Journal of Computing and Electronic Information Management

ISSN: 2413-1660

A Robust to Tilt Angle and Real-time Face Detection Algorithm

Qiming Li *, Lu Xu and Liqing Hu

College of Information Engineering, Shanghai Maritime University, Shanghai, China

*Correspondence: qmli@shmtu.edu.cn

Abstract: In order to reduce the negative effect of face tilt angle in image and achieve accurate and real-time face detection, a novel robust to angle and real-time face detection algorithm is proposed. Firstly, image scaling and adaptive threshold based on YCbCr+Ostu are used to reduce face search area for shortening computation time. Secondly, for multi angle face images, especially the images with large pitch angle, the MB_LBP feature and the improved Haar-T feature are combined. And the AdaBoost and SVM algorithm are used to train and cascade classifier for face detection. The experimental results show that when the horizontal tilt angle of face from left to right can reach (-90,+90) and the pitch angle from bottom to top can reach (-40,+45), the proposed algorithm can improve accuracy and reduce time consumption. And the robustness of face detection is excellent.

Keywords: Real Time Face Detection, Angle Robustness, Cascade Classifier.

1. Introduction

Face detection, as a core technology in face information processing [1], is the first problem to be solved in hot areas such as face tracking, face reconstruction, and face recognition. Therefore, it has become a popular topic in the field of pattern recognition and computer vision, and has developed into an independent research field [2].

At present, there are many face detection methods. The mainstream method is based on machine learning [3], including algorithms based on neural networks [4-5], Support Vector Machine (SVM) [6-9] and AdaBoost [10-14]. Detection methods based on convolutional neural networks [15-17] have also developed rapidly in recent years. In 2016, Opitz et al. [18] divided the window of the input network into grids and calculated the loss for the grids at each location, which made the convolutional neural network robust for partially occluded faces. Zhang et al. proposed MTCNN [19], whose accuracy on FDDB can reach 95%. However, the detection time based on convolutional neural network is long, and high requirements on the hardware environment are required. The advantage of face detection algorithm based SVM is that it is effective and simple for the binary classification of targets. The disadvantage is that when the number of training samples is too much, the training of SVM requires a lot of storage space and the training speed is slow. The AdaBoost algorithm [11] was proposed by Viola and Jones in 2001 and applied to face detection. It is widely used due to its fast detection speed and low false detection rate. The AdaBoost algorithm generally uses Haar features [20-22] and adopts a cascade method to form a cascaded classifier for detection faces. The disadvantage is that the training feature dimension of each image is very large and the training time is long. Although the construction of the cascade classifier is simple, it will sacrifice a certain speed and accuracy at the expense. At the same time, it is easy to misjudge the background area similar to the gray distribution of human face, and the effect of multi-angle face detection with the horizontal tilt exceeding 60 degrees is not good. In order to improve timeliness, the Comprehensive Color Model (CCM) [23] method was used to extract the skin color area of face in the image and remove the complex non-skin background. Thus, the area of sliding window is reduced during face detection, and the complex background of the face image is simplified. Although the detection speed and accuracy are improved, it is difficult to achieve real-time face detection. Aiming at detecting the tilt angle and real-time of the image, Ojala et al. proposed face detection based on the Local Binary Pattern (LBP) feature [24] in 2002. LBP is used to extract the local features of image [25]. The MB_LBP (Multi-block LBP) feature changed by the LBP feature [26-28] has significant advantages such as rotation invariance [29], so that it can overcome changes in face angle. And for images with the same size, the number of MB_LBP features is much smaller than Haar features, which greatly saves training and detection time. At present, most researches on tilt face detection focus on the horizontal direction and have achieved many results. However, there are relatively few researches on face detection with larger pitch angles, and the performance of detection algorithm needs to be improved.

According to information distribution of face model, especially when face angle is changed, the geometric distribution of face can be extracted well. The improved Haar-T feature is proposed, and the MB_LBP feature is combined to construct classifier in this article. It is proposed to first use image scaling and skin color information to reduce the search face area. And on the basis of satisfying real-time applications, the accuracy of face detection is greatly improved. In order to achieve angle robustness, the AdaBoost algorithm is used to train front face classifier according to the number of samples, and SVM algorithm is used to train tilt face classifiers. Finally, the classifiers are cascaded in series. The experimental results show that on the basis of ensuring the speed and accuracy in this article, the tilt angle range of the face is further enlarged, that is, horizontal tilt angle can reach (-90, +90), and the pitch angle range reaches (-40, +45).

2. Face Detection Algorithm

In order to improve the real-time performance of face detection, scaling images and YCbCr+Ostu skin color extraction are used to narrow the search area of faces. And in order to reduce adverse effects of tilt angle and improve accuracy, MB_LBP+Haar-T features are used to train samples. Taking into account the number of training sample and training time, AdaBoost and SVM are used to train the classifiers separately, and the cascade method is used to connect into a strong classifier. The specific process is shown in Fig. 1.

