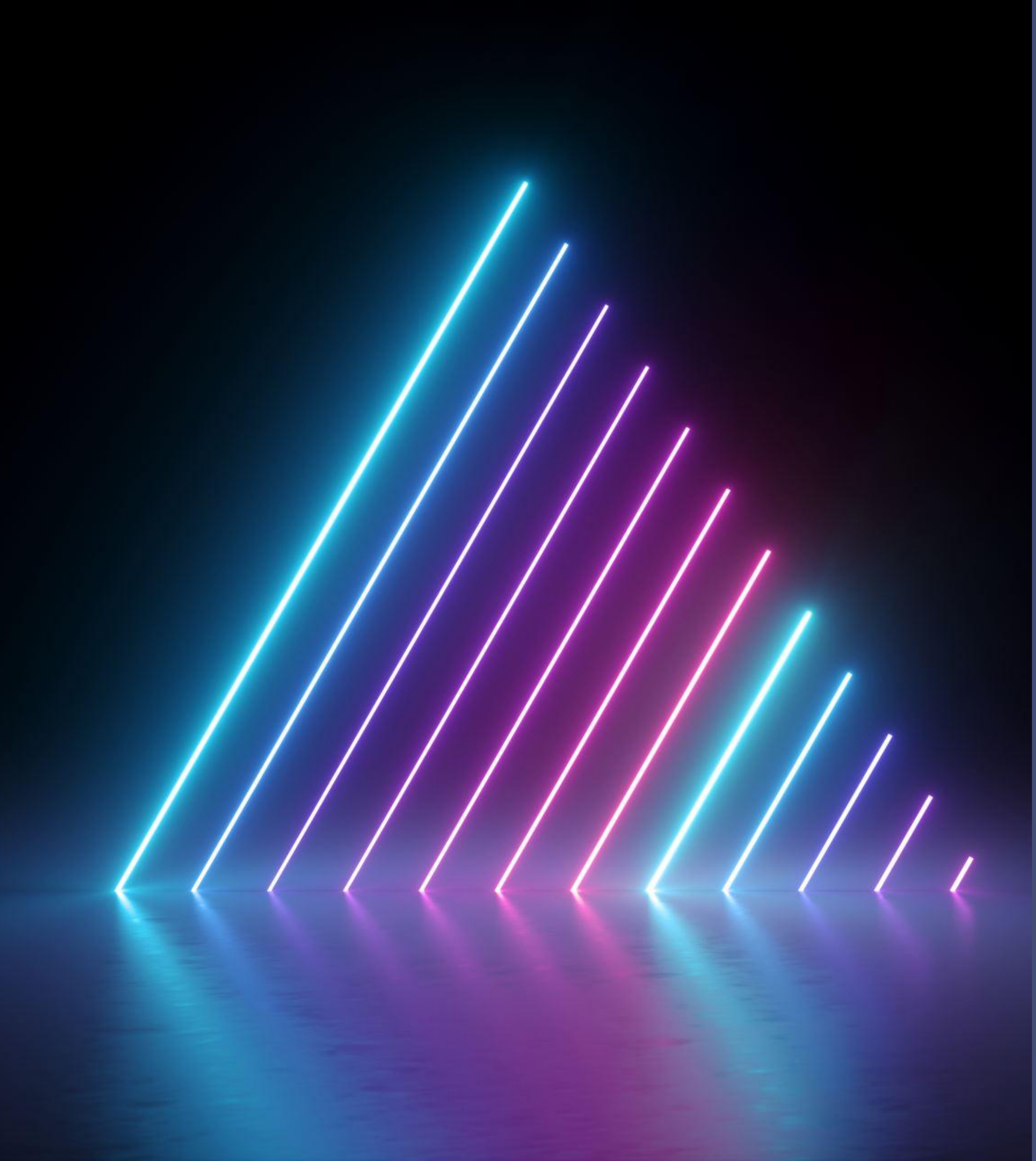


FACE ENCRYPTION

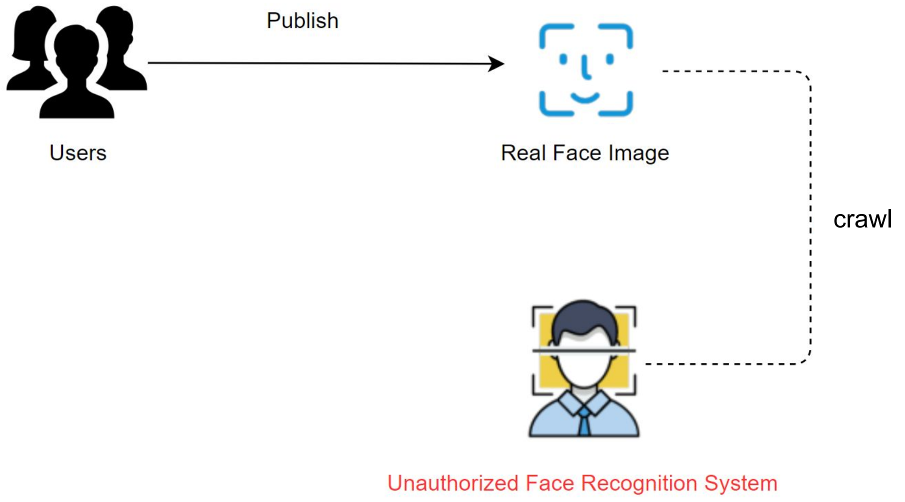
Now you don't.

Team TensorOverflow
Jiaxun Gao 300063462
Bowen Zeng 300115382
Xiang Li 300056427

Recap

