the true identity of the user in the image anymore. While for the users, after adding the adversarial identity mask, the image would still be natural for them to view, and won't cause any bad influence on their experience with viewing the images.

## 3. Scope of reproducibility

According to the original paper, the experiments we reproduced in this work are

1. Using ArcFace model with gamma = 0.0 to generate adversarial identity masks to the images and tested the Rank-N-T and Rank-N-UT accuracy with N = 5 and N = 1.
2. Using ArcFace model with gamma = 0.1 to generate adversarial identity masks to the images and tested the Rank-N-T and Rank-N-UT accuracy with N = 5 and N = 1.
3. Using ArcFace model with gamma = 1.75 to generate adversarial identity masks to the images and tested the Rank-N-T and Rank-N-UT accuracy with N = 5 and N = 1.
4. Using ArcFace model with gamma = 0.007 to generate adversarial identity masks to the images and tested the Rank-N-T and Rank-N-UT accuracy with N = 5 and N = 1.
5. Using ArcFace model with gamma = 0.004 to generate adversarial identity masks to the images and tested the Rank-N-T and Rank-N-UT accuracy with N = 5 and N = 1.
6. Using ArcFace model with gamma = 0.004 to generate adversarial identity masks to the images and tested the Rank-N-T and Rank-N-UT accuracy with N = 5 and N = 1.
7. Using CosFace model with gamma = 0.0 to generate adversarial identity masks to the images and tested the Rank-N-T and Rank-N-UT accuracy with N = 5 and N = 1.
8. Using CosFace model with gamma = 0.1 to generate adversarial identity masks to the images and tested the Rank-N-T and Rank-N-UT accuracy with N = 5 and N = 1.

9. Using SphereFace model with gamma = 0.0 to generate adversarial identity masks to the images and tested the Rank-N-T and Rank-N-UT accuracy with N = 5 and N = 1.

10. Using SphereFace model with gamma = 0.1 to generate adversarial identity masks to the images and tested the Rank-N-T and Rank-N-UT accuracy with N = 5 and N = 1.

11. For the experiments, we generated the PSNR and SSIM value between the original probe image and the generated protected image for ArcFace, CosFace and SphereFace models with gamma = 0.0 and gamma = 0.1 to see how good/natural the protected image is.

## 4. Background, Related Work and Literature Review

There are several existing methods and researches [12] that aim to solve the problem of user privacy information leakage due to unauthorized systems, and three of them will be talked about in this paper.

The first one is the Obfuscation-based method. This is rather straightforward from its name, it contains methods such as blurring, pixelation, darkening, occlusion, etc. For some simple obfuscation-based methods, they are not very effective against the more advanced face recognition systems as they can adapt to the obfuscation patterns. However, there are better methods such as Face-Off [1], this method introduces strategic perturbations to the user's face so that it can not be correctly recognized, and this method has successfully deceived three commercial face recognition services from Microsoft, Amazon and Face++. And there's an obfuscation method [11] that based on head inpainting generates images that are less unnatural images that could fool the face recognition systems. Meanwhile, the generative adversarial networks (GANs) [6] could synthesize the probe images on data distribution in order to achieve the purpose of image obfuscation.Nevertheless, although some of these current existing methods do work well against some face recognition systems, they often tend to lead to unnatural results, and the visual appearance of the results is not ideal for the users and would lead to a bad experience.

The second method is the Adversarial method. As the paper [2, 7] states, most deep neural networks are susceptible to adversarial examples, hence this would make the face recognition systems formed by deep neural networks more likely to be predicting the wrong result. For example, Fawkes [10] can fool unauthorized facial recognition models by introducing adversarial examples into the training data [13]. And there's a recent work [3] that tends to protect the true identity of the probe image by exploring it through a game theory perspective. However, unlike the Targeted Identity-Protection Iterative Method, this kind of method is often experimented with in a closed-set classification scenario and the naturalness of the generated protected image can not be controlled.