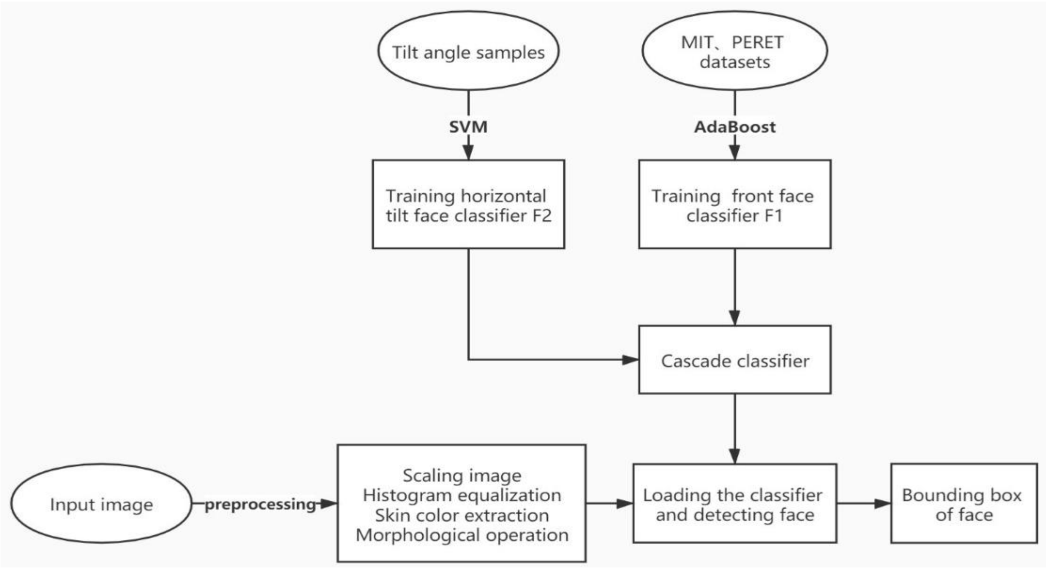

Fig. 1 The flow chart of algorithm

The traditional AdaBoost algorithm uses one feature to train the classifier, so that the detection effect is difficult to break through. Haar feature classification ability is better, but it has a large amount of calculation, long training time, and serious distortion when the weight is updated [30]. MB_LBP has a small number of features, fast training time, and a good detection effect on image pixels, edges, and corners. Therefore, these two features are used to train the classifier. The front face samples are trained by the MIT and FERET datasets (2706+1400 images), 400 horizontal tilt and pitch face images respectively, and 8381 non-face images for negative samples. Images are all grayed and positive samples are normalized to 20*20. The main work is image preprocessing, using AdaBoost algorithm and SVM algorithm to train feature classifier mixed by improved Haar-T feature and MB_LBP and cascade classifier in this article. The image preprocessing includes image scaling, histogram equalization, skin color extraction, and image morphological operations.

2.1 Image Preprocessing

Image preprocessing is related to the accuracy of face detection. Effective preprocessing can not only shorten the detection time but also improve the accuracy. The image preprocessing in this article mainly includes several key steps such as image scaling, histogram equalization, skin color extraction and morphological operation.

2.1.1 Image Scaling

The size used to detect image cannot be too large or too small. Too small will cause the accuracy to decrease, and too large will not achieve real-time performance. Therefore, the image scales first. This method is test on Caltech face database, and normalize images to be detected to (300, 200), which has the best accuracy and real-time performance.

2.1.2 Histogram Equalization

Histogram equalization uses image histograms to adjust contrast in the field of image processing. The method is usually used to increase local contrast in some images, especially when the contrast of the useful data in image is quite close. The histogram equalization effectively expands brightness in common use, so that the brightness can be better distributed on the histogram. The principle is formula (1).

$$S_k = \sum_{j=0}^{k} \frac{n_j}{n} \quad k = 0,1,2\ldots\ldots, L-1 \tag{1}$$

Where n is the sum of pixels in the image, and n_j is the number of pixels in the current gray level. L is the total number of image gray levels.

2.1.3 Skin Color Extraction

In face detection, skin color has better stability than other information, and is not affected by some factors such as posture, expression, and size. Skin color extraction can greatly reduce the face search area and the detection time. Although the skin color of people from different races or ages is different, a large number of experiments have proved that this difference is mainly reflected in brightness, and the skin color has good clustering in the chromaticity space [31]. Since most images detected are RGB, other color models are used to separate image chroma and brightness.

In this article, various skin color models are combined with Haar+AdaBoost and are tested on the face dataset of California Institute of Technology. The results are shown in Table 1.