How can we protect privacy when sharing facial images?

- Privacy leakage on social media
- Personal identity stealing by 3rd parties

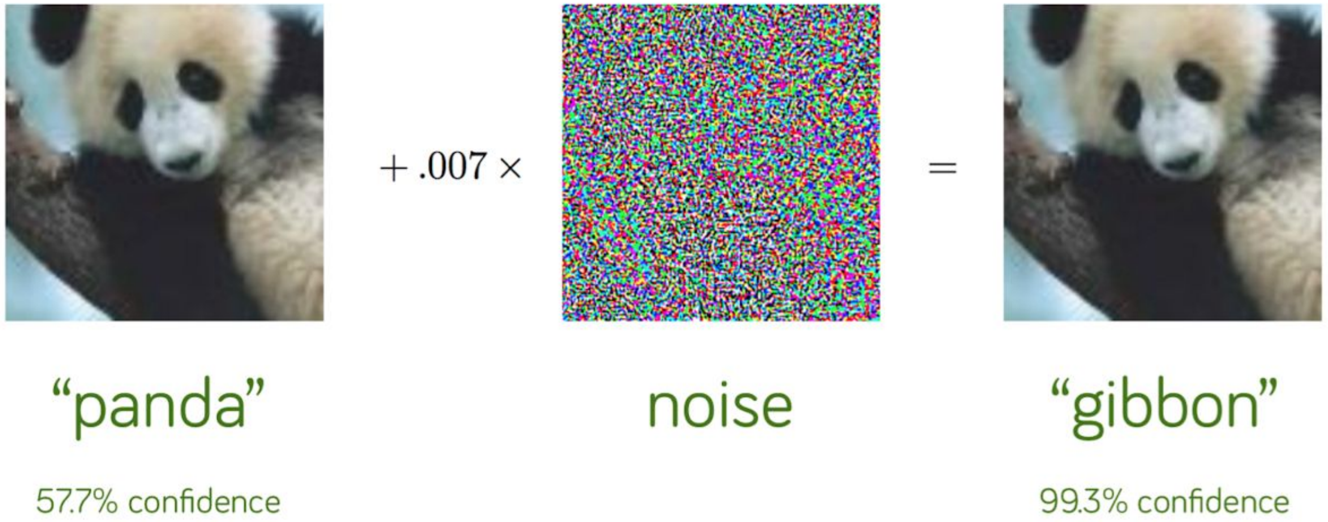


How can we protect privacy when sharing facial images?