The third method would be Differential Privacy (DP) [4]. Many successful and robust methods were developed based on the idea of Differential privacy, and it was first introduced in the context of machine learning. The difference between DP and TIP-IM is that DP withholds the existence of entities in a dataset while TIP-IM focuses on concealing the identity of a single person by exploiting the vulnerability of neural networks [13].

## 5. Methodology

Let $x^r$ denote the real user images, $m^a$ denoted the adversarial identity mask, and $x^p$ denoted the protected image after adding the adversarial identity mask. Therefore, $x^p$ = $x^r$ + $m^a$. As stated in the original paper, the purpose of the mask is to mislead the unauthorized face recognition system, so that when the system is processing the user's image, it will recognize the user's image as an image in the target set. So, to achieve this goal, the distance between the protected image $x^p$ and the target image must be smaller than the distance between the protected image $x^p$ and the real image $x^r$, which is

$$\exists x^r \in G_I, \forall x \in G_y : D_f(x^p, x) > D_f(x^p, x^t) \quad (1)$$

Where $G_y$ represents the gallery set that contains all the real images and $G_I$ represents the target set.$D_f$ represent the distance function, which is calculated as $D_f(x_1, x_2) = ||f(x_1) - f(x_2)||_2^2$.

However, the paper also mentions three considerations: naturalness, unawareness of the gallery set, and unknown face system.

### 5.1. Naturalness

From the user's perspective, most users do not want their photos on social media to be blurry. In the original paper, they adjust the naturalness of the image by restricting the $l_p$ norm function between the protected image and the real image, as $||m^a||_p \leq \varepsilon$. We keep the in the original paper without any modification during the reproduction.

### 5.2. Unawareness of the gallery set

Since we have no ways to know the gallery set of the real-word face recognition system. The paper uses their target set as a surrogate for $G_I$, and uses $x^r$ instead of $G_y$

### 5.3. Unknown face system

Since we also have no ways to know the face recognition model of the unauthorized face recognition system. The paper used ArcFace, MobileFace, ResNet50, SphereFace, FaceNet, and CosFace models as surrogate models separately against the protected image produced by ArcFace, MobileFace, and ResNet50 separately. In our reproduction, we only use ArcFace, CosFace, and SphereFace models to produce the mask, and use ArcFace, CosFace, and

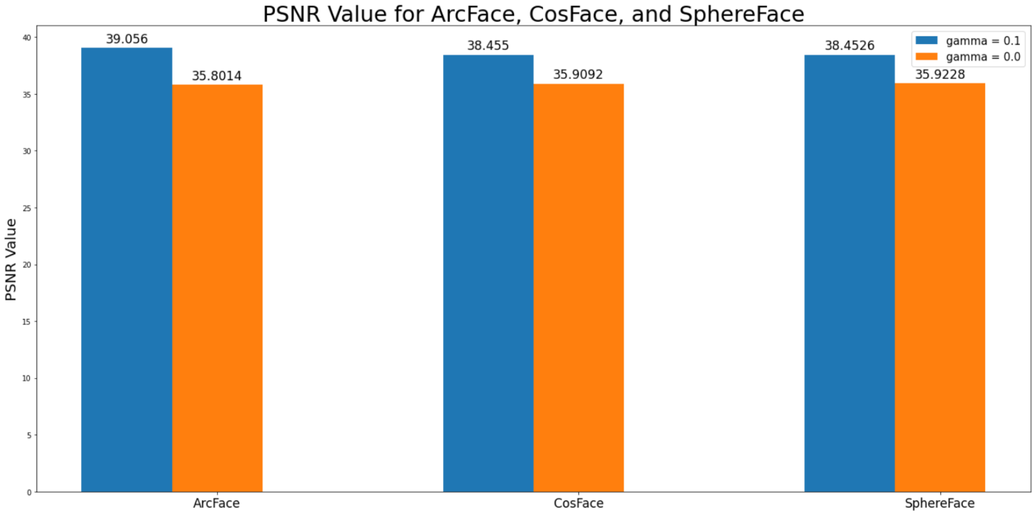


Figure 1. This shows the comparison of the PSNR values between the probe images and protected images generated by ArcFace, CosFace, and SphereFace models with gamma = 0.0 and gamma = 0.1. Since larger gamma would cause the generated protected image to be more natural, it's reasonable for them to have a higher PSNR value than the protected images generated by models with gamma = 0.0.

