Face Recognition Using Haar Cascade and HOG

[1] Prof. Dr Vijay Gaikwad, [2] Ankur Bhattad, [3] Shraddha Gupte, [4] Anant Gavali
[1]Associate Professor and Head, [2,3,4] Research Scholar
[1,2,3,4]Department of Electronics Engineering,
[1,2,3,4] Vishwakarma Institute of Technology, Pune, India

Abstract : Face recognition from image or video is a popular topic in biometrics research. Many public places usually have surveillance cameras for video capture and these cameras have their significant value for security purpose. It is widely acknowledged that the face recognition has played an important role in surveillance system as it doesn't need the object's cooperation. The actual advantages of face based identification over other biometrics are uniqueness and acceptance. As human face is a dynamic object having high degree of variability in its appearance, that makes face detection a difficult problem in computer vision. In this field, accuracy and speed of identification is a main issue. In this paper we have introduced a technique to recognize faces using Adaboost Haar cascade classifier and HOG descriptor. We have achieved an accuracy of 84% for face recognition over a database of 200 images.

***Index Terms* - Haar, Adaboost, HOG, face detection, face recognition.**

I. Introduction

In the past decades, many researches have started with the aim of teaching the machine to recognize human faces and facial expressions. The need to extract information from images is enormous. Face detection and extraction as computer vision tasks have many applications and have direct relevance to the face-recognition and facial expression recognition problem. Face detection is the first stage towards automatic face recognition. Potential applications of face recognition and extraction are in human computer interfaces, surveillance systems, census systems and many more.

The research was divided into two stages first was face detection and second was face recognition. To reduce the code complexity and for the implementation on small single board computers we used Haar Cascade Classifier given by Viola and Jones [1]. The second stage comprises recognizing the detected faces using HOG descriptor. 128-d vector embedding were made for each database of face and then compared during the runtime for the provided test images.

An accuracy of 84% was achieved a with an average time per image of 0.45 seconds for recognition.

II. Methodology

2.1 Face Detection

Harr Cascade classifier was used for the face detection, the default frontal face Haar file was used which can be found on the standard GitHub directory for Haar features. The classifier was tested with images of different light conditions and expressions. Majority of the detection were correct but some false occurring were found too.

2.2 Face Recognition

A face recognition library built by Davis king and trained over the standard dataset "Labeled Faces in the Wild" achieved an accuracy of 99.38%, was used. The library itself has a base of Dlib's state-of-the-art face recognition built with deep learning. Python as a syntactical language was used as it provided various modules and low code complexity.

III. Face Detection

The Viola-Jones face detector contains three main ideas that make it possible to build a successful face detector that is able to run in real time:

- Haar-like features: Offers a new display of image called "Integral Image" which enables Haar-like features to be computed rapidly.
- Adaboost Learning Algorithm: Offers a learning algorithm based on Adaboost, which is able to select few Haar-like features from a larger set and will be transformed into a suitable classifier.
- Cascade Classifier: Offers a method for combining simple to complex Adaboost classifiers increasingly, which enables background regions of image to be discarded by primary classifiers while more complicated Adaboost classifiers are required for the face-like regions.

3.1 Haar Like Features

Haar-like features are digital image features used in object recognition. They owe their name to their intuitive similarity with Haar wavelets and were used in the first real-time face detector.

An object that has to be detected is described by a combination of a set of simple Haar-wavelet like features shown in Fig. 1. The sums of pixels in the white boxes are subtracted from the sum of pixels in the black areas. The advantage of using these simple features is that they can be calculated very quickly by using "integral image".

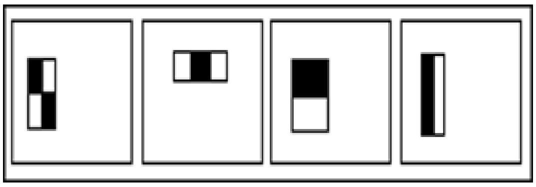

Figure 1 Haar Features

3.2 Units

Adaboost training process selects only those features known to improve the predictive power of the model, reducing dimensionality and potentially improving execution time as irrelevant features need not be computed. It focuses on classification problems and aims to convert a set of weak classifiers into a strong one.

The final equation for classification can be represented as :

$$F(x) = \text{sign}(\sum \theta_m f_m(x)) \qquad (1)$$

Here f(x) are the weak classifiers and theta is their weight. The computational sum of all the weak classifiers results in a strong classifier.

