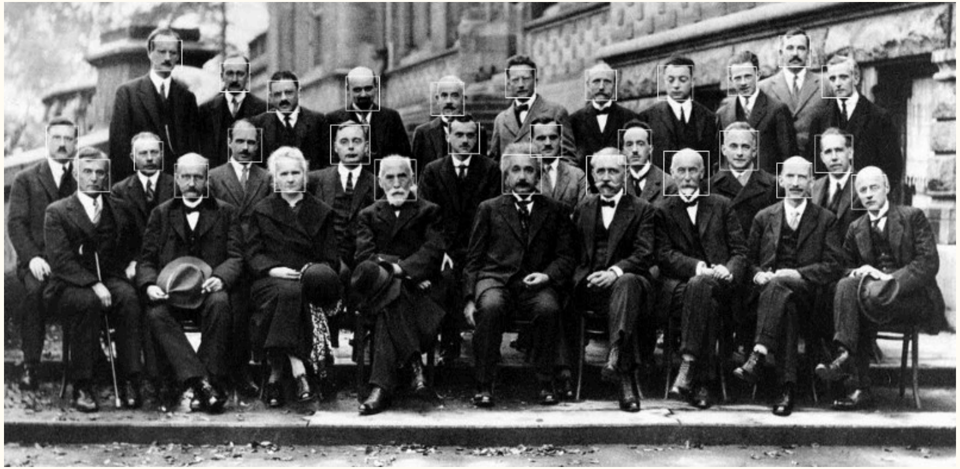
Face Detection

Team Name: Power Machine
Team Members: Yu Du, Bincheng Huang, Siyi Liu

Introduction

Background

- Face detection is the pre-procedure of face recognition.
- Face detection and face recognition are widely used currently. Such as DSLR camera can focus on a face by face detection.
- Our GOAL is to find out the human face(s) in a digital photo



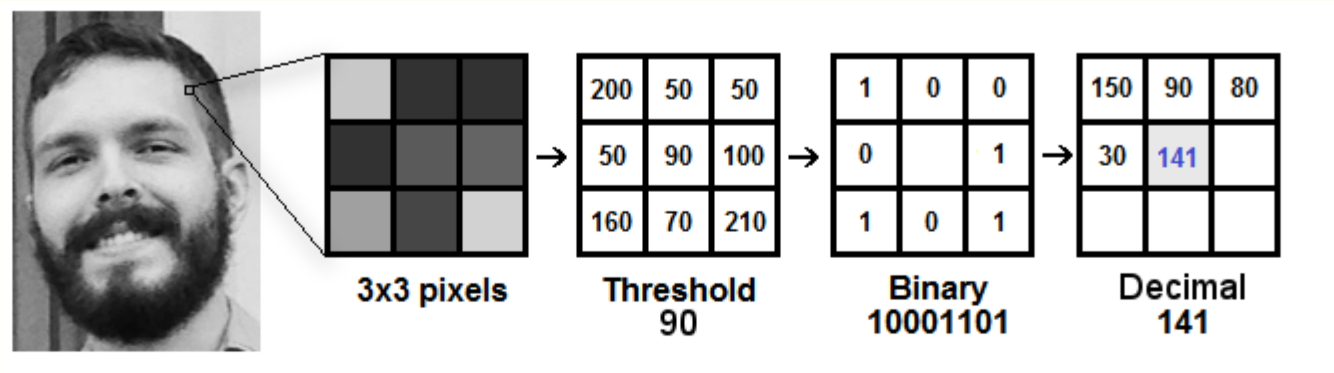
Dataset

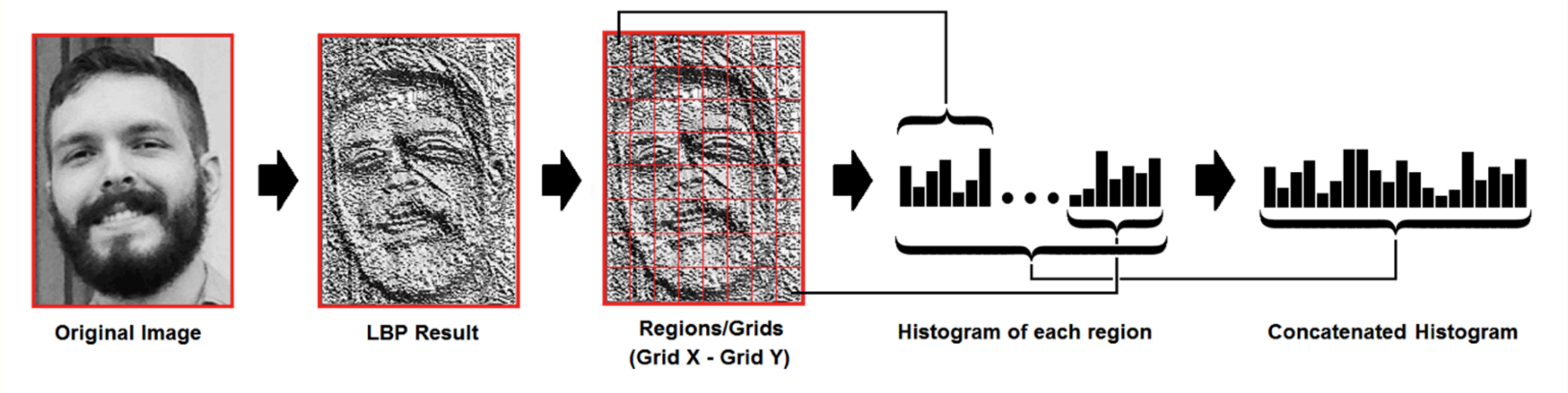
- There are two parts of dataset: training dataset and testing dataset. The training dataset contains 1100 celebrity face images (.png file), while the testing has 100 images. (Extract from https://www.kaggle.com/jessicali9530/celeba-dataset and http://www.milbo.org/muct/)
- All images are about the whole front faces, including explicit eyes, ears, nose, and mouth.
- For, training dataset, 1100 of them are used to train distance threshold and the remain 100 images are used to train probability bar (will be introduced later).

