Research and Improvement of Face Detection Algorithm Based on the OpenCV

Min Zuo
College of Computer and Information Engineering
Beijing Technology and Business University
Beijing, China
zuomin1234@163.com

Min Zuo,Guangping Zeng and Xuyan Tu
School of Information Engineering
University of Science and Technology Beijing
Beijing, China
zuomin@th.btbu.edu.cn

***Abstract*—This article describes functions and advantages of the OpenCV library, and explains the meaning , development status, applications, and difficulties of face detection technology, analysis the idea of Adaboost classifier algorithm ,and achieves the detection procedures for face using OpenCV, and proposes the improved method based on the the original algorithm.**

Keywords-Face detection; OpenCV; Adaboost algorithm;Classifier

I. INTRODUCTION

Face detection is that some strategy is adopted to search and determine whether it contains human face in any given image. And the location, size and attitude of the human face will be returned if it can be found. In recent years[1], Face detection technology as a key technology of human face information processing has attracted universal attention and become a very active research topic in the field of pattern recognition and computer vision.

At present, there are many research institutions studying on face detection technology at home and abroad, such as MIT, CMU at abroad and Tsinghua University, Beijing University of Technology, Computing Institute and automation Institute of Chinese Academy of Sciences at home where there are many researchers engaged in both face detection and related. Moreover, MPEG7 Standard Organization has established draft committee in face recognition and face detection algorithm is also a collection content. With the intensive study on face detection technology, the number of the relevant papers internationally published substantial increased. There are a large number of papers on face detection which represent 1/3 as much in related papers every year in the important international conferences, such as FG of IEEE, ICIP\CVPP, etc. From this we can see that the world attach great importance to human face detection technology.

Face detection is a complex and challenging pattern detection problem, and the main difficulties have two aspects.

On the one hand, it is caused by inner changes of the face:

1）the human face has rather complicated detail changes. For example, a different appearance, such as shape of face, skin color, etc., and different expressions, such as open and close of the eyes and mouth, etc.;

2） block of the face, such as glasses, hair and head accessories, as well as other external objects, and so on.

On the other hand, it is caused by changes of external conditions:

1）multi-gesture of face is caused by different imaging angle, such as the plane rotation, deep rotation, up and down spin, in which deep rotation has a greater impact;

2） impact of light, such as brightness, contrast changes and shadows of the image;

3） imaging condition of the image, such as focal length of the camera equipment, imaging distance, means of the image obtained and so on.

These difficulties increased the difficulty of the face detection. If we can find some practical algorithms to filter out the noise generated by the above-mentioned cases, then we will be possible to construct face detection and tracking system which has practical application value.

II. ADABOOST ALGORITHM

For any object in the image, to detect it, we must get something which is called feature in image processing to separated from other objects, such as color, grey value, texture, contour, etc. If you want to distinguish one object from others through features, it will involve how to organize various concerned features, and Adaboost is a well method of organized features in the target detection.

Target detection method is originally developed by Paul Viola [Viola01] and improved by Rainer Lienhart [Lienhart02] then. First, a cascade boosted classifier can be gotten through haar feature classifier training of samples (about a few hundred pieces of sample images). Training samples can be divided into positive sample and negative sample, in which positive sample is the target sample (such as face and cars and so on), and negative sample is other arbitrary images. All the samples will be normalized into the same size (for example 20*20).

After training, classifier can be used to interested region (same size with training sample) detection for input image. Classifier will output 1 if the target area (such as faces) is detected, otherwise the output is 0. In order to detect the entire image, we can move the search window in the image, testing each location to identify possible targets. In order to search different size object, classifier is designed that the size can be changed, so it is more effective than to change the image size. Therefore, in order to detect unknown size target in the image, the scanning program usually scan the image some times with different size search window.

Cascading in the classifier refers that the final

978-1-4244- 7618-3 /10/$26.00 ©2010 IEEE

classifier is composed of a few simple classifier. In the image detection, the windows tested will pass each level classifier in turn, so that most of the candidate region is filtered out in front layers. And the region detected which through each classifier shall be the target area.