Table 1 Performance comparison of various skin color models for face detection

Skin color model	Accuracy
HSV	82%
YCbCr	91%
YCbCr+Ostu	95.3%

It can be seen from Table 1 that the YCbCr+Ostu skin color extraction method has higher accuracy than the HSV and YCbCr methods. Therefore, YCbCr color space and adaptive threshold method are used to characterize chrominance difference component Cr of image, ignoring the influence of brightness, which makes the skin color pixels produce better clustering. The conversion from RGB to YCbCr is shown in formula (2) (the value range of RGB: 0-255).

$$\begin{pmatrix} Y \\ C_b \\ C_r \end{pmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ 0.500 & -0.419 & -0.081 \\ -0.169 & -0.031 & -0.500 \end{pmatrix} * \begin{pmatrix} R \\ G \\ B \end{pmatrix} \tag{2}$$

Although human skin color has independent brightness and is concentrated in a small area of Cb and Cr, the error is still large because of using fixed values for skin color segmentation. Therefore, Ostu adaptive threshold segmentation is used in this article. Its calculation is formula (3).

$$p_i = \frac{n_i}{\mathrm{N}} \quad i = 0,1,2\ldots\ldots L-1 \tag{3}$$

Where N is the number of pixels of image, and the gray scale range of image is [0, *L*-1]. n_i represents the number of pixels of gray level *i*, and p_i is the probability of gray level *i*.

$$\sum_{i=0}^{L-1} p_i = 1 \tag{4}$$

A threshold K is used to divide the pixels of image into two classes. The pixel gray value is the background class C_0 when it is less than K, otherwise the target class C_1.

$$u_k = \sum_{i=0}^{L-1} i p_i \tag{5}$$

Thus, the average of C_0 and C_1 is obtained.

$$u_0 = \sum_{i=0}^{k-1} i p_i / w_0 \tag{6}$$

$$u_1 = \sum_{i=k}^{l-1} i p_i / w_1 \tag{7}$$

Where w_0 and w_1 are the probabilities of target classes C_0 and C_1.

$$w_0 = \sum_{i=0}^{k} p_i \tag{8}$$

$$w_1 = \sum_{i=k}^{L-1} p_i = 1 - w_0 \tag{9}$$

Therefore, the average of the entire image is obtained.

$$u_k = w_0 u_0 + w_1 u_1 \tag{10}$$

The variance is derived from the average.

$$\delta_B^2 = w_0 \left(u_0 - u_k\right)^2 + w_1 \left(u_1 - u_k\right)^2 \tag{11}$$

The value K varies from 0 to 255, and the maximum value of the variance calculated by formula (11) is the best threshold segmentation.

Since some pixels in image are similar to skin color and may be detected as skin color, and some images have lots of noise. In order to solve the experimental errors caused by above factors, the eroding and dilating in morphology are used in this experiment.

2.2 Design of AdaBoost Classifier

2.2.1 Training of Weak Classifier

In order to solve the problem of face pitch and horizontal tilt angle, considering that the face model has a large amount of T-shaped structure information, the improved Haar-T feature combined with the MB_LBP feature is proposed to train an optimal classifier in this article.

The Haar-T feature is a rectangular feature with a ratio of black and white width of 1:1. The eigenvalue is that the gray value of image pixel in the white area subtracts the value in the black area within the rectangular box. Haar-T features are divided into four directions, and features are shown in Fig. 2a-d. In order to overcome the influence of face angle, four Haar-T features with a ratio of black and white width of 1:2 are added, as shown in Fig. 2e-f.

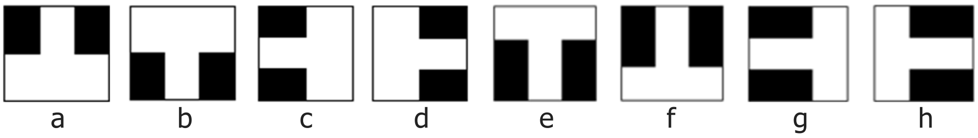

Fig. 2 Haar-T feature

T-shaped features can better extract texture information of human face, and different angles or areas of face are suitable for Haar-T-shaped features of different proportions. As shown in Fig. 3a-d, when image is the front face, the Haar-T feature with a width ratio of 1:1 has good description of the facial texture information, and the gray value of the eyebrow and eye area is larger than the nose and forehead area. Compared with e and m or n and f in Fig. 3, it can be seen that when face looks up or down, the distance between the eyebrows and the nose becomes larger or smaller, and the Haar-T feature with a width ratio of 1:2 can better represent the texture information of face. And compared with g, h, p, q in Fig. 3, when the face is horizontally deviated, the nose is more prominent than eyes. It can be seen that Haar-T feature with a width ratio of 1:2 can better extract the texture of face.