LBPH Algorithm

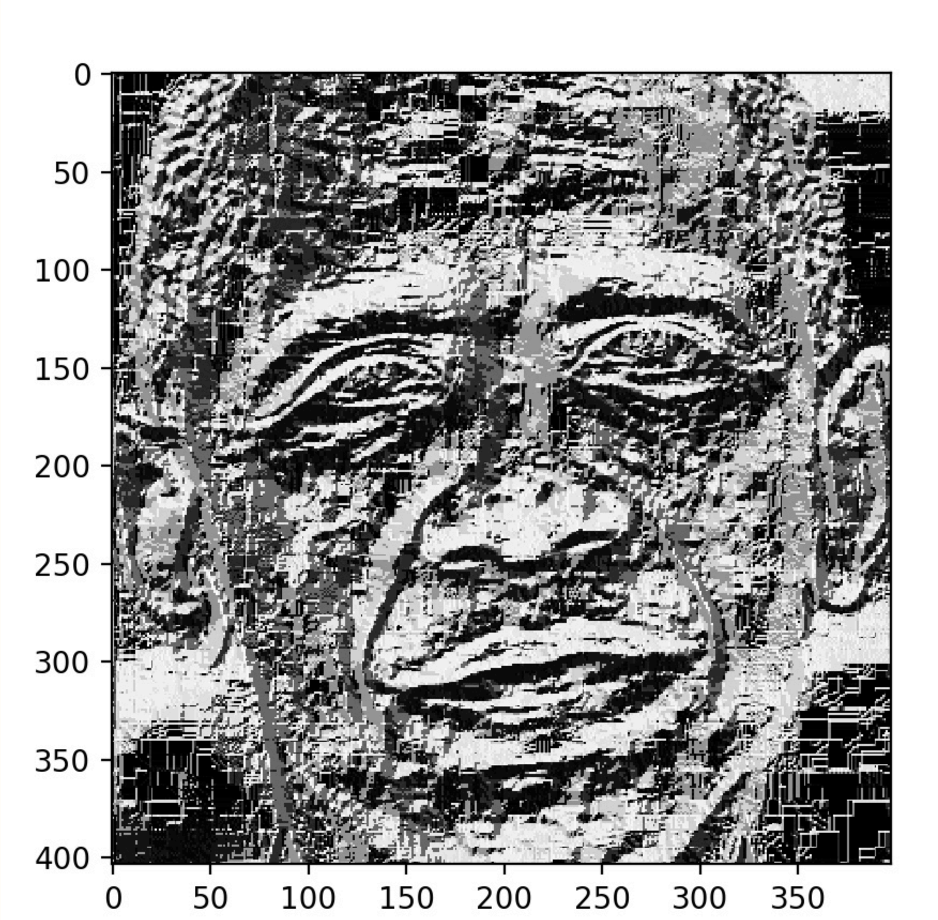
Local Binary Patterns Histogram Procedure

- **Step 1:** Convert the RBG image into grayscale image.
- **Step 2:** Set the LBP environment as Radius = 1, Neighbor = 8.
- **Step 3:** For non-boundary pixels, each pixel related to a 8 bits binary number, i.e. a decimal number between 0 and 255.
- **Step 4:** Set number of cells in x-axis (Grid_X) and y-axis (Grid_Y) for dividing the image.
- **Step 5:** Count the frequency of LBP result for each small piece of divided image.
- **Step 6:** Concatenate all Grid_X * Grid_Y histogram into a whole histogram (with Grid_X * Grid Y * 256 positions). That whole histogram is the pattern for the original image.

