.Image Cloaking

Mrinal Srivastava[#1], Chirag Gupta[*2], Devansh Rastogi[#3]

[#1,2,3]*Department of Computer Science and Engineering, Raj Kumar Goel Institute of Technology, Ghaziabad, Uttar Pradesh, India*

[1]mrinal.srivastava606@gmail.com

[2]chiragupta005@gmail.com

[3]devanshrastogi789@gmail.com

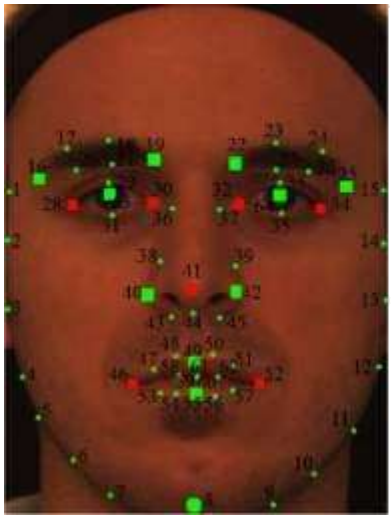
***Abstract*— Today's proliferation of powerful facial recognition systems poses a real threat to personal privacy. Ubiquitous facial recognition is a serious threat to privacy. The idea that the photos we share are being collected by companies to train algorithms that are sold commercially is worrying. Anyone can buy these tools, snap a photo of a stranger, and find out who they are in seconds. Regulations can and will help restrict the use of machine learning by public companies but will have negligible impact on private organisations, individuals, or even other nation states with similar goals. So, how do we protect ourselves against unauthorized third parties building facial recognition models that recognize us wherever we may go? This software uses Artificial Intelligence to subtly and almost imperceptibly alter one's photos in order to trick facial recognition systems. The way the software works is a bit complex. Running one's photos through this application does not make one invisible to any facial recognition exactly. Instead, the software makes subtle changes to one's photos so that any algorithm scanning those images in future sees one as a different person altogether. Essentially, cloaking on one's photos is like adding an invisible mask or filter to one's photos. This software creates an inaccurate image without significantly distorting the photo or by adding conspicuous patches or filters, which is used against unauthorised facial recognition models. This is achieved by adding imperceptible pixel-level changes, 'cloaks', to the image. For example, a user who wants to share content, specifically facial image, on social media or the public web, can add small, imperceptible alterations to their photos before uploading them. If collected by a third-party tracker and used to train a facial recognition model to recognise the user, these 'cloaked images' would produce functional models that consistently misidentify that user.**

***Keywords*— Artificial Intelligence, Computer Vision, OpenCV, Image Recognition, Face recognition, Image Cloaking**

INTRODUCTION

Today's proliferation of powerful facial recognition models poses a real threat to personal privacy. Facial recognition systems are scanning millions of citizens. By next year, almost all international travelers will be required to submit to facial recognition systems in airports. Perhaps more importantly, anyone with moderate resources can now canvas the Internet and build highly accurate facial recognition models of us without our knowledge or awareness. Opportunities for misuse of this technology are numerous and potentially disastrous. Anywhere we go, we can be identified at any time through street cameras, video doorbells, security cameras, and personal cell phones. Stalkers can find out our identity and social media profiles with a single snapshot . Stores can associate our precise in-store shopping behavior with online ads and browsing profiles . Identity thieves can easily identify (and perhaps gain access to) our personal accounts .We believe that private citizens need tools to protect themselves from being identified by unauthorized facial recognition models. Unfortunately, previous work in this space is sparse and limited in both practicality and efficacy. Some have proposed distorting images to make them unrecognizable and thus avoiding facial recognition . Others produce adversarial patches in the form of bright patterns printed on sweatshirts or signs, which prevent facial recognition algorithms from even registering their wearer as a person.