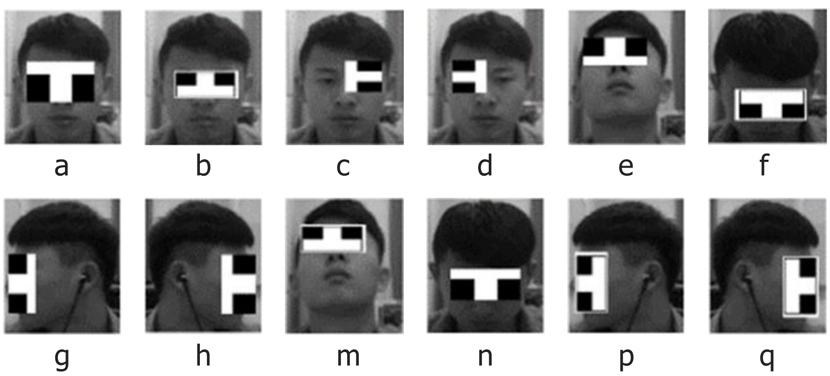

Fig. 3 Matching relationship between Haar-T features and face geometric distribution

The calculation of its eigenvalues is the same as the calculation of Haar features, which uses integral graphs to calculate (T-shaped features are divided into four rectangles: R1, R2, R3, R4).

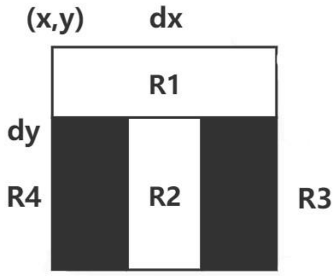

Fig. 4 Haar-T eigenvalue calculation

The eigenvalue of the entire T-shaped feature is R1+R2-R3-R4.

$$R1 = S(x,y) + S(x+3dx, y+dy) - S(x+3dx, y) - S(x, y+dy) \tag{12}$$

The function *S(x, y)* represents the sum of gray levels of pixels on the upper left of pixel *(x, y)*, and the calculation of *R2*, *R3*, and *R4* is same. The calculation process of eigenvalues of Haar-T features in other three directions is similar.

The MB_LBP feature is improved by the LBP feature. The LBP eigenvalue is obtained by encoding relationship of the size between the central pixel and its eight neighborhood pixels, as shown in Fig. 5 (The size of eigenvalue is 254).

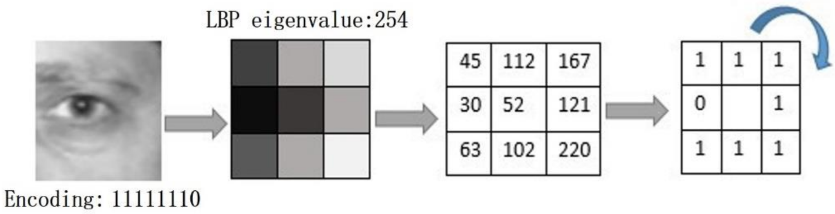

Fig. 5 LBP coding process

The MB_LBP feature is composed of nine LBP features, that is, g_i represents a feature of LBP in Fig. 6. Formula (13) is calculation of MB_LBP feature.

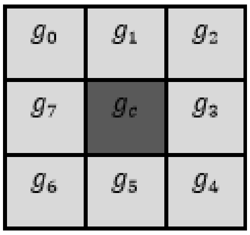

Fig. 6 MB_LBP feature

$$f(g_c) = \sum_{i=0}^{7} s(g_i - g_c) 2^i \tag{13}$$

Where g_i represents the average of gray value in each 3*3 sub-region, and *s(x)* is a sign function.

The value of MB_LBP is composed of eight binary numbers, which is 0-255 when converted to decimal. Therefore, multi-branch tree structure is used to construct a weak classifier in the article. There are 256 branches in total, and the function value of each branch corresponds to the MB_LBP eigenvalue.

$$f(x_i)=\begin{cases} a_0 & \mathrm{x}_i=0 \\ a_i & \mathrm{x}_i=i \\ a_{255} & \mathrm{x}_i=255 \end{cases} \tag{14}$$

x_i represents the eigenvalue of MB_LBP of the i-th sample, which is obtained by formula (12). ∂_j represents corresponding coefficient of determination, and its calculation is formula (15).

$$\mathrm{a}_\mathrm{j}=\frac{\sum_i w_i y_i \delta(x_i)}{\sum_i w_i \delta(x_i)} \tag{15}$$

Where y_i represents the training sample. the positive sample is 1, and the negative sample is -1. If ∂_j is greater than 0, the positive sample with an eigenvalue j is more likely. In the similar way, Haar-T eigenvalues are divided into N regions, and a_j (j=1, 2, 3, …, N) in each region is calculated.

