Face Encryption

SDS3386 Data Science Lab

Fall 2022

Team TensorOverflow

Xiang Li 300056427
Jiaxun Gao 300063462
Bowen Zeng 300115382

Course Coordinator: Tanya Schmah

November 30, 2022

University of Ottawa

uOttawa

Contents

1 Introduction

Recent research (Smith, Szongott, Henne, & von Voigt, 2012) has verified that there are recognized privacy dangers related to the massive amounts of user-generated information published on public social media platforms. Internet behemoths like Google and Facebook have been secretly mining this data without the users' awareness or permission (Esteve, 2017). When personal photos of people's faces are uploaded, this creates a severe privacy risk for those depicted. Companies may use these photographs to train machine learning algorithms, which could compromise users' personal information. **What measures can we take to ensure that individuals' privacy is maintained when exchanging photographs of their faces?**

We present a method that safeguards users' privacy when exchanging facial images by utilizing adversarial attacks and facial recognition algorithms. To accomplish this, we first use a facial recognition model to identify individuals in a picture, and then we encrypt their likenesses using noise produced by an adversarial attacking model. While the resulting image may have high visual similarities between human eyes, facial recognition software will fail to re-identify the right person in it. Such a facial encryption process can help people feel more comfortable sharing photographs online while maintaining their privacy.

2 Dataset

2.1 CelebA

CelebA is a large-scale face attributes dataset with more than 200K celebrity images, each with 40 attribute annotations. The images in this dataset cover large pose variations and background clutter. CelebA has large diversities, large quantities, and rich annotations, including **5,000 celebrity identities**, **202,599 face images**, and **40 binary attributes** annotations per image. The dataset can be employed as the training and test sets for the following computer vision tasks: face attribute recognition, face detection, landmark (or facial part) localization, and face editing/synthesis.

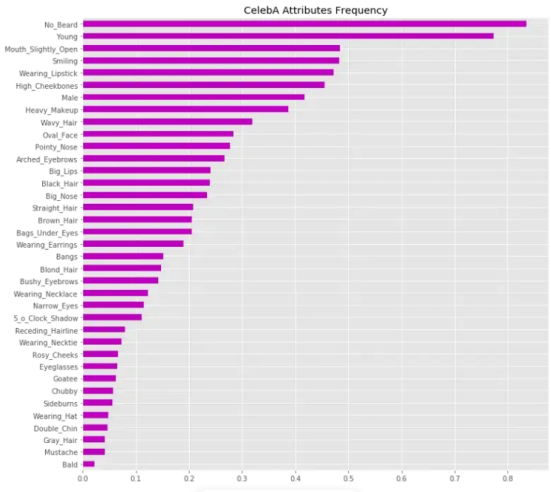
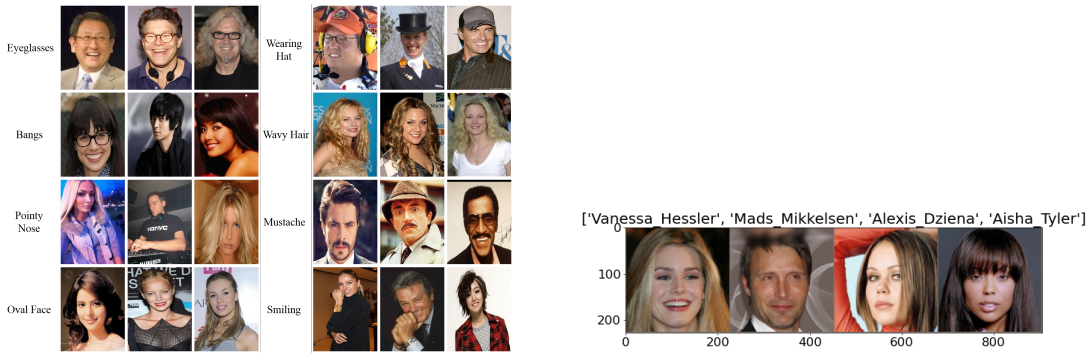


Figure 1: CelebA dataset distribution

In this project, we utilized a subset of CelebA as the face recognition model's dataset. The dataset includes 5696 images of 307 different people, with 4429 images used for training and 1267 images used for testing.