Fig. 1 Face Mapping with Face Landmarks



Primary landmarks		Secondary landmarks	
Number	Definition	Number	Definition
16	Left eyebrow outer corner	1	Left temple
19	Left eyebrow inner corner	8	Chin tip
22	Right eyebrow inner corner	2-7, 9-14	Cheek contours
25	Right eyebrow inner corner	15	Right temple
28	Left eye outer corner	16-19	Left eyebrow contours
30	Left eye inner corner	22-25	Right eyebrow corners
32	Right eye inner corner	29, 33	Upper eyelid centers
34	Right eye outer corner	31, 35	Lower eyelid centers
41	Nose tip	36, 37	Nose saddles
46	Left mouth corner	40, 42	Nose peaks (Nostrils)
52	Right mouth corner	38-40, 42-45	Nose contours
63,64	Eye centers	47-51,53-62	Mouth contours

Fig. 2 Face Landmarks with Designation

Literature Review

Literature Survey is mainly carried out in order to analyze the background of the current project which helps to find out flaws in the existing system and guides on which unsolved problems we can work out. So the following topics not only illustrate the background of the project but also uncover the problems and flaws which motivated us to propose solutions and work on this project.

Today's proliferation of powerful facial recognition systems poses a real threat to personal privacy. As Clearview.ai, an American facial recognition company, demonstrated, anyone can canvas the Internet for data and train highly accurate facial recognition models of individuals without their knowledge. Privacy advocates have considered the problem of protecting individuals from facial recognition systems, generally by making images difficult for a facial recognition model to recognize. Some rely on creating adversarial examples, inputs to the model designed to cause misclassification. These attacks have since been proven possible "in the wild," creating specially printed glasses that cause the wearer to be misidentified.

An alternative to evading models is to disrupt their training. This approach leverages "data poisoning attacks" against deep learning models. These attacks affect deep learning models by modifying the initial data used to train them, usually by adding a set of samples S and associated labels LS. Previous work has used data poisoning to induce unexpected behaviors in trained DNNs. DNN models are trained to identify and extract (often hidden) features in input data and use them to perform classification. Yet their ability to identify features is easily disrupted by data poisoning attacks during model training, where small perturbations on training data with a particular label can shift the model's view of what features uniquely identify. The model thinks it is successful, because it correctly recognizes its sample of (modified) images of the user. However, when unaltered images of the user, e.g. from a surveillance video, are fed into the model, the model does not detect the features it associates with the user. Instead, it identifies someone else as the person in the video. By simply modifying their online photos, the user successfully prevents unauthorized trackers and their DNN models from recognizing their true face.

All of these approaches share two limitations. First, they require the user to wear fairly obvious and conspicuous accessories (hats, glasses, sweaters) that are impractical for normal use. Second, in order to evade tracking, they require full and unrestricted access (white box access) to the precise model tracking them. Thus they are easily broken (and user privacy compromised) by any tracker that updates its model. Another line of work seeks to edit facial images so that human-like characteristics are preserved but facial recognition model accuracy is significantly reduced.

Recent work has shown that ML models can memorize (and subsequently leak) parts of their training data. This can be exploited to expose private details about members of the training dataset. These attacks have spurred a push towards differentially private model training, which uses techniques from the field of differential privacy to protect sensitive characteristics of training data. We note these techniques imply a trusted model trainer and are ineffective against an unauthorized model trainer.