$$a_j=\frac{\sum_i w_i y_i \varphi(x_i)}{\sum_i w_i \varphi(x_i)} \qquad \varphi(x_i)\in(\theta_j-1,\theta_j) \tag{16}$$

The error is calculated as formula (17).

$$e_j=\begin{cases} \sum_i w_i y_i+\sum_i w_i & \mathrm{a}_i<0 \\ \sum_i w_i-\sum_i w_i y_i & \mathrm{a}_i>0 \\ 0 & \mathrm{a}_i=0 \end{cases} \tag{17}$$

Thus, the error of each Haar-T feature is obtained.

$$e_{haar}=e_1+e_2+e_3+\ldots+e_N \tag{18}$$

Formula (19) is the error of each MB_LBP feature.

$$e_{MB\_LBP}=e_1+e_2+e_3+\ldots+e_{255} \tag{19}$$

The training of weak classifier aims to find the best weak classifier, that is, the minimum error and the minimum $f_{Haar(x)}$ or $f_{MB\_LBP(x)}$.

2.2.2 Composition of Strong Classifier

Generally, a strong classifier is composed of T optimal weak classifiers. Two weak classifiers are used in this article, which are $f_{Haar(x)}$ and $f_{MB\_LBP(x)}$ corresponding to Haar-T feature and MB_LBP feature. The composition process is as follows.

Initialize the sample weight y_i=1 (positive sample), w_i=1/2m; y_i=-1 (negative sample), w_i=1/2p (m and p respectively indicate the number of positive and negative samples);

Normalize weight (t=1, 2, 3..., t is the number of iterations);

$$w_{tj}=\frac{w_{ti}}{\sum_{j=1}^{N} w_{tj}} \tag{20}$$

Iterate the training samples to find the best Haar-T feature, make e_{Haar} minimum, generate f_{Haar}, and generate f_{MB_LBP} in the same way (p and q represent the number of Haar and MB_LBP weak classifiers respectively);

If $e_{Haar} \leq e_{MB\_LBP}$, then

$$\begin{cases} f_t(x) = f_{Haar}(x) \\ F_{Haar}(x) = F_{Haar}(x) + f_{Haar}(x) \\ p = p+1 \end{cases} \tag{21}$$

Otherwise,

$$\begin{cases} f_t(x) = f_{MB\_LBP}(x) \\ F_{MB\_LBP}(x) = F_{MB\_LBP}(x) + f_{MB\_LBP}(x) \\ q = q+1 \end{cases} \tag{22}$$

Update weights, in order to save time and prevent distortion, only update the current sample weights which is less than threshold;

$$W_{t+1i} = \begin{cases} W_{ti} e^{y_i f_t(x_i)} & \mathrm{W}_{ti} < WT_t \\ W_{ti} & \mathrm{W}_{ti} \geq WT_t \end{cases} \tag{23}$$

$$\mathrm{WT}_t = \frac{1}{N} \sum_{I=1}^{N} w_i \tag{24}$$

Obtain strong classifier.

$$c(x) = \begin{cases} 1 & \mathrm{F}_{MB\_LBP}(X) + F_{Haar}(x) \geq thresh \\ 0 & \mathrm{F}_{MB\_LBP}(X) + F_{Haar}(x) \leq thresh \end{cases} \tag{25}$$

Where *thresh* is the threshold of strong classifier.

2.3 Design of SVM Classifier

Taking into account the effectiveness of SVM for the two-classification problem and the number of face samples at the horizontal and pitch angle, SVM is used to train the face classifier of horizontal tilt and pitch angle in this article. In order to improve the training and detection time, the Discrete Cosine Transform (DCT) is performed on the image before the feature is extracted, which makes the image dimension decrease, and the MB_LBP feature is selected to train the classifier. The algorithm flow is shown in Fig. 7.

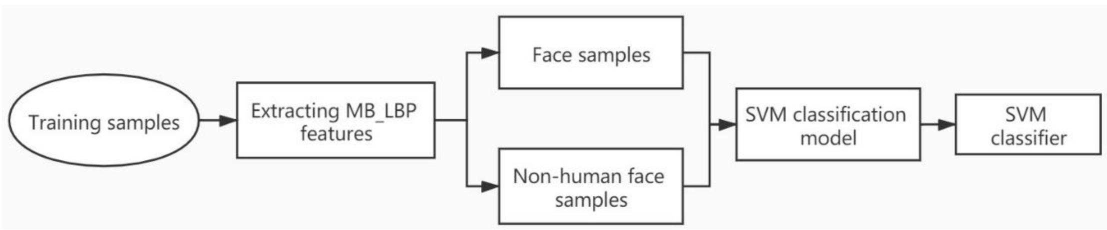

Fig. 7 Training process of SVM classifier

The principle of SVM classification is to find an optimal plane to segment the extracted feature data. The classifier SVM is trained for faces of horizontal tilt and pitch angle, and algorithm process of binary classification SVM is as follows:

Prepare data to be classified and select proper kernel function;

Solve the quadratic optimization equation and get Lagrange operator of SVM;

$$Q(a) = \sum_{i=1}^{n} a_i - \frac{1}{2} \sum_{i,j=1}^{n} a_i\, a_j y_i y_j K(x_i, x_j) \tag{26}$$

Get the optimal classification equation;

$$f(x) = sgn\{\sum_{i=1}^{n} a_i^* y_i K(x_i, x) + b^*\} \tag{27}$$

Solve the class of the sample.