The implementation principle of face detection program in this paper is based on Adaboost classifier algorithm. Its idea is that if one thing can not be separated by one feature, it can be done by a gradually enhanced classifier which is combined by some features. So Adaboost is a combination of weak classifiers. Adaboost algorithm is improved from original Boosting algorithm. It can achieve the same efficiency, and it can also improve the detection rate and reduce the number of false detection.

In the concrete realization, a large number of basic classifiers with weak detection capacity will be boosted by means of some method to construct a strong classifier with strong detection capacity. Then some strong classifiers are concatenated into a classifier cascade to complete the image search detection. The level number depends on system's requirements for error rate and recognition speed.

III. FACE DETECTION WITH OPENCV ALGORITHM[2]

A. **Flow chart**

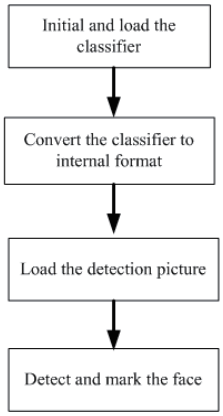

Fig. 1. Flow chart of the detection program

B. **The Core code**

```
void detect_and_draw( IplImage* img )
{
   static CvScalar colors[] = {

{{0,0,255}},{{0,128,255}},{{0,255,255}},{{0,255,0}},

{{255,128,0}},{{255,255,0}},{{255,0,0}},{{255,0,255}}
}
    };
   double scale = 1.3;
   int i;
   IplImage* gray = cvCreateImage(
            cvSize(img->width,img->height), 8, 1 );
   IplImage* small_img = cvCreateImage(
            cvSize(cvRound (img->width/scale),
          cvRound (img->height/scale)),8, 1 );

   cvCvtColor( img, gray, CV_BGR2GRAY );
   cvResize( gray, small_img,
CV_INTER_LINEAR );
   cvClearMemStorage( storage );
   if( cascade )
   {
  double t = (double)cvGetTickCount();
  CvSeq* faces = cvHaarDetectObjects( small_img,
        cascade, storage, 1.1,2,0,cvSize(15, 15) );
  t = (double)cvGetTickCount() - t;
  printf("detection time = %gms\n",

t/((double)cvGetTickFrequency()*1000.));
  for( i = 0; i < (faces ? faces->total : 0); i++ )
  {
    CvRect* r = (CvRect*)cvGetSeqElem( faces, i );
    CvPoint center;
    int radius;
    center.x = cvRound((r->x + r->width*0.5)*scale);
    center.y = cvRound((r->y + r-
>height*0.5)*scale);
    radius = cvRound((r->width + r-
>height)*0.25*scale);
    cvCircle( img, center, radius, colors[i%8], 3, 8,
0);
   }
   }
  cvShowImage( "result", img );
  cvReleaseImage( &gray );
  cvReleaseImage( &small_img );
}
```

C. **Algorithm analysis**

Implementation tool of this procedure is win32 console program of VC++, and OpenCV libraries should be installed and loaded before running. The code above is the core function program to achieve the face detection, and it includes four phases (figure 1) of entire detection process, in which calls some functions of the OpenCV library.

Function cvHaarDetectObjects() will detect the target in the image, and the specific function prototype is:

```
CvSeq* cvHaarDetectObjects( const CvArr* image,
    CvHaarClassifierCascade* cascade,
    CvMemStorage* storage, double scale_factor=1.1,
    int min_neighbors=3, int flags=0,
    CvSize min_size=cvSize(0,0) );
```

Function cvHaarDetectObjtcts uses a cascade classi fier training against some object to find a rectangle region containing the target object, and returns these regions as a sequence of rectangle. Function search the image several times with different size scanning windows (referring to cvSetImagesForHaarClassifier Cascade).Function is used to detect target in the ove rlap regions of the image every time. Sometimes, w e also use certain succession to reduce candidate reg

ions, like using the reductions method of Canny. After handling and collecting the candidate rectangles (after through the each level of cascade classifier), the function will combine these regions and return a series of average rectangle from each combination large enough. The default para-meters (scale_factor=1.1, min_nerghbors=3, f-lags=0) in the program will be regulated, and it will be used to further detected, which is more accurate but also time-consuming. In order to achieve more accurate real-time detection for the video image, the program parameter is set as: scale_factor=1.1, min_neighbors=2, flags=0, min_size=cvSize(15, 15)[1-3].