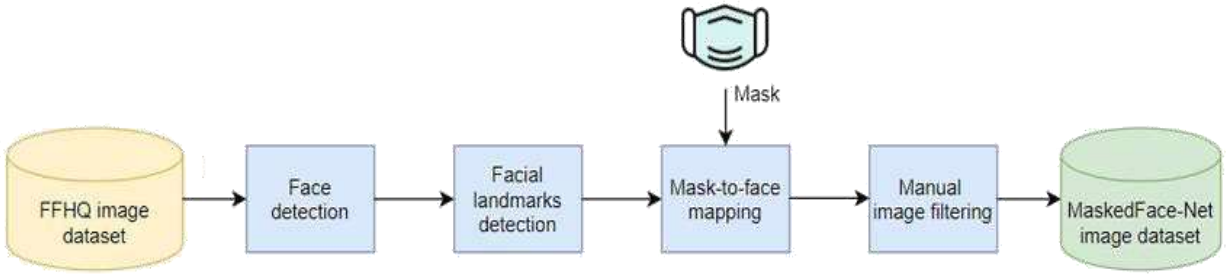


Fig. 3 An Example of using the Face Landmarks and adding a Face Mask to the face

We propose a software that helps individuals to inoculate their images against unauthorized facial recognition models at any time without significantly distorting their own photos, or wearing conspicuous patches. It achieves this by helping users add imperceptible pixel-level changes ("cloaks") to their own photos. The cloak effect is not easily detectable by humans or machines and will not cause errors in model training. However, when someone tries to identify you by presenting an unaltered, uncloaked image of you (e.g. a photo taken in public) to the model, the model will fail to recognize you. If collected by a third-party "tracker" and used to train a facial recognition model to recognize the user, these "cloaked" images would produce functional models that consistently misidentify them. The distortion or "cloaking" algorithm takes the user's photos and computes minimal perturbations that shift them significantly in the feature space of a facial recognition model (using real or synthetic images of a third party as a landmark)

Problem Identification and Challenges

In recent years, the images shared over social media, saved snapshots in mobile phones are more in number. Companies have been stalking the web for public photos associated with people's names that they can use to build enormous databases of faces and improve their facial recognition systems, in a sense the personal privacy is being lost. We are presented with two big questions:

1. *How to protect users from these kinds of facial recognition tools?*
2. *How do we determine what perturbation(as we call them 'cloaks') to apply to a person's face in photos?*

We need tools to protect ourselves from potential misuses of unauthorized facial recognition systems. Unfortunately, no practical or effective solution exists.

Although building facial recognition seems easy it is not as easy in the real world images that are being taken without any constraint. There are several challenges that are faced by the Facial Recognitions System are as follows:

1. *Illumination*: It changes the face appearance drastically, it is observed that the slight changes in lighting conditions cause a significant impact on its results.
2. *Pose*: Facial Recognition systems are highly sensitive to the pose, Which may result in faulty recognition or no recognition if the database is only trained on frontal face view.
3. *Facial Expressions*: Different expressions of the same individual are another significant factor that needs to be taken into account. Modern Recognizers can easily deal with it though.
4. *Low Resolution*: Training of recognizer must be done on a good resolution picture, otherwise the model will fail to extract features.
5. *Aging*: With increasing age, the human face features shape, lines, texture changes which are yet another challenge.

Problem Solution

We are developing a program which will integrate various modules. Module such as:

1. *Face Detection*
2. *Face Mapping using Face Landmarks*
3. *Grouping similar Faces together*
4. *Altering the face in the image*
5. *Showing the altered image*

as output Software Requirements:

1. *Windows/Linux/Mac OS*
2. *Python 3*
3. *Packages in Python:*
    a. *openCV*
    b. *numpy*
    c. *m*
    ediapipe

Hardware Requirements:

1. *Working PC/Laptop*

2. *Installed drivers*
3. *Webcam, if live feed required, (flash/LED for night feed)*

We propose a software that helps individuals to inoculate their images against unauthorized facial recognition models at any time without significantly distorting their own photos, or wearing conspicuous patches. It achieves this by helping users add imperceptible pixel-level changes ("cloaks") to their own photos. The cloak effect is not easily detectable by humans or machines and will not cause errors in model training. However, when someone tries to identify you by presenting an unaltered, uncloaked image of you (e.g. a photo taken in public) to the model, the model will fail to recognize you. If collected by a third-party "tracker" and used to train a facial recognition model to recognize the user, these "cloaked" images would produce functional models that consistently misidentify them. The distortion or "cloaking" algorithm takes the user's photos and computes minimal perturbations that shift them significantly in the feature space of a facial recognition model (using real or synthetic images of a third party as a landmark).

Any facial recognition model trained using these images of the user learns an altered set of "features" of what makes them look like them. When presented with a clean, uncloaked image of the user, e.g. photos from a camera phone or streetlight camera, the model finds no labels associated with the user in the feature space near the image, and classifies the photo to another label (identity) nearby in the feature space. An effective cloak would teach a face recognition model to associate a person's face with erroneous features that are quite different from real features defining that person. Intuitively, the more dissimilar or distinct these erroneous features are from the real person, the less likely the model will be able to recognize the real person.