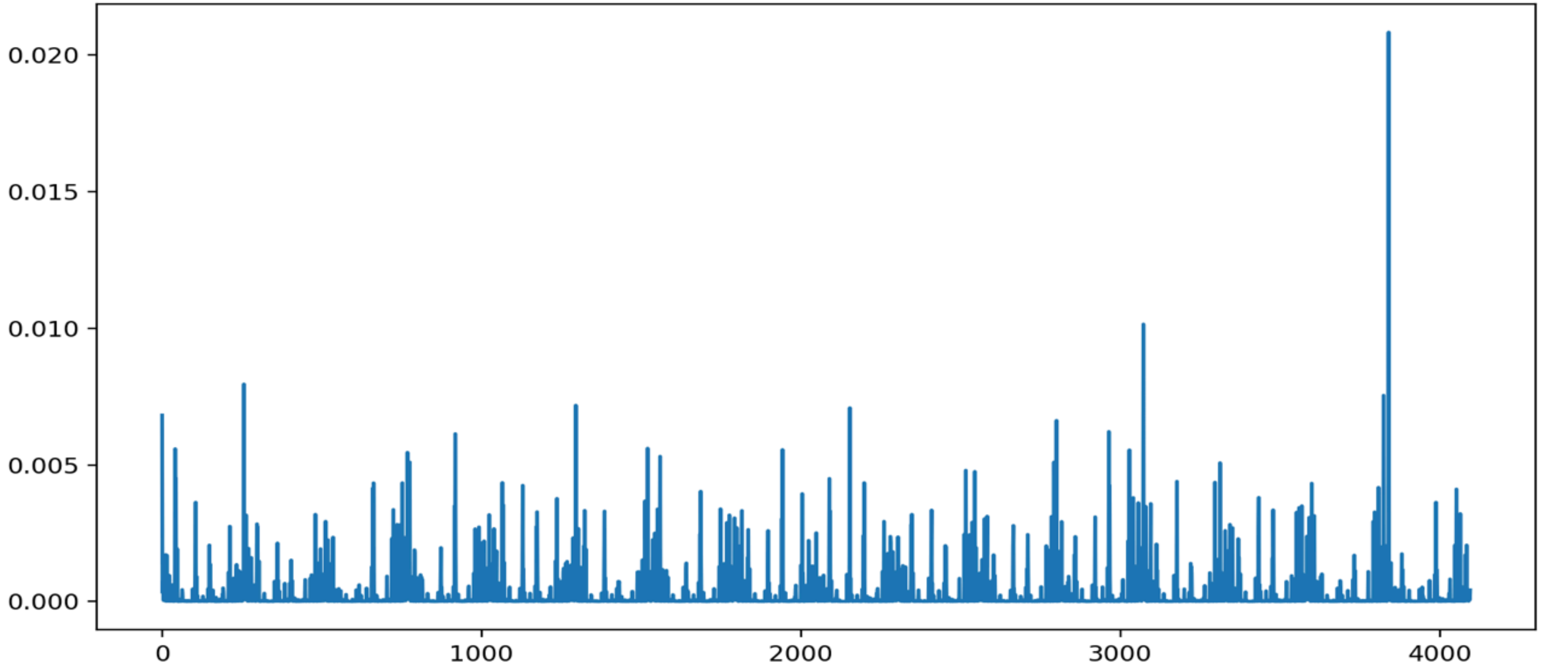

200 50 50
50 90 100
160 70 210
1 0 0
0 1
1 0 1
150 90 80
30 141
3x3 pixels
Threshold
90
Binary
10001101
Decimal
141


Original Image
LBP Result
Regions/Grids
(Grid X - Grid Y)
Histogram of each region
Concatenated Histogram

Local Binary Patterns



Histogram



Training Model

Distance Threshold

Training Dataset 1

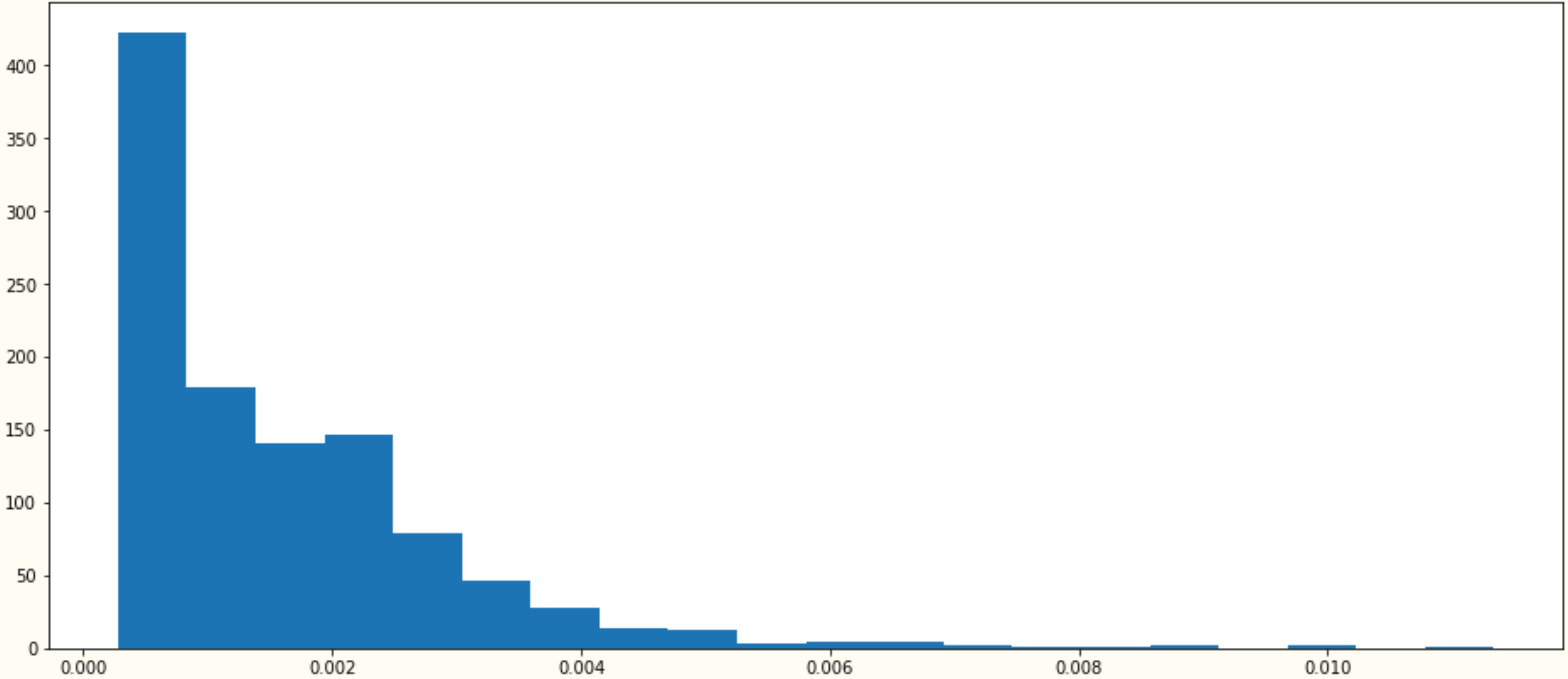
1000 hist patterns(faces)