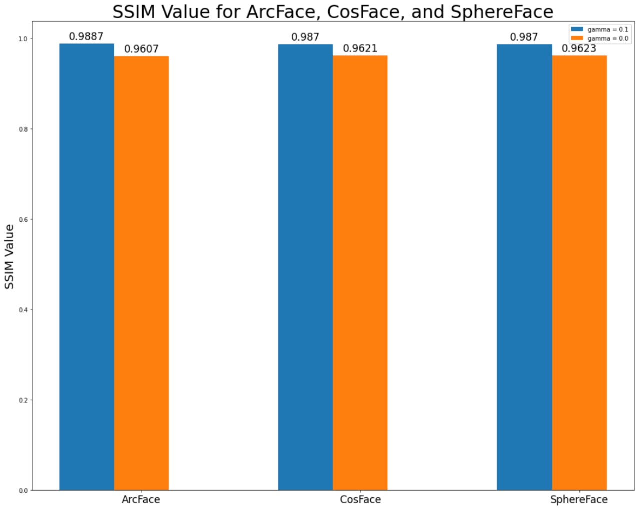


Figure 2. This shows the comparison of the SSIM values between the probe images and protected images generated by ArcFace, CosFace, and SphereFace models with gamma = 0.0 and gamma = 0.1. Same with PSNR value, Since larger gamma would cause the generated protected image to be more natural, it's reasonable for them to have a higher SSIM value than the protected images generated by models with gamma = 0.0.

SphereFace models as surrogate models against them separately.

### 5.4. The loss function

To address the above concerns, the loss function of the targeted identity-protection iterative method (TIP-IM)

$$\begin{aligned} \min L_{iden}(x^t, x^p) = D_f(x^p, x^t) - D_f(x^p . x^r) \\ \text{s.t.} ||x^p - x^r||_p \leq \varepsilon, L_{nat}(x^p) \leq \eta \end{aligned} \tag{2}$$

Where $L_{iden}$ represent the relative identification loss and $L_{nat}(x^p) \leq \eta$ is a constrain condition to make x look natural.

In order to further make the image more natural. The paper uses the *maximum mean discrepancy*(MMD) to the $L_{nat}$

$$\text{MMD}(X^p, X^r) = ||\frac{1}{N}\sum_{i=1}^{N}\phi(x_i^p) - \frac{1}{N}\sum_{J=1}^{N}\phi(x_J^R)||^2 \tag{3}$$

where $X^p = \{x_1^p, ..., x_N^p\}$, and $X^r = \{x_1^r, ..., x_N^r\}$.

Then put the MMD function back to the equation (2), the final loss function is

$$\min L(X^p) = \frac{1}{N}\sum_{i=1}^{N} L_{iden}(x_i^t, x_i^p) + \gamma * \mathrm{MMD}(X^p, X^r),$$
$$\text{s.t. } ||x_i^p - x_i^r||_p \le \varepsilon, \tag{4}$$

where $\gamma$ is a hyper parameter to balance these two losses. And in our reproduction, we adjust the naturalness of the image by modifying the value of $\gamma$.