Cloaking is highly effective when users share a feature extractor with the tracker; efficacy could drop when feature extractors are different, but can be restored to near perfection by making the user's feature extractor robust (via adversarial training); and, similarly, cloaks generated on robust feature extractors work well even when trackers train models from scratch.

Like most privacy enhancing tools and technologies, this software can also be used by malicious bad actors. For example, criminals could use it to hide their identity from agencies that rely on third-party facial recognition systems. We believe our software will have the biggest impact on those using public images to build unauthorized facial recognition models and less so on agencies with legal access to facial images such as federal agencies or law enforcement.Protecting content using cloaks faces the inherent challenge of being future-proof, since any technique we use to cloak images today might be overcome by a workaround in some future date, which would render previously protected images vulnerable. While we are under no illusion that this proposed system is itself future-proof, we believe it is an important and necessary first step in the development of user centric privacy tools to resist unauthorized machine learning models. This project can be further enhanced to provide greater flexibility and performance with certain modifications whenever necessary.

The basic idea behind this system is that any image containing a person's face will be altered a little to stop any facial recognition softwares in identifying that person . Secondly, even if that tweaked image is being used to train facial recognition softwares, during that simulation, identifying a person will be marked as success, since all the images of that person in the dataset are altered. When facial recognition software will be used, let's say in security cameras, it won't be able to identify that person or will find a similar looking person but not a specific one.

Face Detection: This might sound similar with facial recognition but actually, they are not the same, rather face recognition implies the system is able to detect faces in the given image/video/live feed. Face Detection is the process of detecting faces. The program doesn't do anything more than finding the faces. But in face recognition, the program first uses face detection to find faces, and then it starts analyzing those faces to identify the person. So, face recognition is more informational than just detecting them. The steps involved in face recognition models are:

1. *Face Detection:* Locate faces and draw bounding boxes around faces and keep the coordinates of bounding boxes
2. *Face Alignments:* Normalize the faces to be consistent with the training database.
3. *Feature Extraction:* Extract features of faces that will be used for training and recognition tasks.
4. *Face Recognition:* Matching of the face against one or more known faces in a prepared database.

