Research on Face Detection Based on Deep Learning

Zhuoran Li[1, a]

[1]Shanghai Maritime University, Shanghai, China.

[a]15628408008@163.com

Abstract

With the rapid development of science and technology, computer technology is playing an increasingly important role in today's life. The ability of computers to process visual information has largely made up for the shortcomings of human vision, thus making computer vision research also has become one of the hot research directions nowadays. This paper proposes two research methods. One is based on the traditional AdaBoost face detection algorithm, and the algorithm is implemented based on OpenCV. The results are tested and analyzed by AFW dataset and FDDB dataset. The second is based on the deep learning algorithm of convolutional neural network, based on the TensorFlow framework, implements a face detection algorithm based on convolutional neural network, and finally uses the AFW data set and FDDB data set to verify the algorithm. The experimental results show that the face detection algorithm based on deep learning can not only detect the front face, but also the side faces that cannot be detected by the traditional AdaBoost algorithm, and the former has better detection effect. The experimental results show that the face detection algorithm based on deep learning can not only detect the front face, but also the side faces that cannot be detected by the traditional AdaBoost algorithm, and the former has better detection effect.

Keywords

Face Detection; AdaBoost; Convolution Neural Network; Deep Learning.

1. RESEARCH BACKGROUND

At present, the analysis of human faces reflects that visual information plays an important role in people's communication. Processing and analysis of features reflected by human faces has become one of the most important topics in the field of computer vision in recent years. On a multimedia platform, thousands of picture information needs to be processed every day [1]. Retrieving and processing these picture's information will directly affect the user's experience of the platform. On a multimedia platform, thousands of picture information needs to be processed every day. Retrieving and processing these picture's information will directly affect the user's experience of the platform.

Face pictures usually combine a variety of personal information including identity, age, expression, etc. Compared to traditional digital passwords, because of their individual independence, it is more difficult to crack, so they are often used in access control systems in various places. In daily life, in order to protect their privacy and the security of their own interests, the commonly used protection measures can be divided into two categories, digital passwords and unique biological characteristics of the human body. Figure 1. shows the characteristics of the two categories.

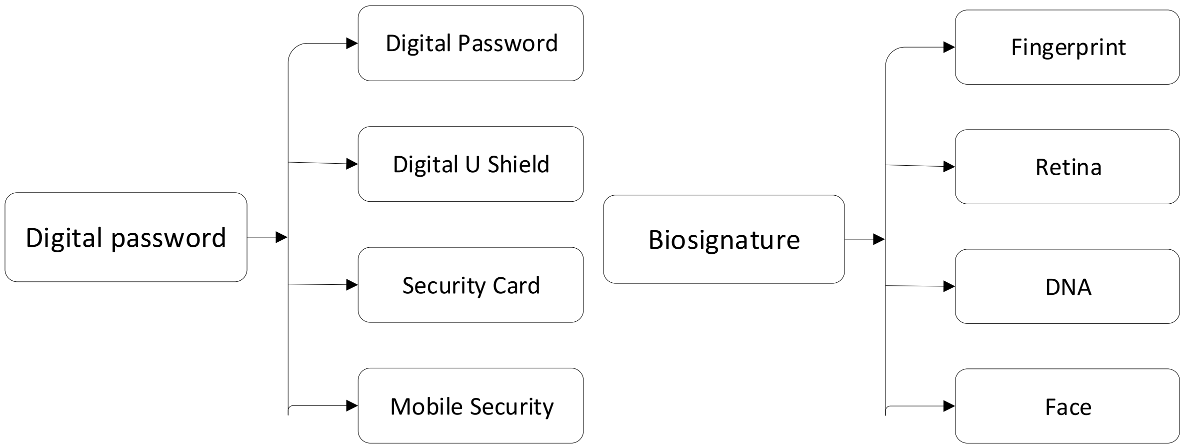

Figure 1. Common ways to protect privacy

However, with the rapid development of face recognition technology in recent years, the background of the face image is complicated, so the accuracy requirements for face detection are getting higher and higher [2]. In this case, the prerequisite for face recognition is that the face can be accurately detected without being affected by the application background during the detection phase. At present, the main direction of face detection research is to solve the impact of face detection in complex environments. The goal of this article is to verify the effectiveness of deep learning technology in face detection in this direction.