(a) CelebA dataset covers large pose variations *(b) CelebA in face recognition*

Figure 2: An excerpt from CelebA dataset

2.2 Private Dataset

For a better presentation, we also constructed a private dataset consisting 121 images of all team members. The photos were cropped from 3 videos, which were uniformly sampled to 40 frames each to vary the facial expression. In each frame, the face image is detected by package Mediapipe and then stored in the local device. [1]
The private dataset and CelebA dataset together form the training and testing dataset of our face recognition model.

3 Methodology

This section will present the primary methodologies we used in our work. First, a data transformation method will be applied to the training data. After the pre-processing, we used ResNet-18 as the backbone model for the facial recognition program, and adversarial attack methods will then be applied to this model to generate perturbations.

3.1 Data Transformation

The first step in the data transformation process was to resize all of the images to a uniform size of 224x224 pixels. This was done to ensure the uniform input dimension and all of the images could be processed by the model consistently.
In order to increase the diversity of the dataset and improve the performance of the face recognition model, data augmentation was also applied to the resized images. The following augmentation techniques were applied randomly: [2]

- Random rotation of up to 30 degrees
- Random horizontal flip
- Random zoom of up to 10%

3.2 Face Recognition

3.2.1 ResNet-18

ResNet-18 is a convolutional neural network (CNN) model developed in 2015 by Microsoft Research Asia. The architecture can be illustrated as figure 3 (Ramzan et al., 2019). It is a deep learning model that has

[1] This process is done in Data_generation.ipynb.ipynb.
[2] This process is a part of simple_model.ipynb.

been trained on a large dataset and is capable of learning to recognize patterns and features in images. Initially, ResNet-18 was trained on the ImageNet dataset, which contains 1.2 million images with 1000 classes. However, in subsequent research, ResNet-18 has been demonstrated to perform well on a wide range of image classification tasks. It has achieved state-of-the-art performance on several benchmarks and has been widely used in various applications. Some tasks may have better choices, but it is a solid general-purpose model worth considering in multiple situations.

Another benefit of ResNet-18 is that it is relatively lightweight, making it suitable for many mobile devices, including modern laptops and smartphones. It has 18 layers, which is relatively shallow compared to other CNN models, which can have hundreds or thousands of layers. ResNet-18 is thus more efficient and easier to train than deeper models.

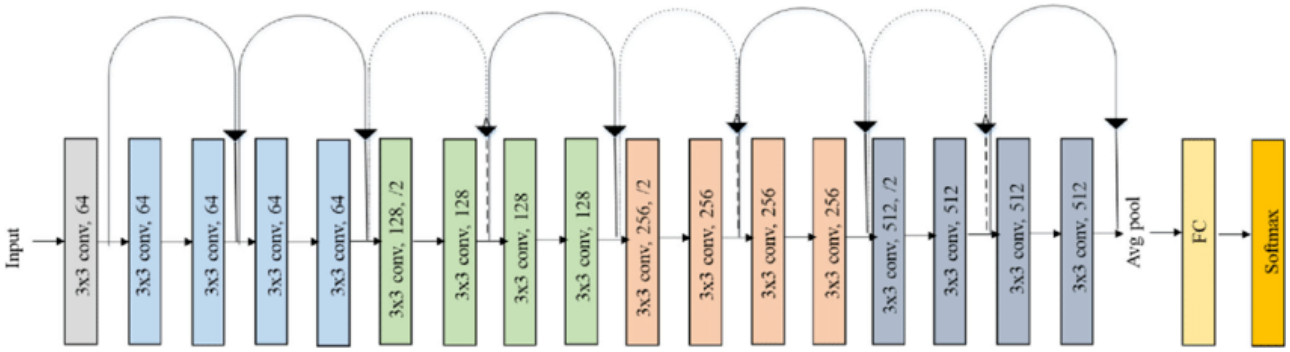


Figure 3: ResNet-18 Architecture

3.2.2 Face recognition system implementation

Before generating perturbation images from users' photos, we need to build a facial recognition system to serve as the attacking target. Before we feed the facial images into ResNet, a critical step is to crop and extract the facial area from the input picture. To achieve this goal, we used MediaPipe to obtain the corner coordinates of the facial area.