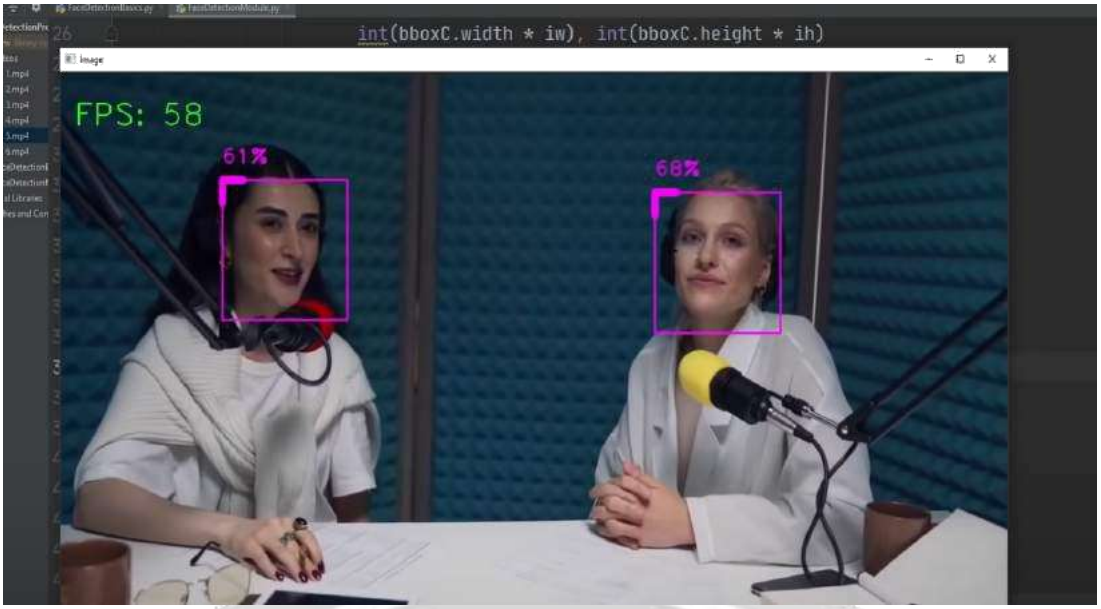


Fig. 4 Face Detection

Face Mapping using Face Landmarks: Identifying faces in photos or videos is very cool, but this isn't enough information to create powerful applications, we need more information about the person's face, like position, whether the mouth is opened or closed, whether the eyes are opened, closed, looking up and etc. In this article I will present to you (in a quick and objective way) the Dlib, a library capable of giving you 68 points (landmarks) of the face.

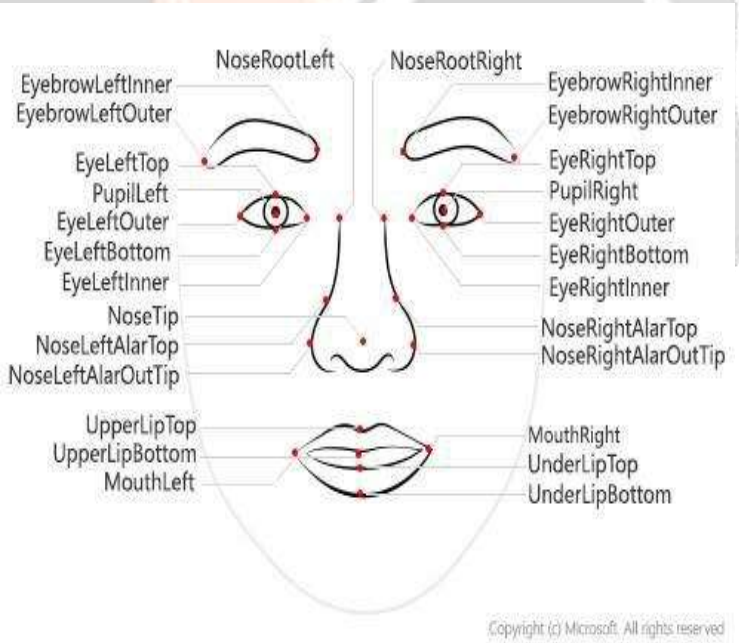


Fig. 5 Face Landmarks

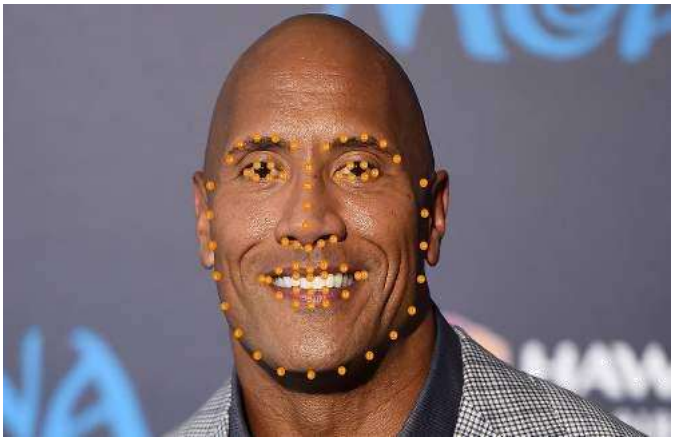


Fig. 6 Face Mapping with Facial Landmarks

Typical applications where face landmarking plays a prominent role are facial expression analysis, face animation, 3D face reconstruction, registration, feature-based face recognition, verification and face tracking, head gesture understanding. Subsequent applications of landmarking could be for anonymization of facial identity in digital photos, image editing software tailored for faces, lip reading, sign language interpretation etc. Below we give more details on four these landmark dependent tasks:The steps involved in face recognition models are:

1. *Expression understanding:* Facial expressions form a visual channel for emotions and nonverbal messages, and they have a role in supporting verbal communication. The spatial configuration and temporal dynamics of landmarks provide a viable way to analyze facial expression and to objectively describe head gestures and facial expressions.
2. *Face Recognition:* Face recognition schemes typically locate the eye region and then extract holistic features from the windows centered on various regions of interest. The located landmark coordinates also give rise to a number of geometric properties such as distances and angles between them.
3. *Face Tracking:* Most face tracking algorithms benefit from tracked landmark sequences. In the model-based group of methods, a face graph model is fitted to 60-80 facial landmarks. Face tracking is realized then by letting the model graph evolve according to face shape parameters, facial components and geometrical relations between them. The alternative tracking approach is model-free and is principally based on motion estimation. In these methods, the motion is estimated at and around the landmarks vis-à-vis some reference frame. The advantage of landmark-based tracking is that both the head motion and the facial deformations are jointly estimated. This enables us to detect and classify head gestures, head and facial emblems, interpret certain mental states as well as to extract clues for head and face animation.
4. *Face Registration:* Face registration is the single most important factor affecting face recognition performance. Other applications of landmarking involve building of 3D face models from stereo, from multiple images or from video sequences where landmark points are used to establish point-to-point correspondences. A face can be transformed into those of other individuals (interpersonal) or into different expressions of the same individual (intra-personal, e.g., a neutral face to a smiling face). In summary, face landmarking is a prerequisite for face normalization and registration whether in 2D or 3D.

Fig. 7 Face Mapping using Face Landmarks

Grouping similar Faces together: This module is self explanatory on what the next step of our process is. The important thing to keep in mind is what we are storing as a privacy perspective. Unlike others, who retain the exact faces of people in their databases, we are not doing that. Instead we are updating our database with the demographic map of each individual face.

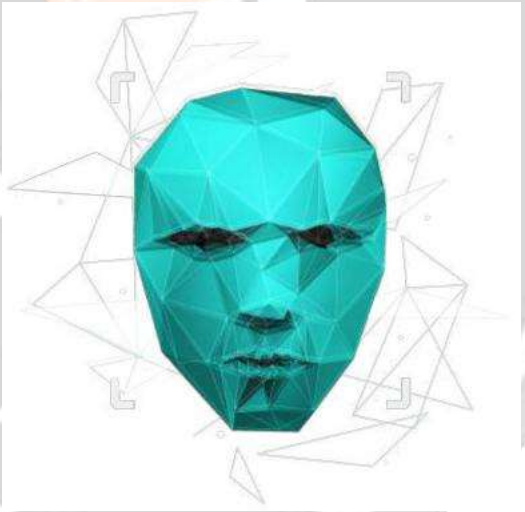


Fig. 8 Demographic structure of a 3D face made using the Face Landmarks

Altering the face in the image: Face altering refers to the image processing technology of the automatic fusion of two or more different faces into one face, which is widely used in fields of video synthesis, privacy protection, picture enhancement, and entertainment applications. For example, when we want to share some of the interesting things on social networks, we can use the face synthesis technique which can be regarded as a fusion of facial features and details to change our appearances appropriately without privacy leaks. As another type of face fusion, face swapping combines some parts of one person's face with other parts of the other's face to form a new face image. For instance, in the application of virtual hairstyle visualization, the client's facial area can be fused with the hair areas of the model images to form new photos, so that customers can virtually browse their own figures with different hairstyles. This paper focuses on the face swapping problem of virtual browsing applications for hairstyle and dressing.