2. FACE DETECTION ALGORITHM BASED ON ADABOOST

2.1. AdaBoost Introduction

AdaBoost is an iterative algorithm. The core idea is to train different classifiers for the same training set, and then combine these weak classifiers to form a stronger final classifier. AdaBoost itself is achieved by changing the data distribution. It determines the weight of each sample based on whether the classification of each sample in each training set is correct and the accuracy of the previous overall distribution. It will modify the new full-time. The data set is sent to the lower classifier for training, and finally the classifiers obtained each time are fused and used as the final decision classifier.

AdaBoost is a Boosting method, in addition to the Bagging method. Boosting is a technology very similar to bagging. The types of multiple classifiers used by both are consistent, but in Boosting, different classifiers are obtained through serial training. Each new classifier both are trained on the basis of the performance of the trained classifiers, and new classifiers are obtained by focusing on what data is misclassified by the existing classifiers. The result of Boosting classification is based on the weighted summation of all classifiers. The weights of the classifiers are not equal. Each weight represents the success of its corresponding classifier in the previous iteration. There are many Boosting algorithms, and AdaBoost is the most popular of them. Figure 2 shows the algorithm flowchart:

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202003_6(3).0012

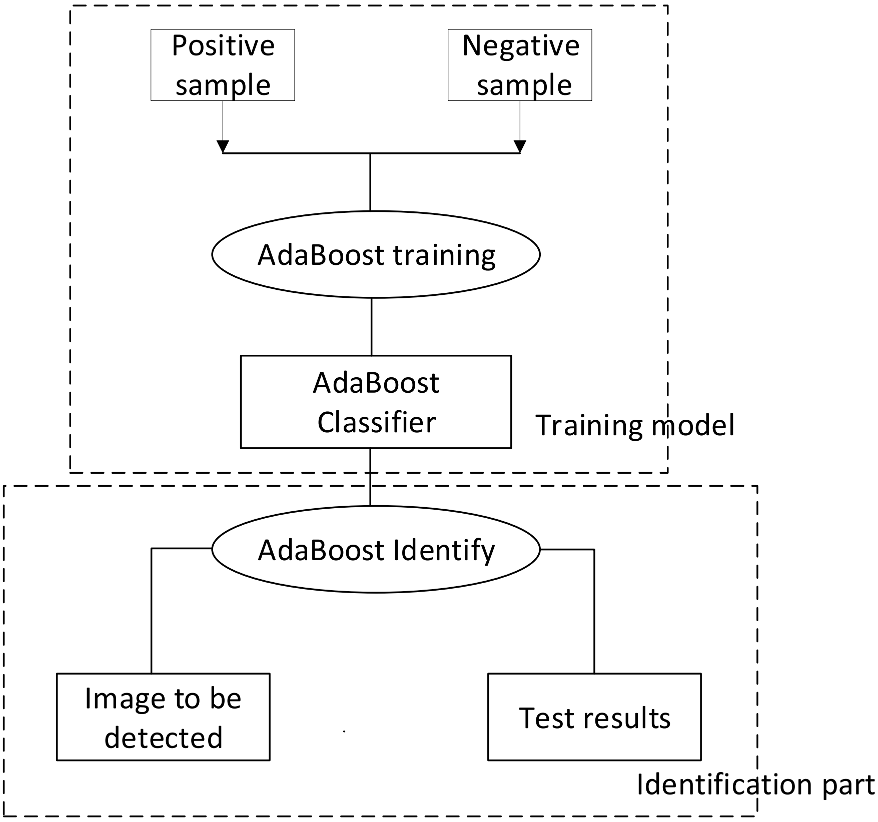

Figure 2. Phase analysis of arrangement

2.2. The Running Process of Adabost Algorithm

For each sample in the training data, a weight is assigned, and these weights constitute a vector D. Initially, these weights are initialized to equal values. First train a weak classifier on the training data and calculate the error rate of the classifier, and then train the weak classifier again on the same data set. First train a weak classifier on the training data and calculate the error rate of the classifier, and then train the weak classifier again on the same data set. In the second training of the classifier, the weight of each sample will be readjusted. The weight of the first paired sample will be reduced, and the weight of the mismatched sample will be increased. In order to get the final classification results from all weak classifiers, AdaBoost configures a weight value α for each classifier. This α value is calculated based on the error rate of each weak classifier. The error rate ε is defined as:

ε= Number of incorrectly classified samples/ Number of all samples

The calculation formula of α is:

$$\alpha = \frac{1}{2}\ln\left(\frac{1-\varepsilon}{\varepsilon}\right) \tag{1}$$

The formula shows that when ε> 0.5, α> 0; when 0 <ε <0.5, α <0.