3.3 Cascade Classifier

The cascade classifier consists of a collection of stages, where each stage is an ensemble of weak learners. The weak learners are simple classifiers called *decision stumps*. Each stage of the classifier labels the region defined by the current location of the sliding window as either positive or negative. *Positive* indicates that an object was found and *negative* indicates no objects were found. If the label is negative, the classification of this region is complete, and the detector slides the window to the next location. If the label is positive, the classifier passes the region to the next stage, finally reaching the final stage and having a positive output we have our object detected. This helps in enabling real time processing.

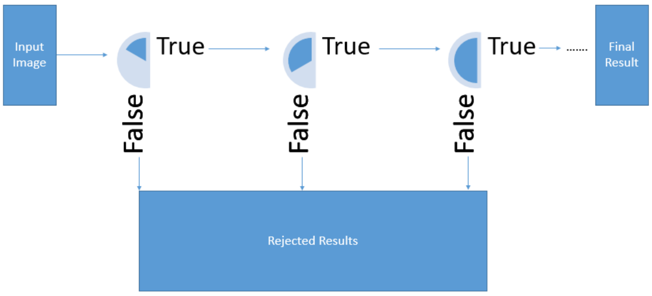

Figure 2 Cascade Classifier

IV. Histogram Of Oriented Gradient (HOG)

The basic idea of HOG features is that the local object appearance and shape can often be characterized rather well by the distribution of the local intensity gradients or edge directions, even without precise knowledge of the corresponding gradient or edge positions. The orientation analysis is robust to lighting changes since the histogram gives translational invariance. The HOG feature summarizes the distribution of measurements within the image regions and is particularly useful for recognition of textured objects with deformable shapes. The method is also simple and fast so the histogram can be calculated quickly.

The original HOG feature is generated for each key point of an image. The neighbouring area around each key point is divided into several uniformly spaced cells and for each cell a local 1-D histogram of gradient directions or edge orientations is accumulated over all the pixels of the cell. The histogram entries of all cells around one key point form the feature of that key point. The combined histogram features of all key points form the image representation.

V. Architecture

5.1 Pre-processing Techniques

The faces in input images maybe have different size from training face templates, therefore for each input image, all features are scaled. In addition, all images areas are searched in order to find the desired face image. These two processes increase calculation time. In order to reduce calculation time in colour images, it recommended to convert the color image to gray images.

5.2 Encodings

128-d vector encodings are created using different images in the database. At the end a single encoding is generated per person, iterating over the images the weight of different vector units is tweaked and a 188-d vector encoding is generated, this process is then repeated for all the persons in the database and a final encodings file is generated which is then used to recognize faces.

5.3 Dataset

A dataset was created using a "Bing search API", this dataset includes 202 images for training (creating the encodings) another 66 images are used for testing purposes. The Bing image search API is part of azure cloud tools, it scrapes over the internet to find the images with the given name and stores it at pre-defined location.

VI. Results

6.1 Hardware Specification

Processor : AMD A8-7410 APU
Graphic Card : Radeon R5 Graphics
Clock Speed : 2.2 Ghz
Operating System: Windows-10 64 bit
Memory : 8-GB

6.2 Encodings

No. of Images : 202
Time Taken : 15 minutes
Average Size
Of Images : 150 KB

6.3 Person 1 :

Table 1 Success/Failure Result Table for Data Set I

Sr No.	False Result	Time Taken
1	no	14575
2	no	50569
3	yes	795270
4	no	126174
5	no	964018
6	no	98747
7	no	329390
8	no	526741
9	no	150143
10	yes	717300
11	no	942642
12	no	283412
13	no	677314
14	yes	526061
15	no	38678
16	no	522236
17	no	552952

Average Time Taken : 430366 microseconds
No. of False result : 3

6.4 Person 2 :

Table 2 Success/Failure Result Table for Data Set II

Sr. NO	False Result	Time Taken
1	no	252605
2	no	83460
3	no	961154
4	no	209462
5	yes	406288
6	no	823076
7	no	201980
8	no	397881
9	yes	296877
10	no	925675
11	no	939977
12	no	549916
13	no	15828
14	yes	848925
15	yes	755319
16	no	30766
17	no	964234
18	yes	974364
19	no	162226

Average Time Taken : 515790.1579 microseconds
No. of False result : 4

6.5 Person 3 :

Table 3 Success/Failure Result Table for Data Set III

Sr No	False Result	Time taken
1	no	416641
2	yes	435886
3	no	918948
4	no	929319
5	no	825289
6	no	740363
7	yes	6584
8	no	111084
9	no	124991
10	no	73078
11	no	937559
12	no	589756
13	no	143775
14	no	937956
15	no	118214
16	no	32077
17	no	67427