## 6. Evaluation

We selected 50 people from the database as an evaluation sample and use the images of those 50 people plus our target set images to build the gallery set, where the target set consists of 10 randomly selected images of different people (other than that 50 people). We use ArcFace, CosFace, and SphereFace models to build the mask respectively, then we use ArcFace, CosFace, and SphereFace models to simulate the unauthorized face recognition system and process the face recognition on the protected image. The method used to measure whether the mask is effective is Rank-N-T and Rank-N-UT. Rank-N-T means that at least one image from the top N predicted result of our protected image belongs to the target set, and Rank-N-UT means that the people in the top N predicted result is different from the people in the protected image.

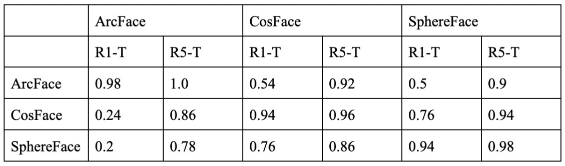
After we got the result, we performed a comparison between the models. For each model, we computed the PSNR and SSIM value between the probe images and their generated protected images to check the quality of the images generated. And the results are shown in Figure 1 and Figure 2. And as a result, the images that got generated by models with higher gamma value tends to have higher PSNR and SSIM value. This is as expected, because a higher gamma value can increase the naturalness of the generated image and make it look more identical to probe image to human eyeballs.

## 7. Result

The figures 3-6 shows the result of using ArcFace, CosFace, and SphereFace to produce the mask and against with the simulated unauthorized face recognition system using ArcFace, CosFace, and SphereFace model, evaluating by Rand-1-T, Rank-5-T, Rank-1-UT, and Rank-5-UT.

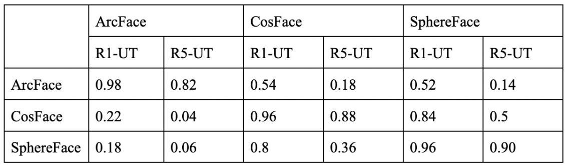
Figure 7 and 8 shows the result of using only ArcFace model to produce the mask but with different gamma value against the simulated unauthorized face recognition system using ArcFace, CosFace, and SphereFace model, and also evaluating by Rand-1-T, Rank-5-T, Rank-1-UT, and Rank-5-UT.

Although the paper states that the protection would be successful even with gamma up to 2.5, however, with gamma = 0.1, the generated protected images were already very natural to users' eyeballs. And the protected images that got generated by models with gamma = 0.1 just fails the protection of the user's identity as we can see that the Rank-N-T and Rank-N-UT values were very low compared with the values of gamma = 0.0. Hence, gamma can't be set to a very high value, and it needs to be between 0.1 and 0.0.



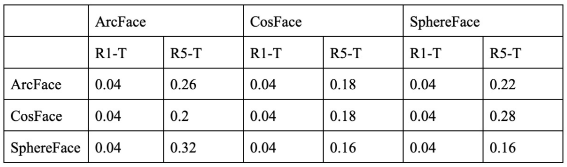
| | ArcFace | | CosFace | | SphereFace | |
|---|---|---|---|---|---|---|
| | R1-T | R5-T | R1-T | R5-T | R1-T | R5-T |
| ArcFace | 0.98 | 1.0 | 0.54 | 0.92 | 0.5 | 0.9 |
| CosFace | 0.24 | 0.86 | 0.94 | 0.96 | 0.76 | 0.94 |
| SphereFace | 0.2 | 0.78 | 0.76 | 0.86 | 0.94 | 0.98 |

Figure 3. Rank-1-T and Rank-5-T with gamma = 0 of using ArcFace, CosFace, and SphereFace models to build the mask against different models



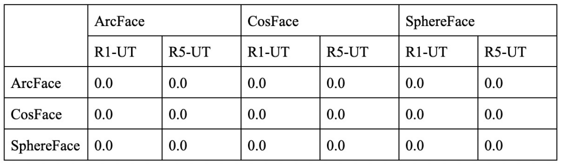
| | ArcFace | | CosFace | | SphereFace | |
|---|---|---|---|---|---|---|
| | R1-UT | R5-UT | R1-UT | R5-UT | R1-UT | R5-UT |
| ArcFace | 0.98 | 0.82 | 0.54 | 0.18 | 0.52 | 0.14 |
| CosFace | 0.22 | 0.04 | 0.96 | 0.88 | 0.84 | 0.5 |
| SphereFace | 0.18 | 0.06 | 0.8 | 0.36 | 0.96 | 0.90 |