Euclidean Distances
between patterns

Training Dataset 2

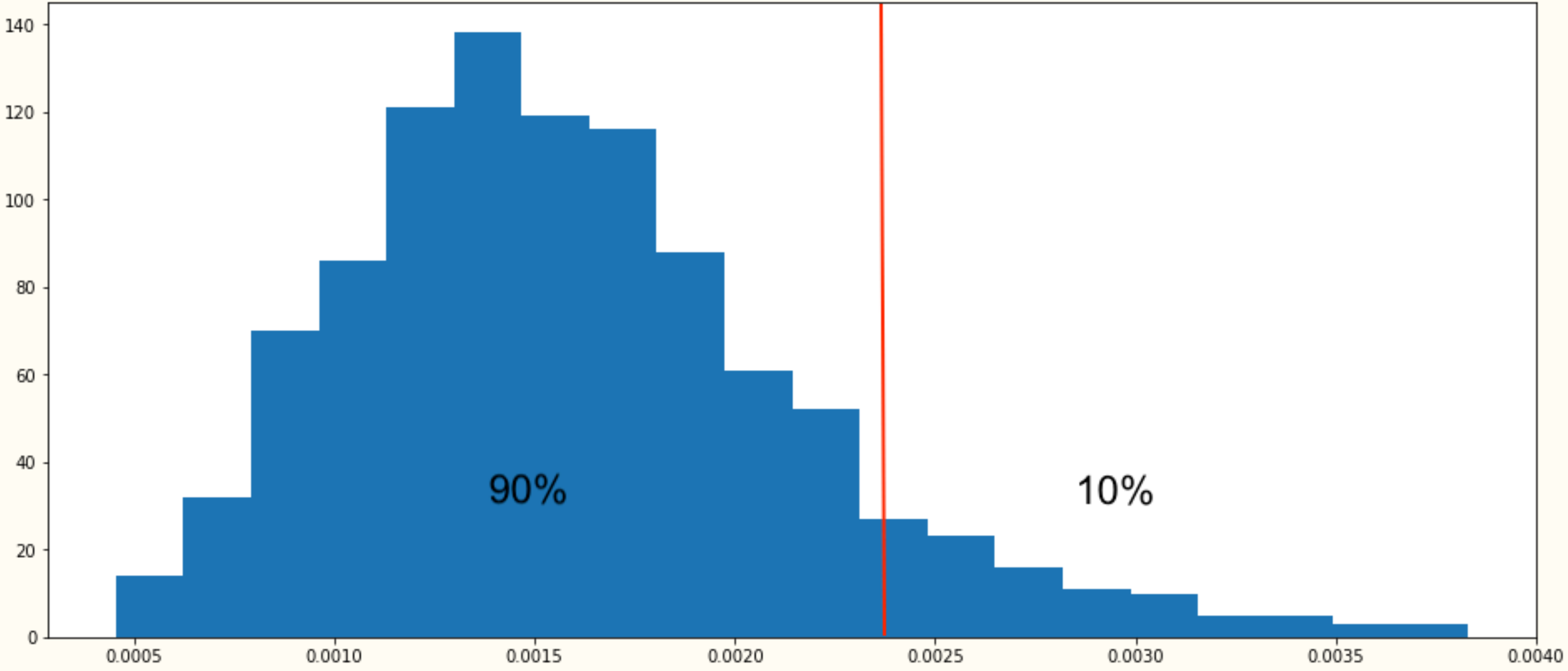
100 hist patterns(faces)