2.4 Cascade Classifier

So as to improve the accuracy of face detection and the robustness of algorithm to angles, in this article, the classifiers are cascaded in parallel, and the training process is shown in Fig. 8.

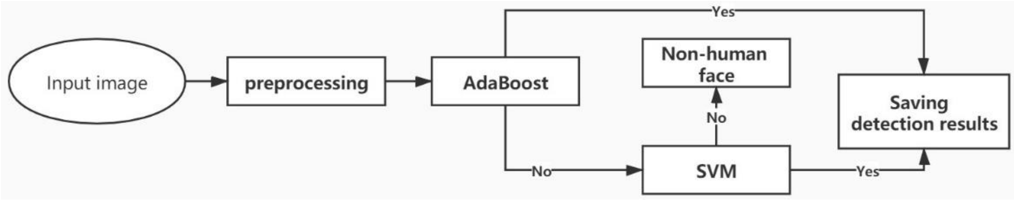

Fig. 8 Cascade classifier

It can be seen from Fig. 8 that the input image is first detected by AdaBoost, if it is not detected, it will be detected by the SVM classifier. This cascading method greatly reduces the detection time while ensuring accuracy.

3. Experimental Data Analysis

The hardware environment of this experiment is i5-5200U CPU@2.20GHz and 8GB memory, and the software environment is Win7 and OpenCV3.2+VS2015. The test dataset includes the face dataset provided by California Institute of Technology, network download and actual images collected, which is a total of 2000 images. The background, expressions, lighting brightness, and the number of faces on each image are different.

3.1 Algorithm Accuracy and Calculation Time

Based on the face database of California Institute of Technology, a comparative experiment is carried out with several typical algorithms, and the results are shown in Table 2.

Table 2 Comparison of face detection performance

Method	Accuracy	False detection rate	Missed detection rate	Average time /ms
Haar	82%	12%	3%	723
Haar+Ostu	95.3%	4.5%	0.2%	650
MB_LBP+Ostu	96.5%	1.8%	1.7%	180
SVM+MB_LBP	94.5%	2.1%	3.4%	150
Haar-T	97.1%	1.8%	1.1%	131
Our	98.9%	0.7%	0.4%	29.7

It can be seen from Table 2 that after skin color extraction, the detection accuracy and efficiency will be improved. Comparisons of several methods in feature extraction capabilities are that Haar-T is the best feature, MB_LBP is second, and Haar is the last. After adding image scaling and a series of preprocessing, the algorithm can meet the needs of real-time applications, and the accuracy of face detection has increased by more than one percentage point.

In terms of accuracy, compared with literature [31] of 78.6%, literature [32] of 83.29% and literature [33] of 82.78%, the algorithm has obvious advantages in this article.

3.2 Face Angle Analysis

In this experiment, 100 images are collected in each angle range, and the algorithm in this article is tested. The test results are shown in Table 3.

Table 3 Test results from different angles

Direction	Horizonal angle			Depression angle				Elevation angle			
Angle	0 -30	30 -60	60 -90	0 -15	15 -30	30 -45	45 -60	0 -15	15 -30	30 -45	45 -60
Accuracy/%	98	97	96	98	95	60	15	98	95	85	30

It can be seen from Table 3 that the horizontal angle does not have much impact on the detection. When pitch angle of face exceeds 45 degrees, the accuracy rate drops more severely. There is main reason that the black and white width ratio of Haar-T feature is always 1:2. When the angle is too large, the improved Haar-T feature value cannot well extract texture information of face.

The algorithm in this article has good robustness for tilt face, and its tilt range is (-90, +90). The maximum angle range of pitch face can reach (-50, +50), and the accuracy can be maintained above 80% within the range of (-40, +45). In addition, the algorithm also has better detection performance for multi-face, multi-angle images, as shown in Fig. 9-11.

