ADABOOST-BASED EYE DETECTION

Jiashen Zhang, Luting Wang, Gary Overett *

Sun Yat-sen University–Carnegie Mellon University, SYSU-CMU Joint Institute of Engineering

*Email: overett@mail.sysu.edu.cn

Joint Institute of Engineering
SUN YAT-SEN UNIVERSITY
Carnegie Mellon University

Jiashen zhang Luting Wang

Introduction

Human face image analysis, detection and recognition have become some of the most important research topics in the field of computer vision and pattern classification. Among these research topics, one fundamental but very important problem to be solved is automatic eye detection. One promising approach is to use Haar features and AdaBoost algorithm to get cascade classifiers. In this project, first we will see some basics of face detection using Haar Feature-based Cascade Classifiers, which is an effective object detection method proposed by Paul Viola and Michael Jones in 2001[1]. Then we will extend the same to eye detection.

There are several steps to achieve it:

- Extract Haar-like features
- Use Integral Image to speed up calculation
- Train strong classifier use Adaboost Algorithm
- Cascade strong classifiers together
- Sliding window to detect

Methods

- Extract Haar-like features

Initially, we have a lot of positive images and negative images. Then we need to extract features from it. Haar features are shown in Figure 1. Each feature is a single value obtained by subtracting sum of pixels under white rectangle from sum of pixels under black rectangle. Given a 24x24 window, it will result in over 160000 features. So integral images is introduced to simplify and speed up calculation.

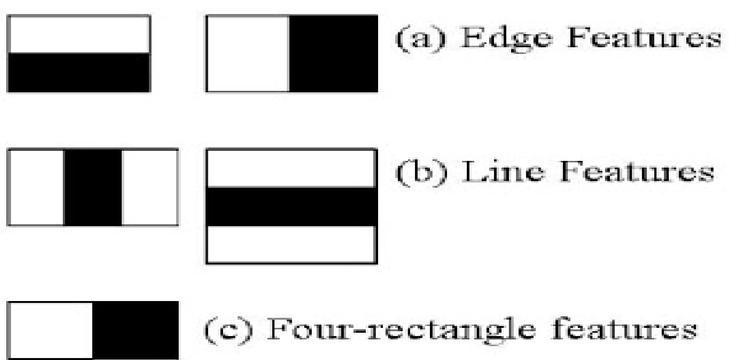

Figure 1. Haar features

- Use Integral Image

The Integral Image is used as a quick and effective way of calculating the sum of values in a given image. Figure 2 shows how it works.

Original Image

3	8	2	1
6	3	9	7
5	2	4	9
6	0	1	8

Sum area table

3	11	13	14
9	20	31	39
14	27	A 42	59 B
20	33	49 D	74 C

Figure 2. Integral Image

As shown in the figure above, to find the sum of the highlighted parts, sum = C+A-B-D. So we have sum = 22.

- The Adaboost Algorithm

Consider the image below. Top row shows two good features. The first feature selected seems to focus on the property that the region of the eyes is often darker than the region of the nose and cheeks. The second feature selected relies on the property that the eyes are darker than the bridge of the nose. But the same windows applying on cheeks or any other place is irrelevant. So we should select the best features, and it is achieved by Adaboost Algorithm.

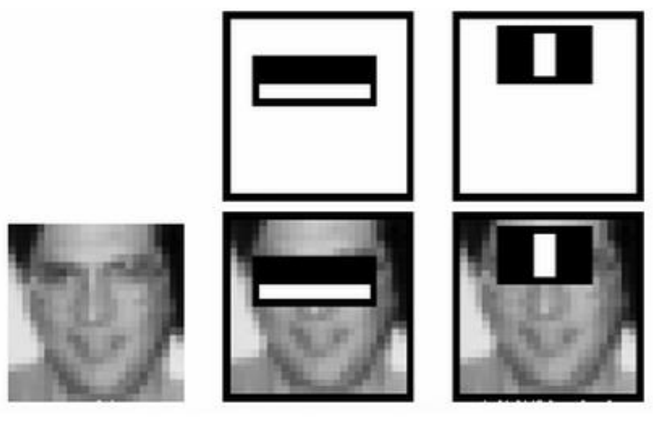

Figure 3. Good Haar features of eyes

- For each feature, find the best threshold classifies faces to positive and negative, select the feature with minimum error rate.
- For each iteration, pick the optimal weak classifier with the lowest error rate, and update sample's weight
- Final classifier is a weighted sum of these weak classifiers

- Cascade strong classifiers

In an image, most of the image region is non-face. So it is better to have a simple method to check if a window is a face region or not. If it is not, discard it in a single shot. It is achieved by Cascade of Classifiers. It is illustrated in the following figure.

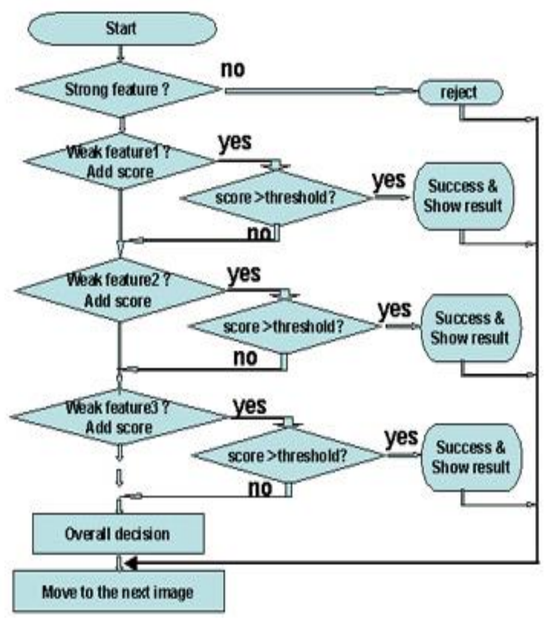

Figure 4. Recognition

Results

Figure 5. result

Conclusions

In this project, we reviewed one eye detection method based on Haar-like features. We implemented it in OpenCV, and the result is shown in Figure 5.

References

[1] Viola P, Jones M. Rapid object detection using a boosted cascade of simple features
[2] Viola P, Jones M J. Robust real-time face detection
[3] Face Detection using Haar Cascades, OpenCV-Python Tutorials
[4]T. Liao et al. "Structure-Aligned Guidance Estimation in Surface Parameterization Using Eigenfunction-based Cross Field." Graphical Models

Address: No. 132, East Waihuan Road, Guangzhou Higher Education Mega Center, Guangzhou, 510006, P.R. China

Tel: 86-20-39943579; 86-20-39943581
E-mail: sysucmu@mail.sysu.edu.cn