Training Dataset 1 Distance Distribution



Distance Threshold

mean+1.28std: 0.0023



Probability Threshold

Training Dataset 2

100 hist patterns

Pick one pattern

Calculate distances

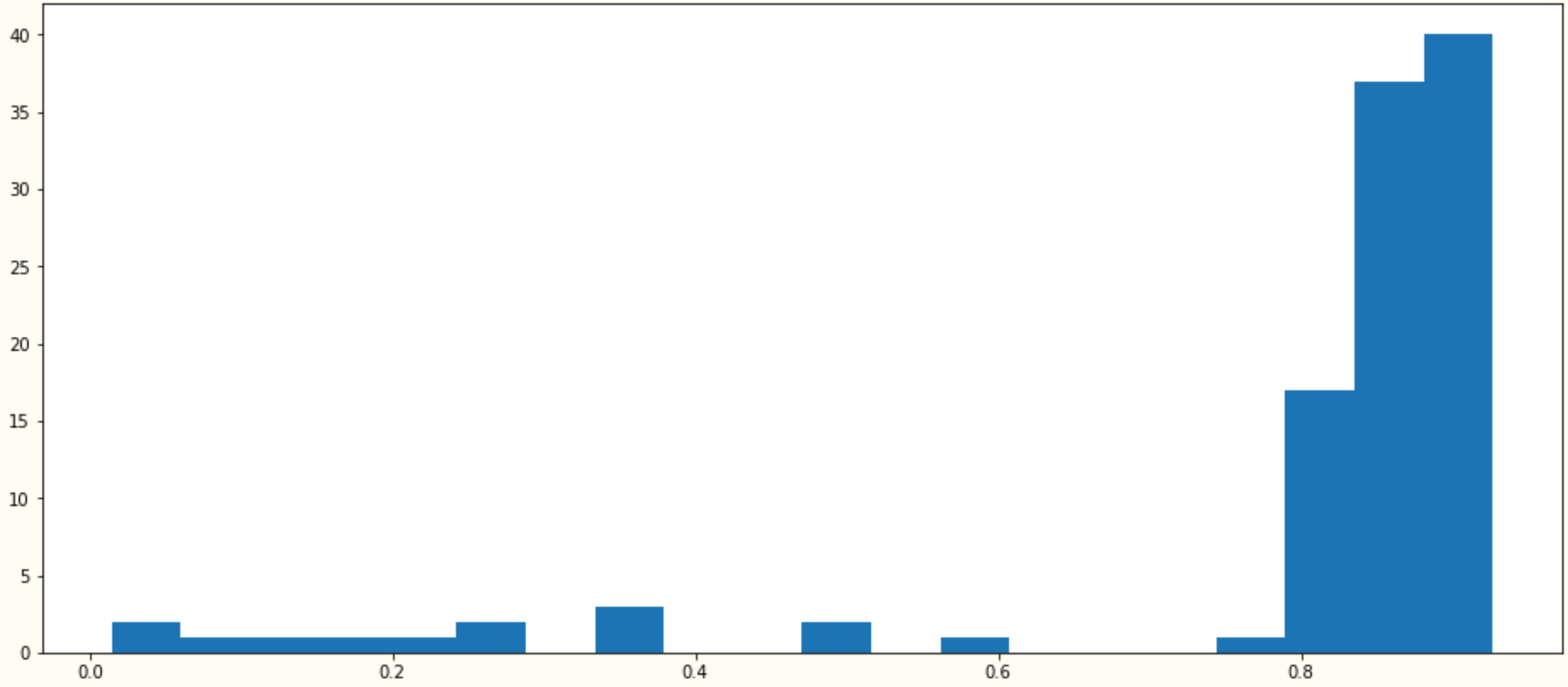
Training Dataset 1

1000 hist patterns