The combination of function cvGetTickCount() and cvGetTickFrequency() achieves that clock is used to count the detection time.

Function cvCircle() will identify each detected target region, namely to mark the faces with circles, in which the previously defined array cvScalar colors[] is used to store a series of color data, and these can replace the face mark color.

D. Results

Fig. 2(a). Before program is improved

Fig. 2(b). After program is improved

IV. ALGORITHM IMPROVEMENT

From the detection results in Figure 2(a) and Figure 3(a) above, we can see that function cvHaarDetectObjects() using Adaboost classifier can reach a very high detection rate, but it can not make a complete identification and none false detection because of the exposure image itself and so on. In order to further improve the detection rate and reduce the number of false detection, some image processing should be done in the early stage to filter noise. Smooth processing combined with histogram equalization is used in this procedure[3-4].

First, function cvSmooth(gray,gray,CV_BILATERAL,3,3,0,0) will be used to smooth the image.

Image smoothing will handle the gray value abnormal point to get more coordinated image. Function cvSmooth() smooth the image with a certain filtering method. This paper uses CV_BILATERAL(bi-directional filtering), that is, using two-way 3*3 filter, colorful sigma = param1, space sigma = param2. Then next step is histogram equalization which uses function cvEqualizeHist(small_img, small_img);

Function cvEqualizeHist() will do histogram equalization, normalized image brightness and enhanced contrast of input images. Histogram equalization is a method that using image histogram to adjust the contrast in the field of image processing. And it is often used to increase local contrast without affecting overall contrast, especially when the contrast of the useful data in the image is very close. After the equalization, the brightness will be better distributed on the histogram. This method is very useful for the background and prospects too bright or too dark. In particular, it can lead to a better skeletal structure show in X-ray image and better details in the overexposed or underexposed pictures. It is a very effective image pre-processing methods[5-7].

In order to improve the efficiency of the Adaboost algorithm, skin color detection technology will be adopted to roughly find face region and to reduce search area of Adaboost algorithm.

In order to rapidly and accurately extract the color pixels, we use the skin color pixel polynomial model[8] proposed by Soriano. The polynomial models do not need a complex matrix calculation and can reduce the computational complexity. Polynomial model consists of two r-g plane quadratic polynomial equation and a circle pose:

$$f_{upper}(r) = a_u r^2 + b_u r + c_u \quad (1)$$

$$f_{lower}(r) = a_l r^2 + b_l r + c_l \quad (2)$$

$$W(r,g) = (r-0.33)^2 + (g-0.33)^2 \quad (3)$$

r=R/(R+G+B), g=G/(R+G+B), and R, G, B respectively stand for red, green and blue components in RGB space. After turning the RGB color space into r and g, impact of brightness on the color pixels will be reduced, so that color pixels has better aggregation feature. Coefficient in the formula (1) and (2) are -1.38, 1.07, 0.145, 0.776, 0.56 and 0.177, respectively. We can know from literature [9] that r component r of color pixel is between 0.2 and 0.6 in the RGB space. Through the formula (1), (2) and (3), we can extract the color region through the rules R1, R2, R3 and R4:

R1: $g > f_{lower}(r)$ and $g < f_{upper}(r)$

R2: $W(r, g) \geq 0.004$

R3: $r > 0.2$ and $r < 0.6$

R4: $R - G > 20$

Through the pre-processing and face skin color detection above, detection results of the improved procedures are shown in Figure 2(b) and Figure 3(b).