18	no	104337

Average Time Taken 417404.6667 microseconds
No. of False result : 2

6.6 Final Result :

Table 4 Detection Accuracy

	Person 1	Person 2	Person 3	Total
False Positive	3	2	3	8
Flase Negativ	2	3	2	8
True Positive	15	15	16	46
Detection Rate	83.333	78.747	88.8889	83.63634
False Detection Rate	16.667	10.526	16.667	14.5454

Average Time Taken : 0.45 seconds

VII. Conclusion

A unique combination of Haar cascade and HOG neural descriptor was implemented and tested, the uniqueness of the research is minimal code complexity, high accuracy rate and small recognition time of 0.45 seconds, further it can be said with better configuration the detection can be increased as well..

References

[1] Viola and Jones, "Rapid object detection using a boosted cascade of simple features", Computer Vision and Pattern Recognition, 2001

[2] M. Yang and D. J. Krieg man, "Detecting faces in images: a survey" IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 24, no. 1, January 2002.

[3] T.K. Leung, M.C. Burl, and P. Perona, “Finding faces in cluttered scenes using random labeled graph matching,” Proc. Fifth IEEE Int. Conf. Computer Vision, pp. 637-644, 1995.

[4] K.C. Yow and R. Cipolla, “A probabilistic framework for perceptual grouping of features for human face detection,” Proc. Second Int. Conf. Automatic Face and Gesture Recognition, pp.1621, 1996.

[5] C. Han, M. Liao, K. Yu, and L. Chen, “Fast face detection via morphology-based pre-processing,” Proc. Ninth Int. Conf. Image Analysis and Processing, pp. 469-476, 1998.

[6] G.E. Hinton, P. Dayan, and M. Revow, “Modeling the manifolds of images of handwritten digits,” IEEE Trans. Neural Networks, vol. 8, no. 1, pp. 65-74, 1997.

[7] H. Rowley, S. Baluja, and T. Kanade, “Neural network-based face detection,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 20, no. 1, pp. 23-38, Jan. 1998.

[8] H. Rowley, S. Baluja, and T. Kanade, “Neural network-based face detection,” Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp.203-208,1996.

[9] E. Osuna, R. Freund, and F. Girosi, “Training support vector machines: an application to face detection,” Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 130-136, 1997

[10] P. Viola and M. J. Jones, "Rapid object detection using a boosted cascade of simple features", IEEE Conference on Computer Vision and Pattern Recognition, 2001.

[11] P. Viola and M. J. Jones, "Robust real-time face detection," Int. J. Computer Vision, vol. 57, no. 2, pp.137-154, 2004.

[12] R. Lienhart and J. Maydt, "An extended set of haar-like features for rapid object detection," IEEE ICIP, vol. 1, pp. 900- 903, Sep. 2002.

[13] S.C. Brubaker, J. Wu, J. Sun, M. Mullin and J. Rehg, "On the design of cascades of boosted ensembles for face detection", Int. J. Computer Vision. vol. 771, no. 3, pp. 65–86. 2008.

[14] C. Zhang and P. Viola, “Multiple-Instance Pruning for Learning Efficient Cascade Detector,” In Proc. Of Neural Information processing Systems , Dec. 2007.

[15] W. Gao, H. Ai, and S. Lao. "Adaptive contour features in oriented granular space for human detection and segmentation". Proc IEEE Computer Vision and Pattern Recognition, 2009.

[16] S. Ma, T. Du, "Improved AdaBoost face detection", International Conference on Measuring Technology and Mechatronics Automation, March 2010.

[17] Y. Hou, A. Q. Peng, "Face detection based on AdaBoost and skin color," Second Int. Symp. Information Science and Engineering, Changsha, 2009.

[18] Y.Chun, L. H. Tseng, "Pornography Detection Using Support Vector Machine", Sixteenth IPPR Conf. Computer Vision, Graphics and Image Processing , Kinmen, 2003.

[19] Viola and Jones, "Rapid object detection using a boosted cascade of simple features", Computer Vision and Pattern Recognition, 2001

[20] "Open source computer vision(OpenCV)", [online document], 2010, April (revision 2.1), Available HTTP: http://opencv.willowgarage.com/wiki/

[21] L. Tamara, A.C. Berg,E. Jaety and D.A. Forsyth, "Names and faces in the news",Computer Science Division, U.C. Berkeley, 2001.