After α is calculated, the weight vector D can be updated so that the weights of the correctly classified samples are reduced, and the weights of the misclassified samples are increased. D calculation method:

$$D_i^{(t+1)} = \frac{D_i^{(t)} e^{-\alpha}}{Sum(D)} \tag{2}$$

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202003_6(3).0012

Sample weights for misclassification:

$$D_i^{(t+1)} = \frac{D_i^{(t)} e^{\alpha}}{Sum(D)} \tag{3}$$

Putting it all together:

$$D_i^{(t+1)} = \frac{D_i^{(t)} e^{-predict*label*\alpha}}{Sum(D)} \tag{4}$$

After D is calculated, AdaBoost starts the next iteration. The AdaBoost algorithm trains a large number of different training sets, and then cascades the new classifiers trained in this way into a strong classifier. If the weak classifiers are better than the speculative ability, then if there are countless The classifier can reduce the error rate of the strong classifier to near zero. The flowchart is as follows:

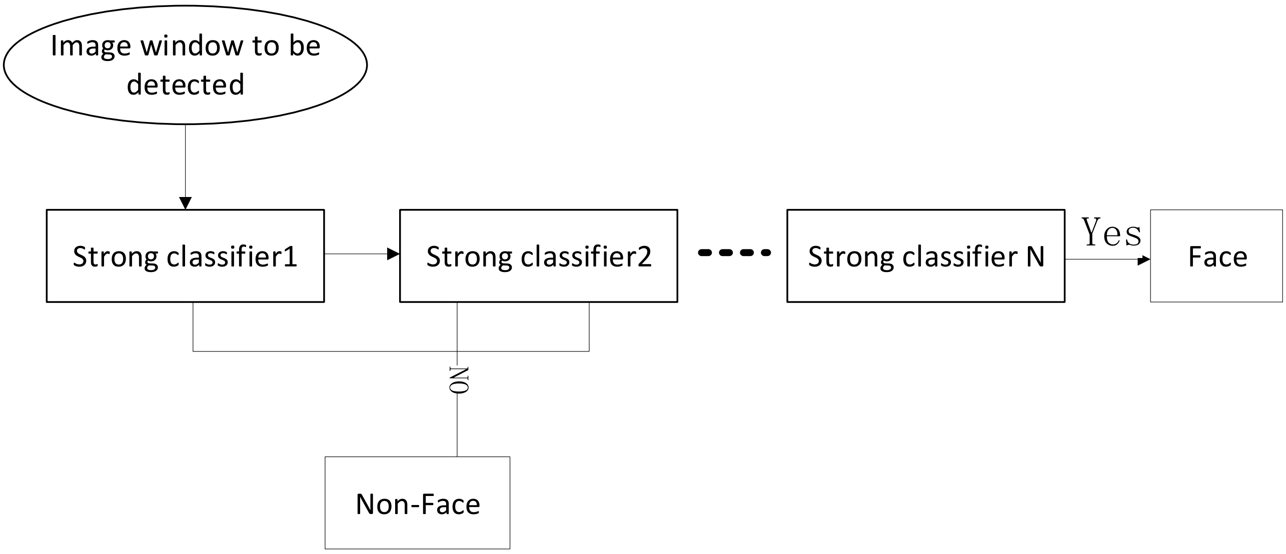

Figure 3. Detection flowchart

2.3. Haar Feature and Integral Graph

Haar feature is used to describe the gray rule of the user's face. It is a weak feature. The advantages of using Haar feature extraction are faster and simpler to calculate. It is often used in the direction of face detection [3].

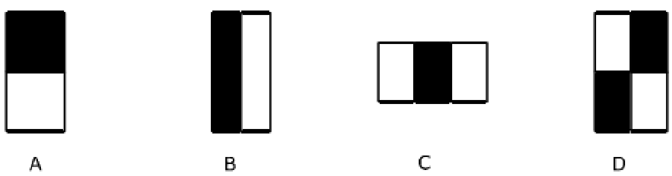

Figure 4. Boundary feature

The figure above can be divided into three categories according to the number of image element distribution features: 1. For example, Figure C is a three-hold feature; 2. Four-

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202003_6(3).0012

rectangular features are shown in Figure D; The calculation of Haar eigenvalues only involves the value of related image elements, and is a simple addition and subtraction operation, that is, the sum of the values of white image elements is subtracted from the black. And the Haar feature works well in describing the grayscale of the face.