Probability of distances
below distance threshold
in 1000 distances

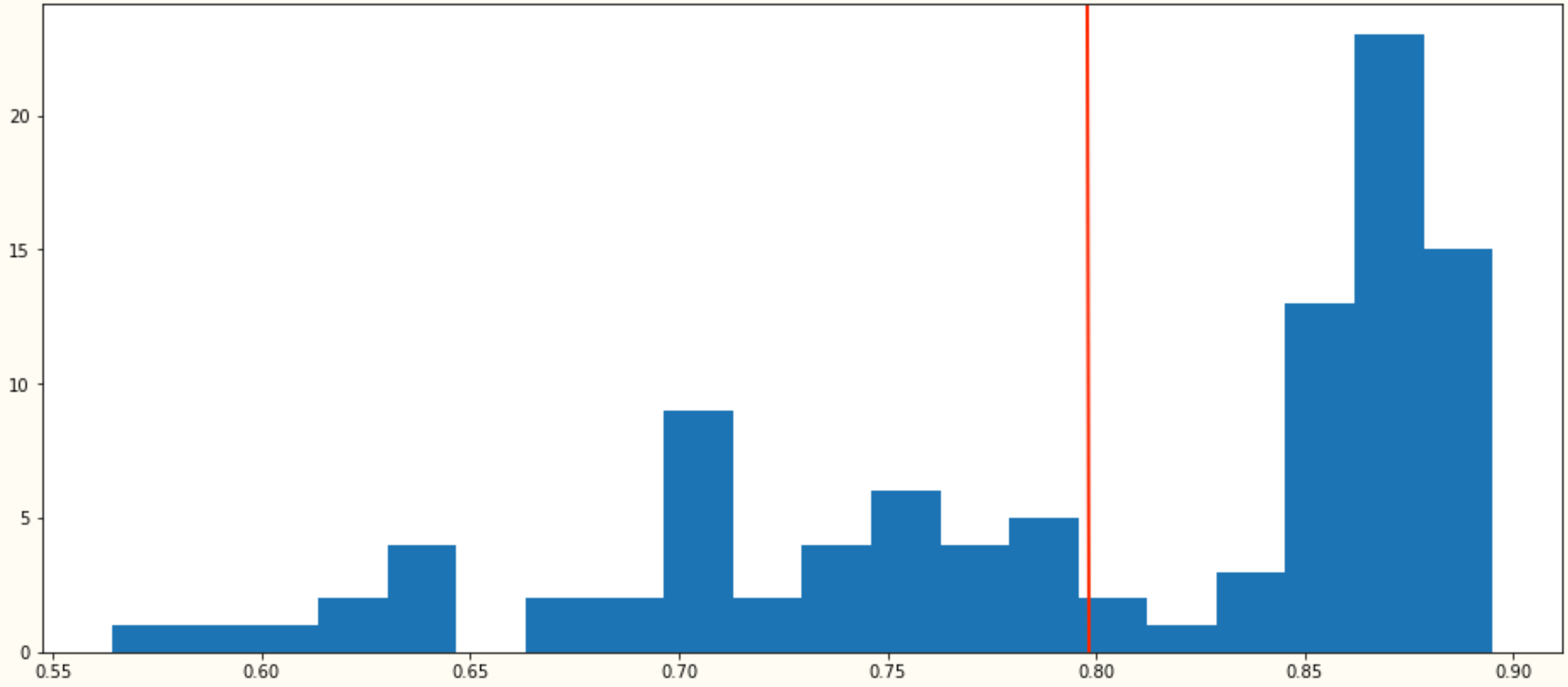
Get 100 probability data

Training Dataset 2 Probability Distribution



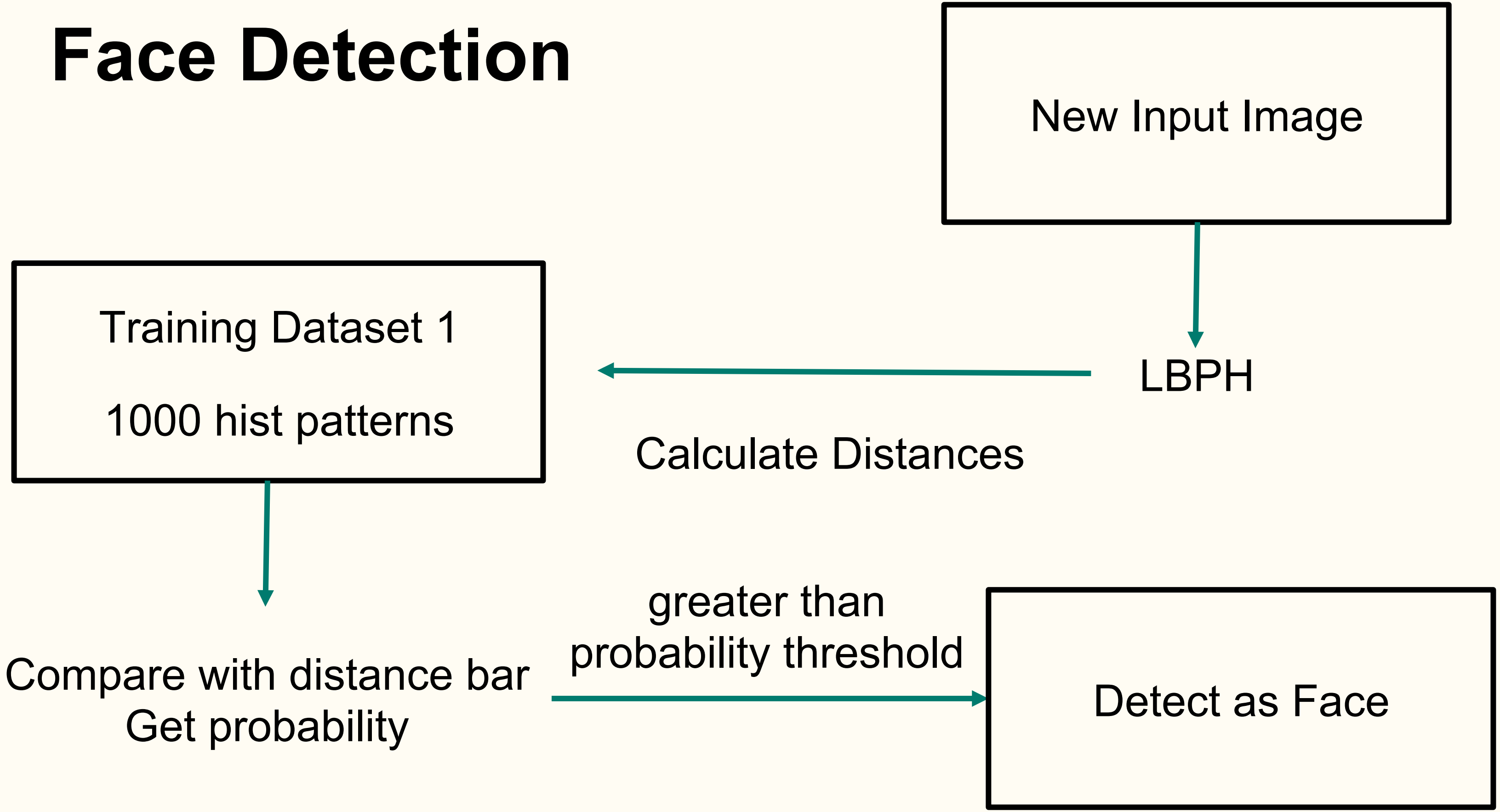
Probability Threshold

mean: 79.8%