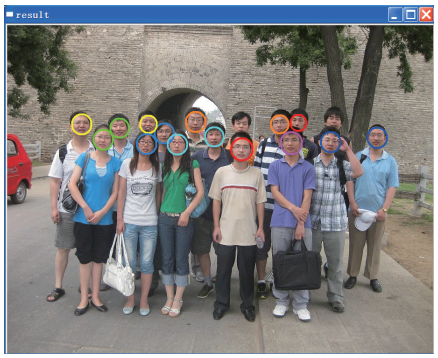

Fig. 3(a). Before program is improved

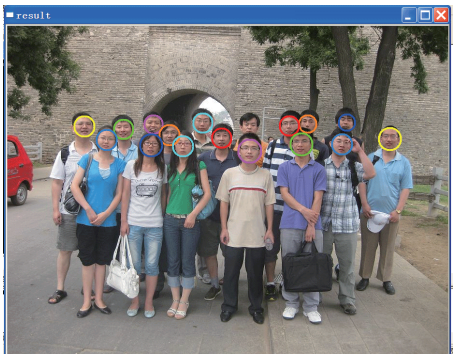

Fig. 3(b). After program is improved

Table 1 The comparative table of detection results before and after program is improved

	Figure 2		Figure 3	
	Detection rate	error	Detection rate	error
Before	14/15	1	15/17	0
After	14/15	0	16/17	0

Table 2 The comparative table of detection speed before and after program is improved

Detection time	Figure 2	Figure 3
Before	0.224ms	0.218ms
After	0.177ms	0.172ms

According to Table 1 and Table 2, we can see that through the pre-processing and face skin-color aid detection of image can improve face detection rate, lower false detection rate and improve detection efficiency.

V. Summary

In this paper, OpenCV combined with Adaboost algorithm is used in the face detection. Corresponding detection procedure is completed and some of them are analyzed. To improve the detection procedure by doing some image pre-processing and face skin-color aid detection, comparison is made before and after improvement. Experimental results show that improved procedure is effective for improving the face detection rate, lowering false detection rate and improving detection efficiency, etc.

ACKNOWLEDGMENT

This work is supported by Beijing Natural Science Foundation of China (No. 4082021) and the National Programs for High Technology Research and Development of China (863 Key Project, No. 2008AA018308) and the National Natural Science Foundation of China (No.60375038) and the Beijing Municipal Natural Science Foundation (No.4072018) for partially supporting our research work and Scientific Research Common Program of Beijing Municipal Commission of Education， KM201010011006 and PHR(IHLB)，PHR201008242

REFERENCES

[1] Li G, Gao ZH.Overview of Automatic Human Face Recognition, Computer Application Research. 2003.8:4-9.

[2] Liu R ZH, Yu SH Q. OpenCV Course-The Basics, Beijing University of Aeronautics and Astronautics Press, 2007.6.

[3] Wu Y, Zhang Ch Y. Grey Histogram Re-distribution Image Enhancement Algorithm Based on Objective Characteristics[J]. Computer & information techmology, 2007.6:14-15,19.

[4] Open Source Computer Image Library Reference Mannual.2001(8).

[5] Ai H Zh. Face detection and Retrieval[J], Journal of Computers, 2003,(7):874-881.

[6] LiXY. Real2time image histogram equalization using FPGA [J]. SPIE, 1998, 3561:293-299.

[7] Yang M H, Kriegman D, Ahuja N. Detecting faces in images: A survey, IEEE Trans. on Pattern Analysis and Machine Intelligence, 2002, 24(1):34-58.

[8] SOR IANO M, HUOV INEN S. Using the skin lotus to cope withchanging illunination conditions in color2based face tracking[C].IEEE Nordic Signal Processing Symposium. Kolmarden, Sweden:IEEE, 2000: 383 - 386.

[9] CH IANG C C, TA IW K. A novel method for detecting lip s, eyes and faces in real time[J]. Real2Time Imageing, 2003, 9 (4) : 277 - 287.