Figure 4. Rank-U-T and Rank-5-UT with gamma = 0 of using ArcFace, CosFace, and SphereFace models to build the mask against different models



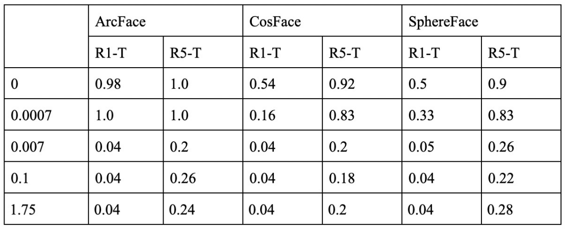
| | ArcFace | | CosFace | | SphereFace | |
|---|---|---|---|---|---|---|
| | R1-T | R5-T | R1-T | R5-T | R1-T | R5-T |
| ArcFace | 0.04 | 0.26 | 0.04 | 0.18 | 0.04 | 0.22 |
| CosFace | 0.04 | 0.2 | 0.04 | 0.18 | 0.04 | 0.28 |
| SphereFace | 0.04 | 0.32 | 0.04 | 0.16 | 0.04 | 0.16 |

Figure 5. Rank-1-T and Rank-5-T with gamma = 0.1 of using ArcFace, CosFace, and SphereFace models to build the mask against different models



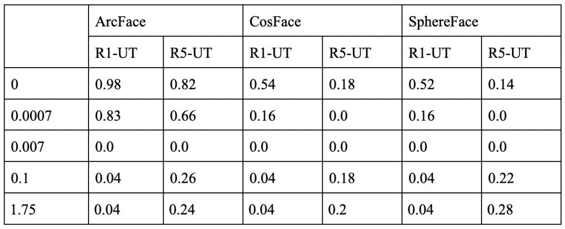
| | ArcFace | | CosFace | | SphereFace | |
|---|---|---|---|---|---|---|
| | R1-UT | R5-UT | R1-UT | R5-UT | R1-UT | R5-UT |
| ArcFace | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| CosFace | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| SphereFace | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |

Figure 6. Rank-1-UT and Rank-5-UT with gamma = 0.1 of using ArcFace, CosFace, and SphereFace models to build the mask against different models



| | ArcFace | | CosFace | | SphereFace | |
|---|---|---|---|---|---|---|
| | R1-T | R5-T | R1-T | R5-T | R1-T | R5-T |
| 0 | 0.98 | 1.0 | 0.54 | 0.92 | 0.5 | 0.9 |
| 0.0007 | 1.0 | 1.0 | 0.16 | 0.83 | 0.33 | 0.83 |
| 0.007 | 0.04 | 0.2 | 0.04 | 0.2 | 0.05 | 0.26 |
| 0.1 | 0.04 | 0.26 | 0.04 | 0.18 | 0.04 | 0.22 |
| 1.75 | 0.04 | 0.24 | 0.04 | 0.2 | 0.04 | 0.28 |

Figure 7. Rank-1-T and Rank-5-T with different gamma value of using ArcFace, models to build the mask against different models



| | ArcFace | | CosFace | | SphereFace | |
|---|---|---|---|---|---|---|
| | R1-UT | R5-UT | R1-UT | R5-UT | R1-UT | R5-UT |
| 0 | 0.98 | 0.82 | 0.54 | 0.18 | 0.52 | 0.14 |
| 0.0007 | 0.83 | 0.66 | 0.16 | 0.0 | 0.16 | 0.0 |
| 0.007 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 0.1 | 0.04 | 0.26 | 0.04 | 0.18 | 0.04 | 0.22 |
| 1.75 | 0.04 | 0.24 | 0.04 | 0.2 | 0.04 | 0.28 |