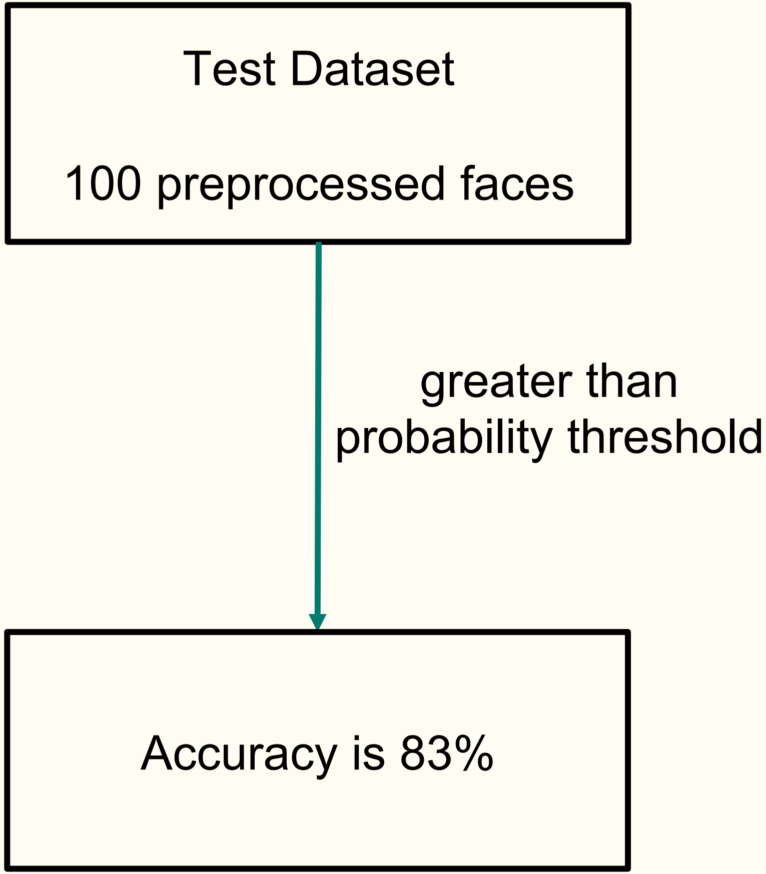


Face Detect Process

Face Detection



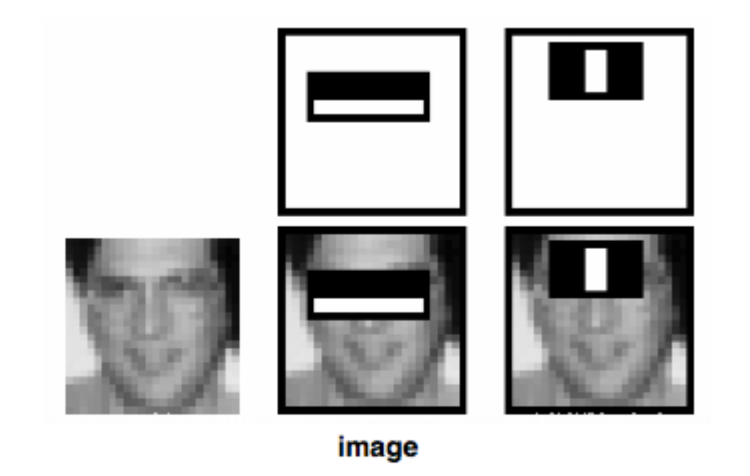
Accuracy


Test Dataset
100 preprocessed faces
greater than
probability threshold
Accuracy is 83%

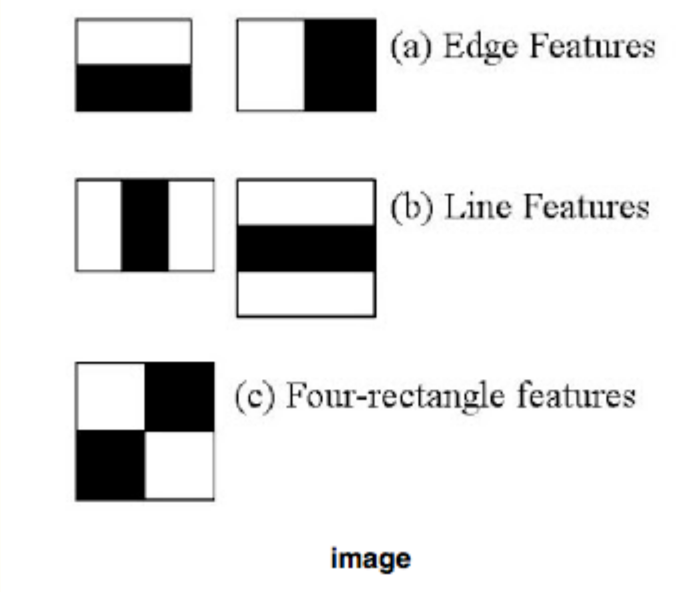
Haar Cascades

How does it work?