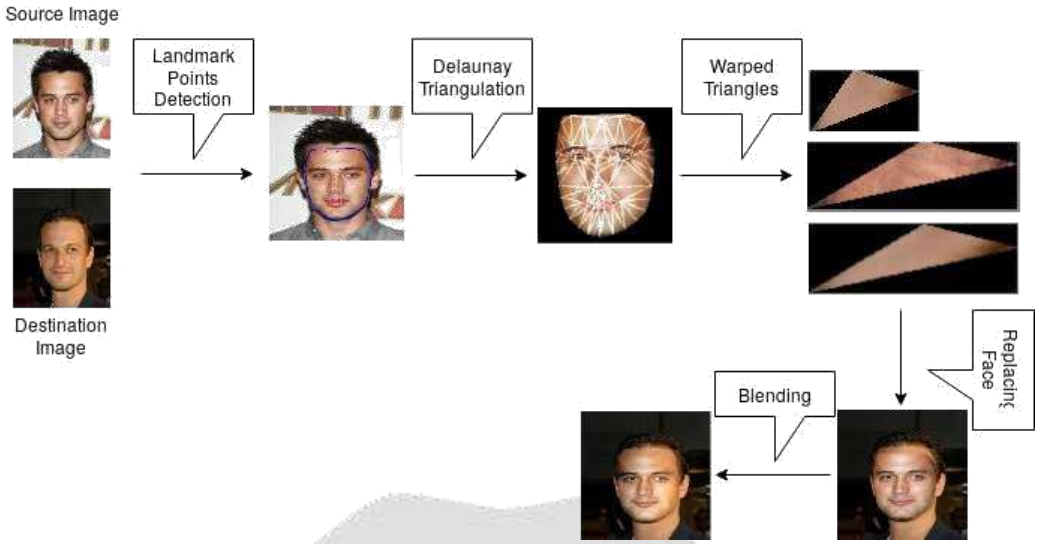


Fig. 9 Swapping face of similar looking people

Conclusion

Cloaking an image would make the personal data less accessible or usable by anyone(specially big-tech companies). It is a protection against unwanted facial recognition accessible to each individual. This would help each individual from being identified by anyone(unwanted) as it will keep one and their photos unidentifiable. Since the user is unidentifiable, it also keeps the user away from cyber attacks like 'phishing'. Adding this layer is pretty easy considering adding noise or blur or any other kind of filter since it works on artificial intelligence. Also pretty much, there are no noticeable changes made to the image.

We present to protect individuals from recognition by unauthorized and unaccountable facial recognition systems.Our approach applies small, carefully computed perturbations to cloak images, so that they are shifted substantially in a recognition model's feature representation space, all while avoiding visible changes. Cloaking is highly effective when users share a feature extractor with the tracker; efficacy could drop when feature extractors are different, but can be restored to near perfection by making the user's feature extractor robust (via adversarial training); and, similarly, cloaks generated on robust feature extractors work well even when trackers train models from scratch.

REFERENCES

[1] http://apodeline.free.fr/DOC/libjpeg/libjpeg-3. html. Using the IJG JPEG library: Advanced features
[2] https://aws.amazon.com/rekognition/. Amazon Rekognition Face Verification API
[3] https://azure.microsoft.com/en-us/services/ cognitive-services/face/. Microsoft Azure Face API
[4] https://www.faceplusplus.com/face-searching/. Face++ Face Searching API
[5] http://vision.seas.harvard.edu/pubfig83/. PubFig83: A resource for studying face recognition in personal photo collections
[6] ABADI, M., CHU, A., GOODFELLOW, I., MCMAHAN, H. B., MIRONOV, I., TALWAR, K., AND ZHANG, L. Deep learning with differential privacy. In Proc. of CCS (2016)
[7] CAO, Q., SHEN, L., XIE, W., PARKHI, O. M., AND ZISSERMAN, A. VGGFace2: A dataset for recognising faces across pose and age. In Proc. of IEEE FG (2018)
[8] CARLINI, N., AND WAGNER, D. Adversarial examples are not easily detected: Bypassing ten detection methods. In Proc. of AISec (2017)
[9] CARLINI, N., AND WAGNER, D. Towards evaluating the robustness of neural networks. In Proc. of IEEE S&P (2017)
[10] CARLINI, N., AND WAGNER, D. Towards evaluating the robustness of neural networks. In Proc. of IEEE S&P (2017)
[11] https://www.innovationmerge.com/2020/08/15/Image-cloaking-for-Personal-Privacy-in-Social-Media
[12] https://jivp-eurasipjournals.springeropen.com/articles/10.1186/1687-5281-2013-13