Viola et al. Established a theory on integral graphs to reduce computational difficulty and speed up calculations. This theory is very similar to mathematical integrals, as shown in the figure below. The sum of the image element points in this box is equal to the (x, y) point integral.

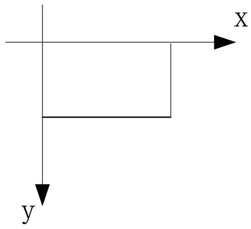

Figure 5. Integral graph

As can be obtained from the above figure, the integral map is saved by traversing the face image. If you need to get the sum of the image elements in a certain area, you only need to add or subtract the end points of each area. Efficiency has been greatly improved.

Integral chart rule: Image element point P (x, y) in the image Integrated image:

$$P_{II} = \sum_{-i \leq x, j \leq y} I(i, j) \tag{5}$$

Where I (i, j) is the gray value of (i, j) point, and P_II is the integrated image from the origin to (i, j) point. Li Yong's integration chart rule calculates the sum of the gray values of the image elements in D area, see the figure 6.:

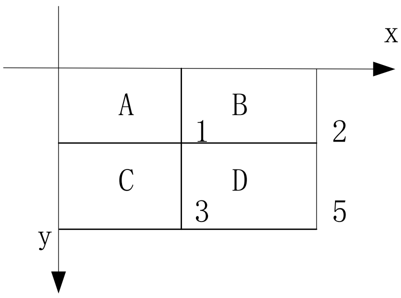

Figure 6. Integral chart calculation diagram

According to the above integration map rules, a point integration map can be obtained. Based on these integration maps, the sum of the gray values of the image elements in the D area can be calculated:

$$f_1 = \sum_A I(x, y) \tag{6}$$

$$f_2 = \sum_{A+B} I(x, y) \tag{7}$$

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202003_6(3).0012

$$f_3 = \sum_{A+C} I(x,y) \tag{8}$$

$$f_4 = \sum_{A+B+C+D} I(x,y) \tag{9}$$

$$S_D = f_1 + f_4 - f_3 - f_2 \tag{10}$$

The values of the four points integrals in the figure above are denoted as f1, f2, f3, f4, and Sd, which is the sum of the gray values of the image elements in the D-frame range. From the above, rectangular feature values can be obtained by adding and subtracting the integral graph of each vertex of the box range. Performing the calculation in this way transforms a very complex eigenvalue operation into a plurality of simple eigenvalue addition and subtraction calculations, reducing a part of the calculation amount.

2.4. Design of Cascade Classifier

Experiments show that according to AdaBoost, a large number of weak classifiers are combined into a strong classifier, which can be used for face detection. But this method will generate a huge number of windows, and each window needs to calculate hundreds of eigenvalues, which is a waste of time. We use a construction method, which is to first use a simple classifier composed of features with higher contributions, and use this to simply remove pictures that are not faces, and then let this picture enter the complex composed of features with lower contributions the classifier performs processing in order to determine that the detected picture is a face picture.

3. ADABOOST ALGORITHM ACHIEVES FACE DETECTION RESULTS

When using OpenCV's built-in detector for face detection, select some images from the AFW dataset for detection. Some of the detection results are shown in the figure. The blue circles and circles are the detected faces.

Figure 7. Face detection pictures

It can be seen from the detection results that in the case of a simple background and a positive face, a human face can be detected correctly. In the case where the background of the image is complex and there are side faces, it is easy to mistakenly detect a human-like region as a human face or successfully detect a human face. Therefore, in the case of complex backgrounds, the AdaBoost face detection algorithm is very susceptible to the impact of complex environments, resulting in unstable detection results and a high false detection rate.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202003_6(3).0012

4. FACE DETECTION ALGORITHM BASED ON CONVOLUTIONAL NEURAL NETWORK

In the research and development of deep learning, convolutional neural network is a representative network in deep neural networks. In recent years, convolutional neural networks have developed rapidly, and have attracted wide attention from all walks of life with their strong recognition capabilities. As early as the 1960s, Hubel and Wiesel discovered in the process of studying the visual cortex of the brain that neurons are connected by a unique network structure, which can reduce the complexity of feedback neural networks. Therefore, CNN is proposed. Now, CNN has become one of the hotspots in the field of machine learning, and has achieved good results in multiple directions, such as face detection and recognition, speech recognition, video image processing, etc.