Each feature is a single value obtained by subtracting sum of pixels under the white rectangle from sum of pixels under the black rectangle.



Haar Features



For a 24*24 pixels image, we can have 160,000 Haar features from all possible sizes and locations of each feature.

AdaBoost

The output of the other learning algorithms ('weak learners') is combined into a weighted sum that represents the final output of the boosted classifier. AdaBoost is adaptive in the sense that subsequent weak learners are tweaked in favor of those instances misclassified by previous classifiers.

160,000 --> 6,000

For this, we apply each and every feature on all the training images. For each feature, it finds the best threshold which will classify the faces to positive and negative. Obviously, there will be errors or misclassifications. We select the features with minimum error rate, which means they are the features that most accurately classify the face and non-face images.

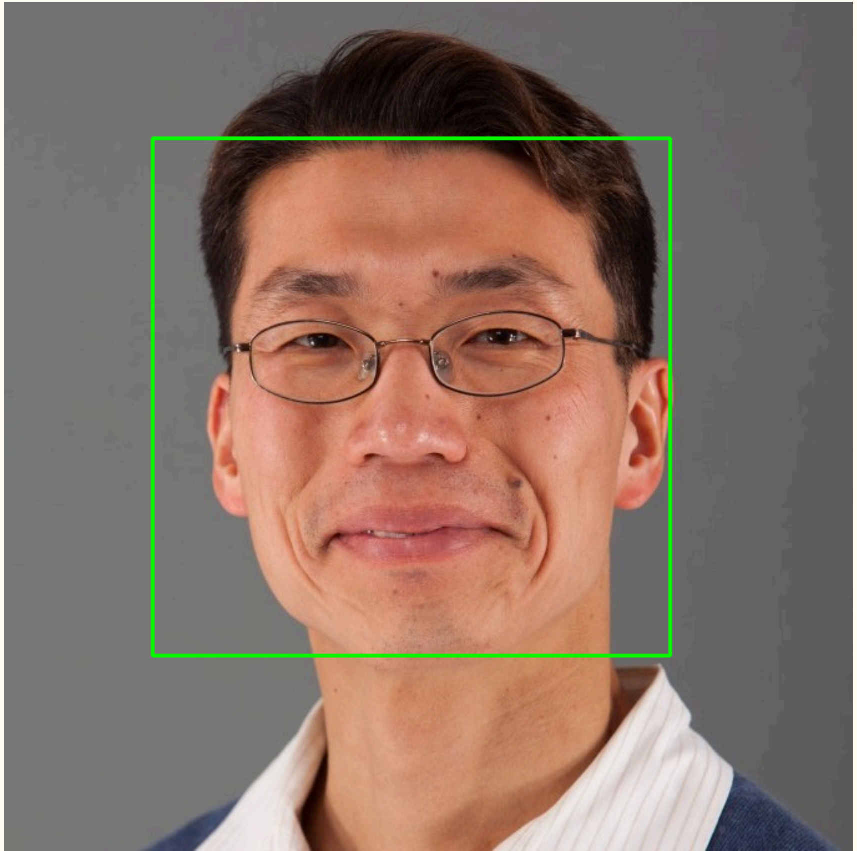
Cascades Classifiers

Take each 24*24 pixels window. Instead of applying all 6000 features on a window, the features are grouped into different stages of classifiers and applied one-by-one.

If a window fails the first stage, discard it. We don't consider the remaining features on it. If it passes, apply the second stage of features and continue the process. The window which passes all stages is a face region.

1 -> 10 -> 25 -> 25 -> 50

Haar Cascades Test Results



Conclusion

Conclusion

- Face Detection Algorithm developed by our own has 83% accuracy.
- Sliding Window Algorithm cannot locate people’s face perfectly.
- Haar Cascades Algorithm has 93% ~ 95% accuracy based on OpenCV history record.
- Next, we will try to develop Face Recognition Algorithm using SVM or Regression Classifier.

Q & A

Q & A

- Q1 (from Prof. Chin): Just detect an image whether is a human face or not?
  - Answer: Not just it. We are also trying to detect the human faces in a normal photo, such as a photo taken in our classroom.
- Q2 (from Prof. Chin): What the result will be just a partial face?
  - Answer: Our algorithm probably not be able to handle that situation. We need more time to tackle this problem.
- Q3 (from classmate): What if the image contain a person with hat or glasses?
  - Answer: Because our training dataset doesn’t include a person with hat or glasses. So, our current model would give a wrong classification. But we can enlarge our training data with those images and re-train our model. That would be better.

Q & A (Cont’d)

- Q4 (from classmate): Will it work if the human face is in the dark environment?
  - Answer: It will. The LBPH can extract the pattern even though the picture is in the dark environment since LBP operator is considering the relative value between center and neighbor. Thus, it doesn’t matter if the environment is dark or bright.
- Q5 (from classmate): What if testing the painting face?
  - Answer: We are not sure, to be honest. It still can exclude some “faces” like the logo of Starbucks, but not sure for the masterpiece of a painting which is really like a real human.