Next, the cropped face area is fed to a ResNet-18 model. We applied transfer learning techniques to speed up the training process - we obtained the initial weight of the neural network from a pre-training task conducted on ImageNet. Then, we fine-tuned the ResNet model on our training set. Experimental results show that the fine-tuned model can effectively identify the facial identities in the test set.
Here is a demonstration of our face-recognition system.

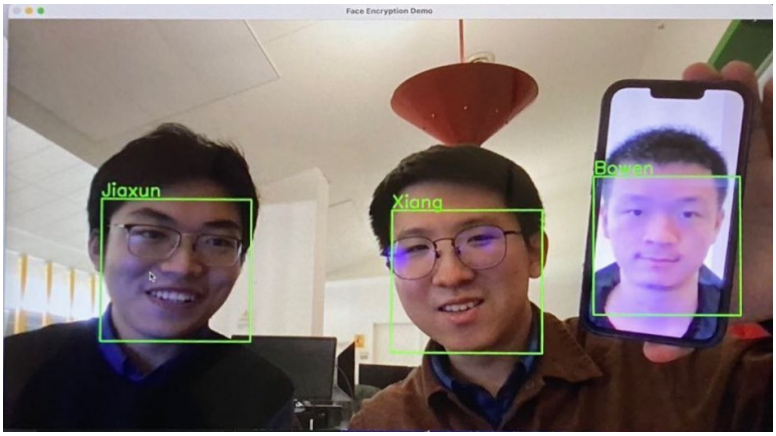


Figure 4: Face recognition system demo

3.2.3 Performance

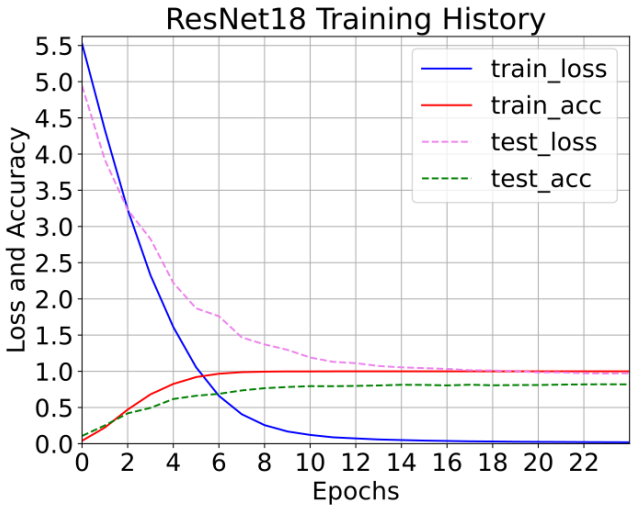


Figure 5: Resnet18 performance

Figure 5 demonstrates that with 25 epochs, the face recognition model Resnet-18 achieves 99.98% accuracy on the training set and 81.93% accuracy on the test set. We halted training at this point because any additional epoch would result in an overfitting problem.

3.3 Adversarial Attacks

Adversarial attacking is a technique that can be used to fool machine learning models. It is a type of attack that aims to change the input data in a way that the model will misclassify it. The adversarial attacking technique is based on the fact that machine learning models are vulnerable to small perturbations in the input data. The perturbations are usually imperceptible to the human eye, but they can cause the model to misclassify the input data. (Goodfellow, 2020)