Figure 8. Rank-1-UT and Rank-5-UT with different gamma value of using ArcFace, models to build the mask against different models

## 8. Procedural / Data Challenges

### 8.1. Labeled Faces in the Wild (LFW) Dataset [5]

This dataset is a public benchmark for face verification, it's a database of face photographs designed for studying the problem of unconstrained face recognition. It contains more than 13,000 images of face collected from the web, and each face has been labeled with the name of the person pictured. Due to the time limit of google colab, while using ArcFace model to generate protected images, we limited to only 200 to 300 identities, and the same goes to Sphere model when generating the protected images. While for CosFace, we chose 500 identities to generate the protected images.

### 8.2. MegaFace [8]

The paper also mentioned that they used the megaface dataset to test their result. However, they didn't show any related results in the paper. Out intention was to use megaface to test this method. However, the official site requires certain access user name and password and we couldn't get those. Also, according to this [9], it shows that the megaface dataset has been retired and retracted and should not be used. As a result, we could not test the result of the method on the megaface dataset.

## 9. Findings of the Study

For the Targeted Identity-Protection Iterative Method, it has a hyperparameter gamma that can adjust the naturalness of the generated protected image. By adjusting the value of gamma, we found that it does have a large impact on the naturalness of the generated protected image. While gamma was set to 0.0, the generated protected image did a rather good job and obtained a good score from Rank-N-T and Rank-N-UT, but there would exists weird artifacts on the image and lead to bad user experience when viewing. Meanwhile, if the gamma was set to 0.1, the generated protected image would look very natural, and look identical with the probe image to human eyes, but it didn't pass the score test of Rank-N-T and Rank-N-UT. Hence, we think a good range for the gamma should be between 0 and 0.1.

Since we also used the PSNR and SSIM value to test the performance of the models, and they were computed between the probe image and the generated protected image. From these two values, we found that for all three models, while using it to generate the protected images, the PSNR and SSIM values were rather high even with gamma set to 0.0. And when gamma was set to 0.1, the PSNR value got even higher while the SSIM value didn't really change compared with gamma equal to 0.0. And just based on this, we can see that the value of gamma does have a large influence on the naturalness and quality of the generated protected images.

## 10. Discussion

For the process of reproducing, the easy part was that the authors already provided the code to us, so we didn't have to implement everything from scratch. Also, for the part of calculating PSNR and SSIM values between the probe images and generated protected images, the code was rather easy to implement since cv2 and skimage already have them implemented, and we just needed to call them.

The hard part was that for the code provided by the authors, we could only find the code that adds the adversarial identity mask to the images, and that's all. And for the original code, we had trouble running it in the beginning, since the versions of tensorflow and numpy the authors used were rather old, and there were many bugs in the beginning that we had to spend time going through and figure out the solution of each. And since we are running the code on colab as it has better hardware performance, there's a time limit so we can't choose many images to add the adversarial masks as the paper described. And whenever we make changes/modifications to the code, we had to re-run it and it's time consuming to run, each time we ran it, it could take at least 2 to 3 hours to complete and due to the characteristics of colab, it would disconnect from time to time and we have to re-run the whole thing again, which is very time-consuming and annoying. And after the provided code finished running, we have no idea how to evaluate the result of the code. And after going through the original paper for several times, we had to implement the evaluation methods ourselves based on the description the authors put in the paper.