A face encryption system that can:

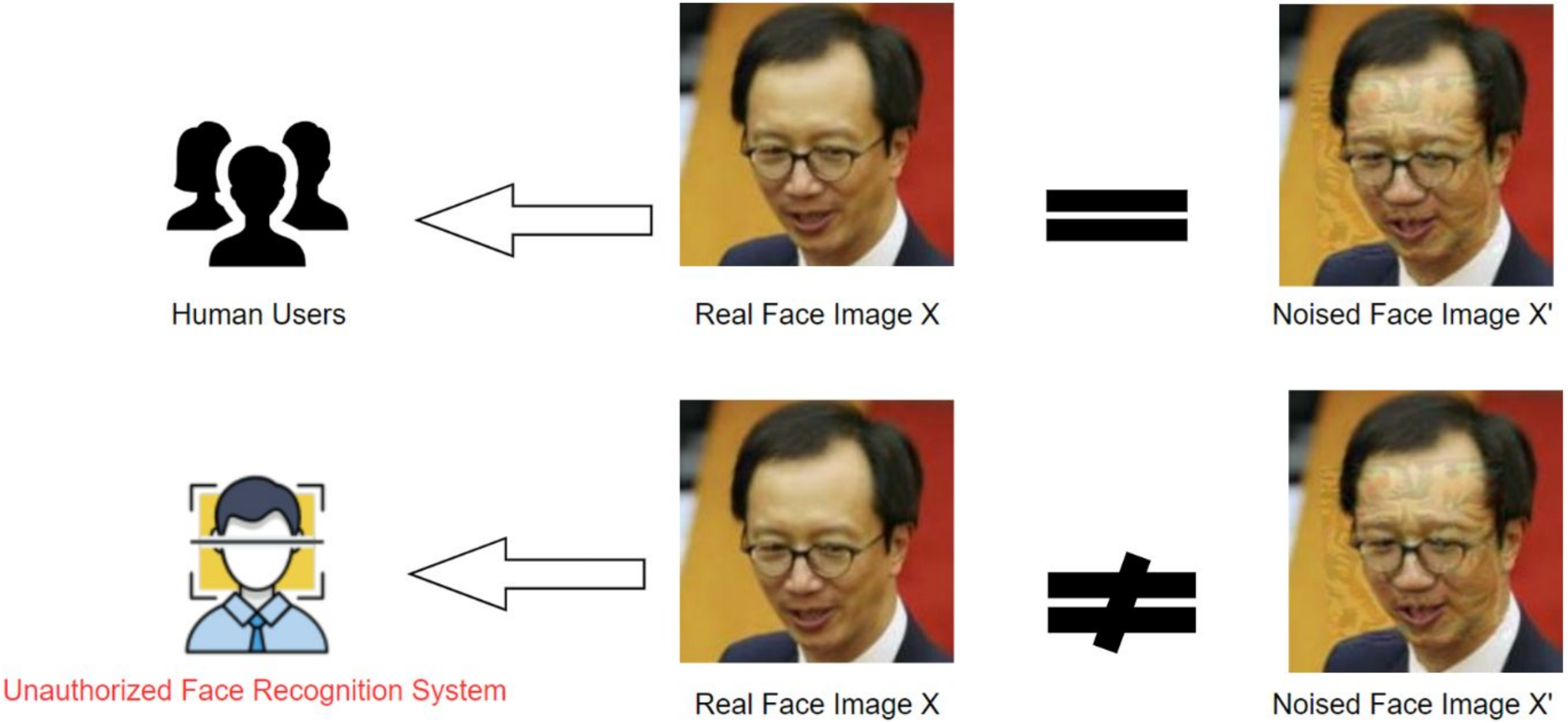
- Fool the SOTA facial recognition algorithms to protect personal privacy
- Doesn’t harm UX
- Lightweight enough to be able to run on phone/laptop

ADVERSARIAL ATTACKING



adversarial
example

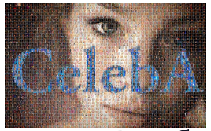
Figure: the picture is taken from (Goodfellow et al).