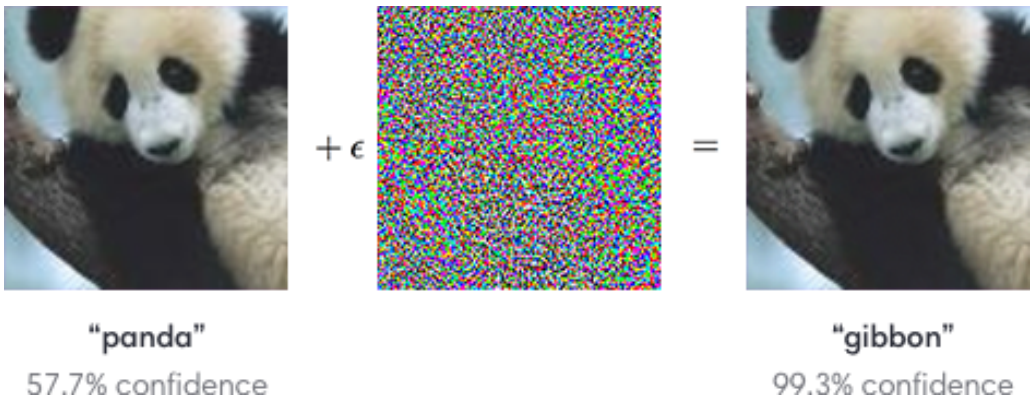


Figure 6: An adversarial input, overlaid on a typical image, can cause a classifier to miscategorize a panda as a gibbon.

3.3.1 PGD Attack

The PGD attack is a white-box adversarial attack which means the attacker knows everything about the model, including gradients, model parameters and even the raining data and training procedure (Madry, Makelov, Schmidt, Tsipras, & Vladu, 2017).

In this project, we use the gradient of the loss function to generate adversarial examples. The attack is iterative and uses a step size to determine the size of the perturbation. The attack is also constrained by a maximum perturbation size.

3.3.2 Non-Targeted Attack

In a non-targeted attack, the goal is to generate an adversarial example that is misclassified by the target model A. Which means we are **maximizing** the loss function with respect to the target class A. The attack is as follows:

- Input: `face image, true label, face recognition model`
- Initialize: δ = random noise
- For each iteration: Modify the δ with step size according to gradient such that:

  `Difference(Model(Face Image + Delta), True Label)` is maximum
- Return δ

where `Model(Face Image + Delta)` is the prediction of perturbed data.

3.3.3 Targeted Attack

In a targeted attack, the goal is to generate an adversarial example that is misclassified A by the target model and classified as a specific class B. Which means we are **minimizing** the loss function with respect to the target class B. The attack is as follows:

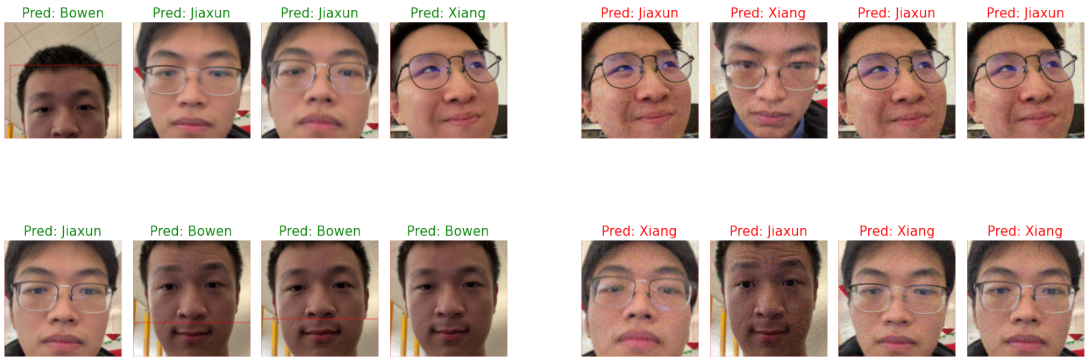
- Input: `face image, target label, face recognition model`
- Initialize: δ = random noise
- For each iteration: Modify the δ with step size according to gradient such that:

  `Difference(Model(Face Image + Delta), Target Label)` is minimum
- Return δ

where `Model(Face Image + Delta)` is the prediction of perturbed data.