Fig. 9 Horizontal tilt detection

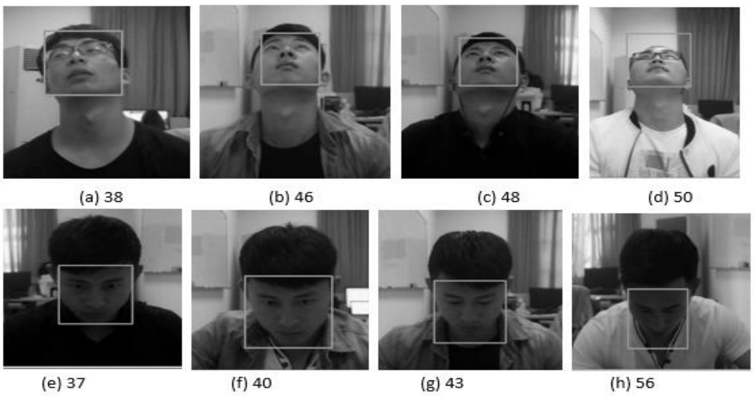

Fig. 10 Pitch detection

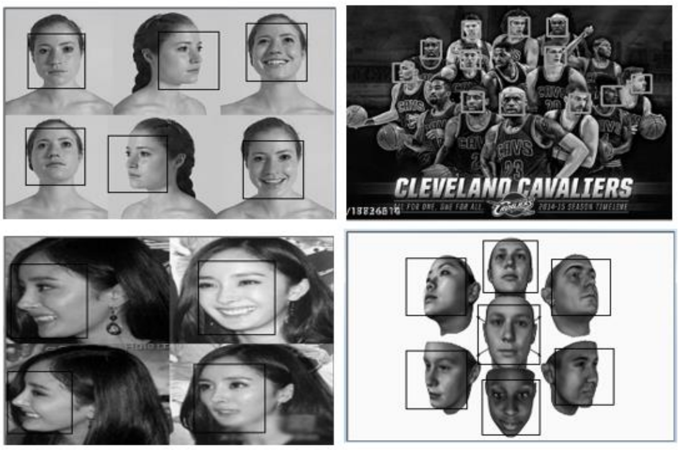

Fig. 11 Multi-face detection

4. Conclusion

In this article, image scaling and skin color extraction are used to reduce the time consumption of detection, and face detection classifier is constructed by improving the Haar-T feature's black and white width ratio, MB_LBP feature, AdaBoost and SVM algorithms to achieve angular robustness. The experimental results show that the algorithm in this article reduces computational time consumption while improving accuracy of face detection, and further expands angle of face to be detected. The detection accuracy can be above 96.5% in horizontal direction (-90, +90) from left to right and 80% in pitch angle (-40, +45) from bottom to top. The follow-up work is to study the detection of partially occluded face and face recognition on this basis.

Acknowledgements

This paper was financially supported by the Natural Science Foundation of Shanghai, China, grant number 14ZR1419700. The authors gratefully thank the science team of Information and Communication System Engineering Lab of College of Information Engineering, Shanghai Maritime University for their general support.

References

[1] LIANG Luhong, AI Haizhou and XU Guan-bian, "A Review of Face Detection Research", Journal of Computers, 2002, Vol. 25 (05), p449-458.

[2] LI Yan and WANG Ling, "Face detection based on skin color and Haar variance", Computer Science and Engineering, 2015, Vol. 37 (01), p146-151.

[3] YAN Bin and Liang Lanzhen, "Application of improved skin color extraction method in face detection", Computer Simulation, 2014, Vol. 31(10), p354-357.

[4] Rowley, Henry A, Shumeet Baluja and Takeo Kanade, "Neural network-based face detection", IEEE Transactions on pattern analysis and machine intelligence, 1998, Vol. 20(1), p23-38.

[5] Hinton, Geoffrey E and Ruslan R. Salakhutdinov, "Reducing the dimensionality of data with neural networks", science, 2006, Vol. 313(5786), p504-507.

[6] Osuna, Edgar, Robert Freund and Federico Girosit, "Training support vector machines: an application to face detection", Proceedings of IEEE computer society conference on computer vision and pattern recognition, 1997, p130-136.

[7] Tao Qin-Qin, Zhan Shu and Li Xiao-Hong, "Robust face detection using local CNN and SVM based on kernel combination", Neurocomputing, 2016, Vol. 211, p98-105.

[8] Kumar, Sandeep, Sukhwinder Singh and Jagdish Kumar, "Multiple face detection using hybrid features with SVM classifier", Data and Communication Networks, 2019, p253-265.

[9] Soula Arbia, Tbarki Khaoula and Ksantini Riadh, "A novel incremental Kernel Nonparametric SVM model (iKN-SVM) for data classification: An application to face detection", Engineering Applications of Artificial Intelligence, 2020, Vol. 89, p103468.1-10.

[10]DU Cui-huan, ZHU Hong and LUO Li-ming, "Face detection in video based on AdaBoost algorithm and skin model", The Journal of China Universities of Posts and Telecommunications, 2013, Vol. 20, p6-24.

[11]Viola Paul and Michael Jones, "Rapid object detection using a boosted cascade of simple features", Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition, 2001, Vol. 1, pI-I.