Human Users
Real Face Image X
Noised Face Image X'
Unauthorized Face Recognition System
Real Face Image X
Noised Face Image X'

Methodology

- Dataset
  - Public available dataset
  - Private dataset
- Backbone Model: ResNet-18
- Adversarial attacking
  - Targeted Attack
  - Non-targeted attack

DATASET: CelebA

- 

dataset
  - Reflect real day scenario on social media
  - Widely used in facial recognition projects
  - 307 identities

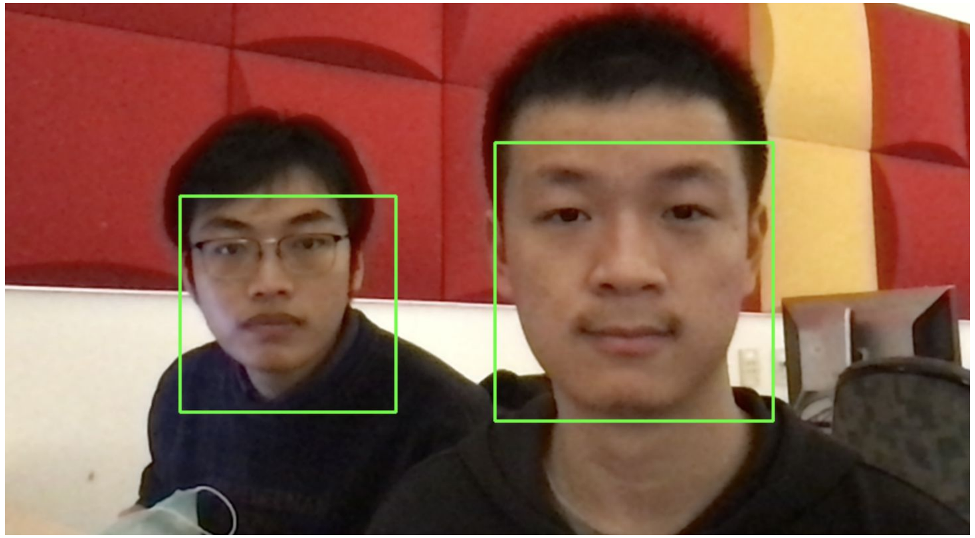
Sample Images (an excerpt from the data)

DATASET: Private

40 custom pictures per team member

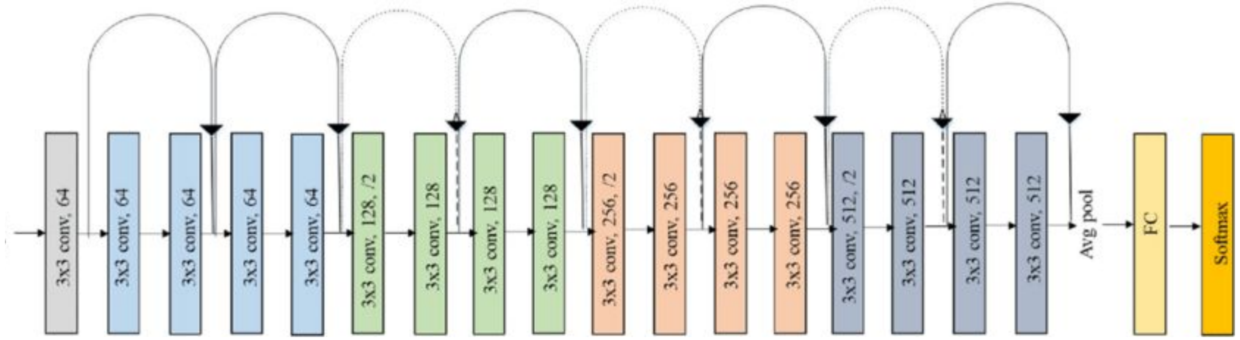
- 30 in training set
- 10 in testing set
- MediaPipe to extract facial images