However, there's one thing that we did find strange and don't understand. As gamma can adjust the naturalness of the generated protected images, it could also hurt the performance of the Targeted Identity-Protection Iterative Method, and too large of the gamma can make the protection useless. While in the paper, the authors stated the gamma can be increased up until it reaches 2.5, and the generated protected images has visible artifacts even when gamma equals 2.0. However, during our implementation and testing, we found that the generated protected images already look very natural to human eyes when gamma had been to to 0.1, and even at this value, the protection method was already ineffective in protecting the person's identity. And this was very strange to us, we have looked on the official github site, and we found out that another person also asked a similar question regarding the value of gamma, while the author never answered this question and it's still a confusion to us.

## 11. Conclusion

- The proposed method, Targeted Identity-Protection Iterative Method indeed can do a great job in protecting the true identity of the person in an image when the gamma has been set to 0.0.

- By increasing the gamma value of the model, the naturalness of the generated protected images can indeed increase and improve user's viewing experience, however, unlike what the author proposed in the paper, the gamma value can't be raised as high as 2.5, which is stated to be useful in the paper, and as soon as the gamma value could let users view the generated protected images in a comfortable way, the protection would go ineffective.

## References

[1] V. Chandrasekaran, C. Gao, B. Tang, K. Fawaz, S. Jha, and S. Banerjee. Face-off: Adversarial face obfuscation. *Microsoft., University of Wisconsin-Madison.*, 15 Dec, 2020. 2

[2] Y. Dong, Q. Fu, X. Yang, T. Pang, H. Su, Z. Xiao, and J. Zhu. Benchmarking adversarial robustness. *Dept. of Comp. Sci. and Tech., BNRist Center, State Key Lab for Intell. Tech. Sys., Institute for AI, THBI Lab, Tsinghua University, Beijing, 100084, China., Real AI.*, 26 Dec, 2019. 2

[3] Y. Dong, T. Pang, H. Su, and J. Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. *Dept. of Comp. Sci. and Tech., BNRist Center, State Key Lab for Intell. Tech. Sys., Institute for AI, THBI Lab, Tsinghua University, Beijing, 100084, China.*, 5 Apr, 2019. 2

[4] C. Dwork. Differential privacy: A survey of results. *Microsoft Research.*, Springer, 2008. 2

[5] T. Berg G. B. Huang, M. Matter and E. Learned-Miller. Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. 2018. 5

[6] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. *Universite de Montreal.*, 10 Jun, 2014. 2

[7] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *Google Inc., Mountain View, CA.*, 20 Mar, 2015. 2

[8] D. Miller I. Kemelmacher-Shlizerman, S. M. Seitz and E. Brossard. The megaface benchmark: 1 million faces for recognition at scale. *Dept. of Computer Science and Engineering, University of Washington.*, 2 Dec, 2015. 5

[9] Paperswithcode. https://paperswithcode.com/dataset/megaface. 5

[10] S. Shan, E. Wenger, J. Zhang, H. Li, H. Zheng, and B. Y. Zhao. Fawkes: Protrecting privacy against unauthorized deep learning models. *Computer Science, University of Chicago.*, 23 Jun, 2020. 2

[11] Q. Sun, L. Ma, S. J. Oh, L. V. Gool, B. Schiele, and M. Fritz. Natural and effrective obfuscation by head inpainting. *Max Planck Institute for Informatics, Saarland Informatics Campus., KU-Leuven/PSI, Toyota Motor Europe (TRACE)., ETH Zurich.*, 16 Mar, 2018. 2

[12] M. J. Wilber, V. Shmatikov, and S. Belongie. Can we still avoid automatic face detection? *IEEE Winter Conference on Applications of Computer Vision (WACV), pages 1-9.*, 2016. 2

[13] X. Yang, Y. Dong, T. Pang, H. Su, J. Zhu, Y. Chen, and H. Xue. Towards face encryption by generating adversarial identity masks. *Dept. of Comp. Sci. and Tech., Institute for AI, BNRist Center, Tsinghua-Bosch Joint ML Center., THBI Lab, Tsinghua University, Beijing, 100084, China., Pazhou Lab, Guangzhou, 510330, China., RealAI., Alibaba Group.*, 16 Aug, 2021. 2