4.1. The Composition of A Convolutional Neural Network

In a convolutional neural network, the convolutional layer is its core component, and its role is to extract various features of the image. The convolutional layer is located after the input layer or the pooling layer. After calculation of the sliding window of the input layer, the size of each parameter in the convolution kernel can be regarded as the weight parameter in the traditional neural network, which is related to the corresponding local pixel After concatenation, the parameters of the convolution kernel are multiplied by the corresponding local pixel values to obtain the results on the convolution layer [4].

After the convolutional layer is calculated, if it uses the calculated features to train the classifier directly, it will face a huge amount of calculation, which will be seriously affected in time and efficiency. For example: suppose a 96 × 96 pixel image has got 400 features defined on 8 × 8 input, and each feature will get a new $(96-8+1) \times (96-8+1) = 7921$ dimensional convolution features. Because there are 400 features of the same size, you end up with a convolutional feature vector that requires 3168400 dimensions. At this time, it is very difficult to train a classifier with more than 3 million features, and overfitting may occur. Pooling the convolutional layer can relatively reduce the degree of overfitting of the model and the parameters of network training. The role of pooling is to compress data, speed up operations, and improve the robustness of extracted features. There are many methods for convolutional neural network pooling, such as average pooling, maximum two pooling, overlapping pooling, mean square pooling, normalized pooling, random pooling, and deformation constraint pooling.

In a convolutional neural network, the fully connected layer is usually at the end of the convolutional neural network. The function of the fully connected layer is to integrate the features in the image feature map through multiple convolutional layers and pooling layers, combine all local features into global features, obtain the high-level meaning of image features, and use it for images classification [5].

The use of activation functions in neural networks can add some non-linear factors to the network so that the neural network can arbitrarily approach any non-linear function. The neural network is applied to many non-linear models to better solve some more complex problems. If the activation function is not used, the linear function of the input of each layer will be used as the output of the function of the next layer. In the end, no matter how many layers of the neural network exist, the output result is a linear combination of the input layer.

4.2. Face Detection Process

The process of face detection mainly includes three core phases: face area selection, feature extraction and classifier classification. The face detection process is shown in the figure 8.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202003_6(3).0012