3.3.4 Real-Time Application

Using the method described in the preceding section, we developed a program to execute PGD attacks through the camera in real time using the method described in the previous section. Before feeding them into the machine learning model, the application would capture images from the camera and then subject them to PGD perturbations. The output of the model would be monitored and the probability of the prediction would be displayed on the screen. This method allows for the testing and evaluation of machine learning models' resistance to PGD attacks in real time.



(a) Prediction without attack *(b) Prediction with non-targeted attack*

Figure 7: Example of a non-targeted attack

4 Result

We can generate an adversarial example using the method described in the preceding steps. In the figure 7, the image on the left demonstrates that Resnet-18 correctly predicted the original photos. After adding the PGD model's generated noise. The image on the right depicts Resnet-18 classifying the adversarial example as a different category. We also generated perturbations on the test set with both targeted and non-targeted attack. Experimental results show that with either adversarial attack method, we achieved less than 5 percent accuracy, demonstrating the effectiveness of our adversarial attack. Thus, we can confidently say that our facial encryption can efficiently protect the users' privacy.

5 Conclusion and Ethical Considerations

In this work, we show how adversarial attacks can safeguard the privacy of facial image sharing without harming image quality. The trained facial recognition system can no longer correctly identify the user's face by applying perturbations to the input images.

This is an important consideration when sharing photos online, as there are often concerns regarding personal privacy and the possibility of malicious actors gaining access to sensitive information.

The malicious application of the targeted-attack algorithm is at the heart of the project's ethical concerns. In particular, the model has been criticized since it might be used to reassign a person's images to a different identity, which could have dire ramifications for the wronged person. The person mistakenly identified may suffer damage to their reputation, finances, or both due to this malicious use. It is crucial to take precautions to prevent this kind of abuse and to have enough safeguards to protect the privacy and security of persons whose identities may be utilized to train the model.

6 Future Work

There are multiple possible future directions for this project's development. One option is to explore using more realistic attack scenarios, such as black box attacks. In our work, the adversarial attacking process still uses the target network gradient computed in the forward pass; in real-day scenarios, the gradient functions are typically invisible to users. Boundary Attack is one potential candidate for black box attacks. However the major problem with Boundary Attack is it requires querying the target model many times until the perturbation noise is fully generated. During the implementation of this report, we came up with another adversarial attack framework based on contrastive learning without querying the

target model. We will continue to work on this topic and report our progress.

Developing a server-client application that processes photos on a server rather than on the user's device is another avenue for future research. This could mitigate some of the ethical concerns associated with the malicious use of the targeted-attack application, as it would permit greater control and oversight over the model's application. Another advantage of using server-client architecture is that the client does not have to handle any computational-heavy tasks or large-scale data processing like encrypting high-resolution images, allowing it to run smoothly on a variety of devices with limited resources.

A Contributions

Xiang and Jiaxun took the lead in the modelling and live demo.
Bowen took the lead in the report.

References

Esteve, A. (2017, 03). The business of personal data: Google, Facebook, and privacy issues in the EU and the USA. *International Data Privacy Law*, *7*(1), 36-47. Retrieved from https://doi.org/10.1093/idpl/ipw026 doi: 10.1093/idpl/ipw026

Goodfellow, I. (2020, Oct). *Attacking machine learning with adversarial examples.* OpenAI. Retrieved from https://openai.com/blog/adversarial-example-research/

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

Ramzan, F., Khan, M. U., Rehmat, A., Iqbal, S., Saba, T., Rehman, A., & Mehmood, Z. (2019, 12). A deep learning approach for automated diagnosis and multi-class classification of alzheimer's disease stages using resting-state fmri and residual neural networks. *Journal of Medical Systems*, *44*. doi: 10.1007/s10916-019-1475-2

Smith, M., Szongott, C., Henne, B., & von Voigt, G. (2012). Big data privacy issues in public social media. In *2012 6th ieee international conference on digital ecosystems and technologies (dest)* (p. 1-6). doi: 10.1109/DEST.2012.6227909