[12]Chen Yan, Wang Zhengqun and Xu Chunlin, "Gentle Adaboost algorithm based on multi-feature fusion for face detection", The Journal of Engineering, 2019, Vol. 2019(15), p609-612.

[13]Li Zhi-Zhen, Zeng Qing-Hua and Li Xiang-Dong, "Face Detection Technology Based on Combining Skin Color Model with Improved Adaboost Algorithm", 2019 IEEE 4th International Conference on Signal and Image Processing, 2019, pp. 381-384.

[14]Vansh, Vishnu, Chandrasekhar Kumar and Sekhar Sahu Sitanshu, "Improved Face Detection Using YCbCr and Adaboost", Computational Intelligence in Data Mining, 2020, pp.689-699.

[15]Li Haoxiang, Lin Zhe and Shen Xiaohui,"A convolutional neural network cascade for face detection", Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 5325-5334.

[16]Zhang Shifeng, Wen Longyin and Shi Hailin, "Single-shot scale-aware network for real-time face detection", International Journal of Computer Vision, 2019, Vol. 127(6-7), p537-559

[17]Zhang Jialiang, Wu Xiongwei and Hoi Steven C.H, "Feature agglomeration networks for single stage face detection", Neurocomputing, 2020, Vol. 380, p180-189

[18]Opitz Michael, Waltner Georg and Poier Georg, "Grid loss: Detecting occluded faces", In European conference on computer vision, 2016, p386-402

[19]Zhang Kaipeng, Zhang Zhanpeng and Li Zhifeng, "Joint face detection and alignment using multitask cascaded convolutional networks", IEEE Signal Processing Letters, 2016, Vol. 23(10), p1499-1503

[20]Zhu Jie and Chen Zhiqian, "Real time face detection system using adaboost and haar-like features", In 2015 2nd International Conference on Information Science and Control Engineering, 2015, p404-407

[21]Chang Zheng, Ban Xiaojuan and Wang Yu, "Fatigue Driving Detection Based on Haar Feature and Extreme Learning Machine", The Journal of China Universities of Posts and Telecommunications, 2016, Vol. 23(4), p91-100

[22]Songyan Ma and Lu Bai, "A face detection algorithm based on Adaboost and new Haar-Like feature", In 2016 7th IEEE International Conference on Software Engineering and Service Science, 2016, p651-654

[23]ZHANG Hongming, Zhao Debin and Gao Wen, "Planar Rotation Face Detection Based on Skin Model, Neural Network and Face Structure Model", Journal of Computer China, 2002, Vol. 25 (11), p1250-1256

[24]Ojala Timo, Pietikäinen Matti and Harwood David, "Performance Evaluation of Texture Measures with Classification Based on Kullback Discrimination of Distributions", Proceedings of the 12th IAPR International Conference on Pattern Recognition, 1994, Vol. 1, p582-585

[25]LIU Hao, Yang Yongquan and Guo Xiancao, "An improved LBP algorithm for texture feature extraction", Computer Engineering & Applications, 2014, Vol. 50 (6), p182-185, 245

[26]Liao Shengcai, Zhu Xiangxin and Lei Zhen, "Learning Multi-Scale Block Local Binary Patterns for Face Recognition", International Conference on Advances in Biometrics, 2007, Vol. 4642, p828-837

[27]Halidou Aminou, You Xinge and Mahamadou, "Fast Pedestrian Detection Based on Region of Interest and Multi-Block Local Binary Pattern Descriptors", Computers & Electrical Engineering, 2014, Vol. 40(8), p375-389

[28]Dandashy Tarek and Amine Bitar, "Enhanced Face Detection Based on Haar-Like and MB-LBP Features", International Journal of Engineering and Management Research, 2019, 9(4), p117-124

[29]LI Gen and Li Wenhui, "Planar rotation face detection based on LBP with principal direction rotation", Journal of Electronics, 2015, Vol. 41 (1), p198-202

[30]PAN Qiuping, YANG Wanguang and SUN Chang-yin, "A license plate detection algorithm based on Haar and MB-LBP", Journal of Southeast University (Natural Science), 2012, Vol. 42 (S1), p74-77

[31]Hsu Rein-Lien, Mohamed Abdel-Mottaleb and Anil K. Jain, "Face detection in color images", IEEE transactions on pattern analysis and machine intelligence, 2002, Vol. 24(5), p696-706

[32]Qi Yongfeng, Zhang Jishu and Huo Yuanlian, "A method of face detection based on skin color and improved LBP", Photol.LaChng, 2009, Vol. (6), p816-821

[33]LU Yi-xing, LIU Ying and GUO Yu, "Features of Human Skin in HSV Color Space and New Recognition Parameter", Optoelectronics Letters, 2007, Vol. 3(4), p312-314