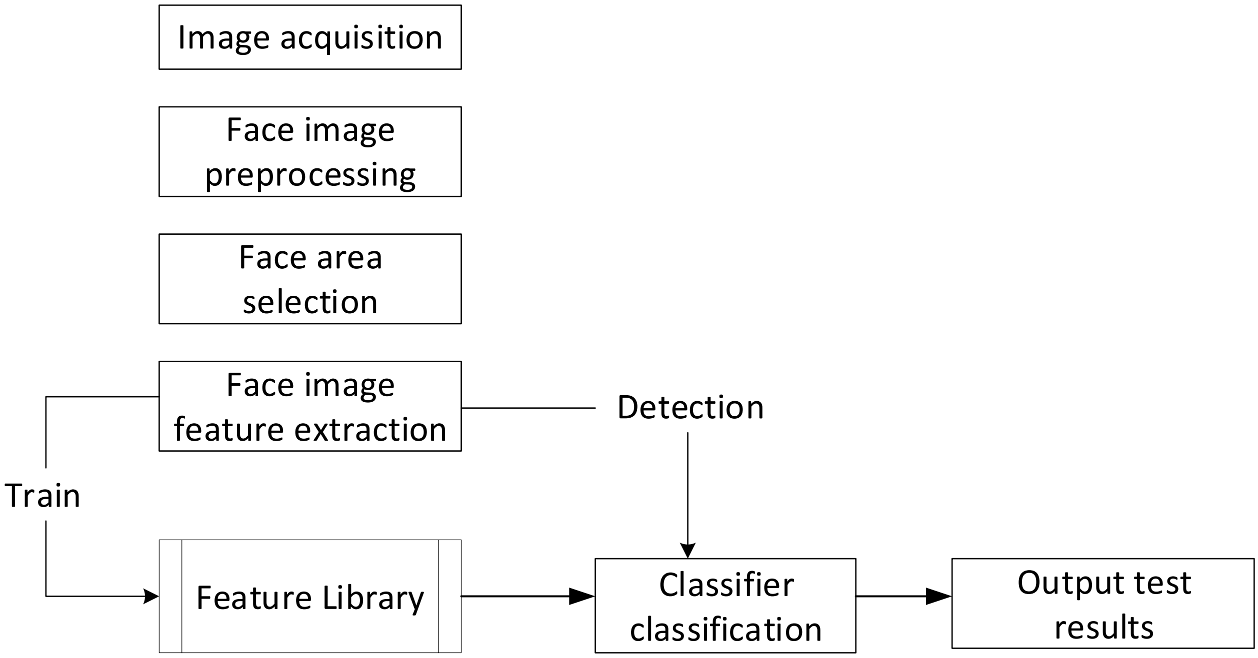

Figure 8. Face detection process

Face area selection: This stage is the first step in face detection. Its main purpose is to find some candidate frames in the face image and use these candidate frames to locate the position of the face. In an image containing a face, the position of the face may appear anywhere in the image, and the size and angle of the face are not determined. Therefore, the selection of candidate frames in the image is in the face detection process. Plays a vital role.

Feature extraction: This stage is the key and core of face detection. Because of the size of the face contour, hairstyle, lighting, shooting angle, background diversity and other factors, it is not so easy to extract accurate feature information. Therefore, the quality of the extracted features directly affects the accuracy of subsequent classification [6].

Classifier classification: This stage is to train the classifier. The feature information extracted in the feature extraction stage is sent to the classifier for training, so that the classifier classifies these feature's information into human faces and non-human faces. Currently popular classifiers include SVM, Adaboost algorithm, and so on.

4.3. Test Results

This experiment mainly uses the AFW and FDDB datasets. The AFW dataset is a face image library created using Flickr (Yahoo's image sharing website) images, including 205 images, of which 473 labeled faces. FDDB is one of the most authoritative face detection and evaluation platforms in the world. It contains 2845 pictures with a total of 5171 human faces as a test set. The test set range includes pictures with different poses, different resolutions, rotations and occlusions, as well as grayscale and color maps. The background changes of the face pictures in these two data sets are diverse, including the front and side face images, which can be used for comparative analysis experiments. The two face detection methods are used to perform face detection experiments on the AFW dataset and the FDDB dataset, respectively.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202003_6(3).0012

Figure 9. Face detection pictures(b)

5. EXPERIMENTAL RESULTS

Compared with the AdaBoost face detection algorithm, the biggest difference between the two is the training of the face classifier. The AdaBoost algorithm mainly obtains cascade classifiers about face features through training. After misclassifying and weighting samples that are misclassified multiple times, the weight will be too large, which will affect the choice of classifiers and eventually cause detection accuracy Lower. Face detection based on deep learning is mainly trained using convolutional neural network models, with high classification accuracy, strong distribution storage and learning ability, and strong robustness and fault tolerance, and can be used in complex backgrounds. Faces can also be detected better.

REFERENCES

[1] Rong Guo: The Research of Several Key Techonolgies of Face Detection (Ph.D. Harbin Engineering University, China 2010), p. 1

[2] Senlin Cheng:Face Recognition Research and Design Based on Deep Learning in Monitoring Scene(MS. Beijing University of Posts and Telecommunications, China 2019), p.2

[3] Schapire R E,Singer Y. Improved boosting algorithms using confidence-rated predictions. Machine Learning. 1999,37(3):297-336.

[4] Li S Z, Zou X L, Hu Y X,et al.Real-Time Multi-View Face Detection, Tracking, Pose Estimation, Alignment, and Recognition. CVPR 2001 Demo Summary, Hawaii,2001.

[5] Jiang W X. Process Consistency for AdaBoost. Annals of Statistics. 2004,32(1):13-29.

[6] J. Oliver, J. K. Klaus, W. F. Robert. Robust Face Detection Using the Hausdorff Distance. 2001. pp.Pages

[7] Hjelmas E, Low B K. Face Detection: A Survey. Computer Vision and Image Understanding. 2001, 83 (31): 236-274.