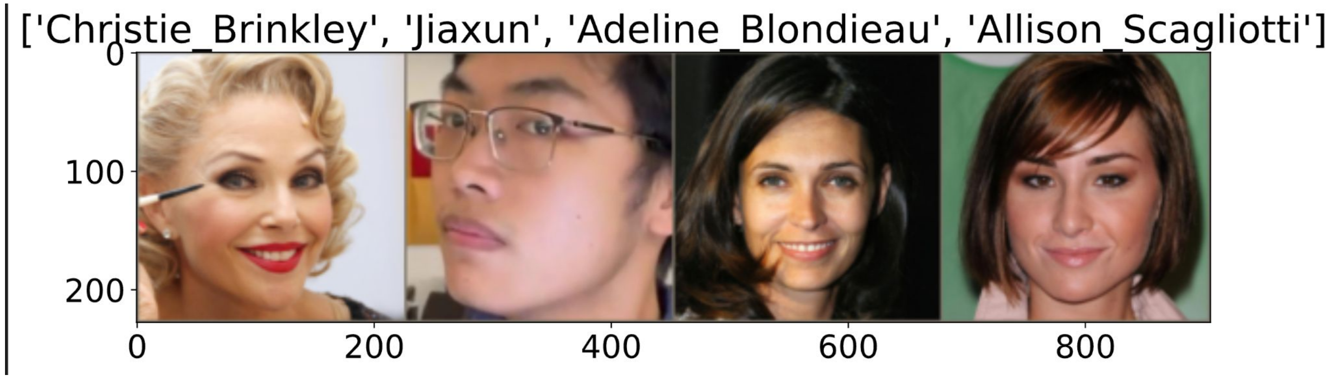
Bowen Zeng Xiang Li Jiaxun Gao



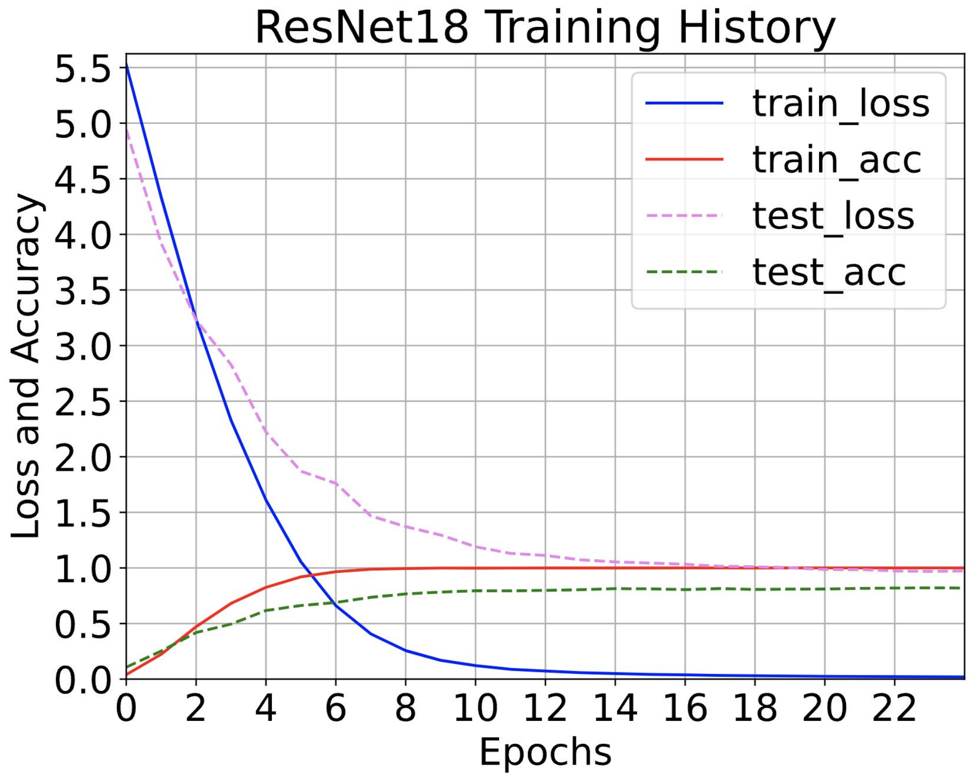
Bounding boxes generated by MediaPipe

Backbone Model: ResNet-18





Backbone Model: ResNet-18

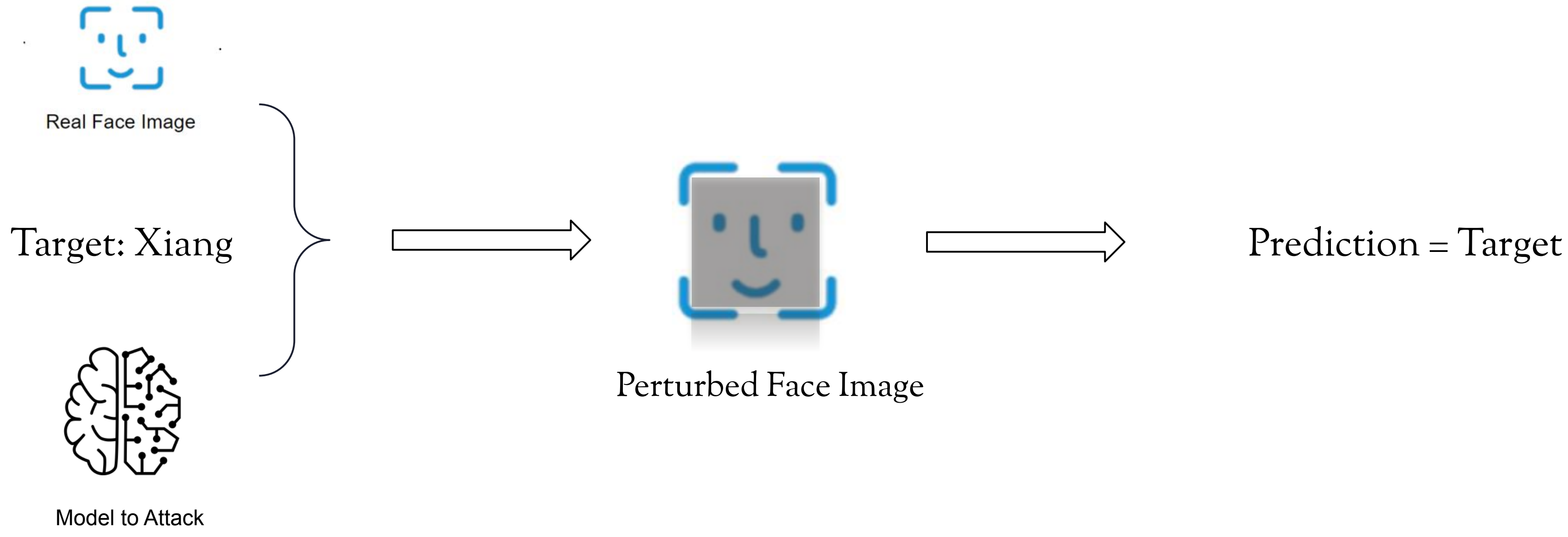


Methodology:
Projected Gradient Descent(PGD)

- Target Attack
  Fool the model with a pre-set target

- Non-Target Attack
  Fool the model with any other label

Target PGD:


Real Face Image
Target: Xiang
Model to Attack
Perturbed Face Image
Prediction = Target

Target PGD:

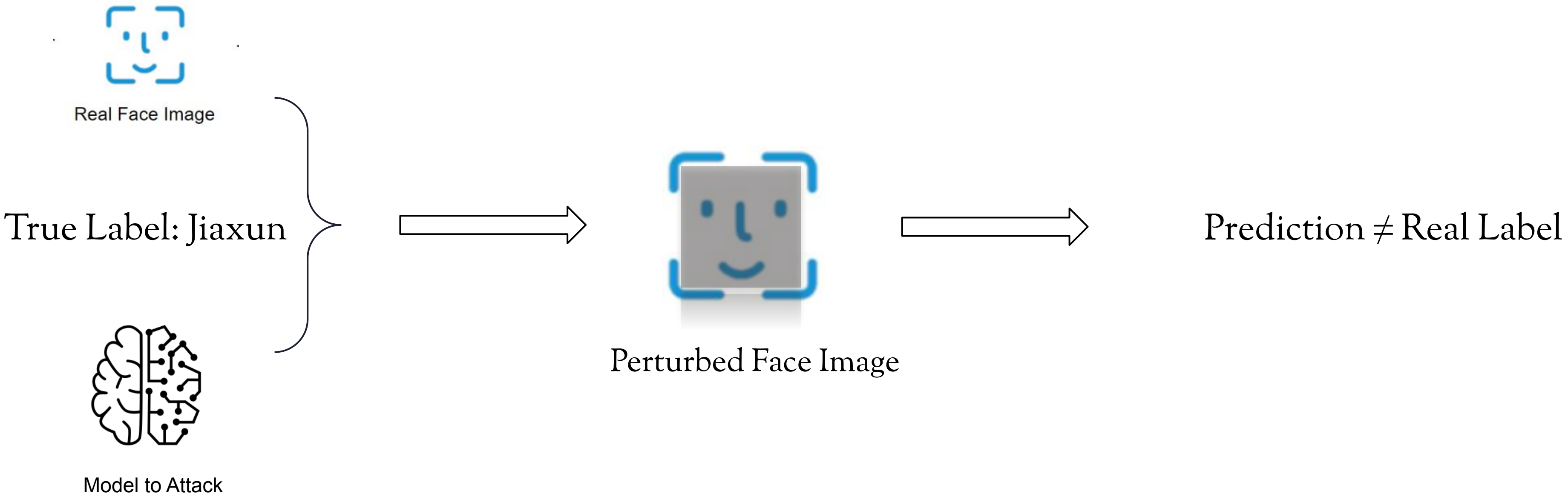
- Input: Real face Image  , Target Label, Model
- Initialize: Delta = Random Noise
- For each Iteration:

  Modify the Delta with Step Size such that:

  $$\text{Difference}(\underbrace{\text{Model(Face Image + Delta)}}_{\text{Prediction of Perturbed Data}}, \text{Target Label})$$

  is minimum
- Return Delta

Non-Target PGD:



Non-Target PGD:

- Input: Real face Image  , True Label, Model
- Initialize: Delta = Random Noise
- For each Iteration:

  Modify the Delta with Step Size such that:

  $$\text{Difference}\left(\underbrace{\text{Model(Face Image + Delta)}}_{\text{Prediction of Perturbed Data}} , \text{True Label}\right)$$

  is Maximum
- Return Delta

Important Parameters

- Input: Real face Image , True Label, Model
- Initialize: Delta = Random Noise
- For each Iteration:

    Modify the Delta with Step Size such that:

    Difference(Model(Face Image + Delta) , True Label)

    is Maximum

- Return Delta

Experimental Results of targeted attack

Step Size \ # Step	1	2	3	4
0.1	0.02	0.965	0.997	0.999
0.2	0.001	0.839	0.983	0.990
0.3	0.001	0.637	0.805	0.993
0.4	0	0.328	0.779	0.976

Accuracy of the encrypted face to be predict to the target class

Future work & Ethic concerns

- Unsupervised, general purpose encryption
  - One step forward: non Ad hoc model encryption

- Ethical issue:
  - I’m happy but Bowen is sad
  - Ethics model

THANKS!

code: github.com/coollx/FaceEncryption