IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-5, May 2019 E-ISSN: 2347-2693

Face Profiler for Face Detection and Recognition

Vivek Kumar Mishra[1]*, Ravindra Gupta[2], Ashvini Chaturvedi[3]

[1] RKDF Institute of Science and Technology, Bhopal, India

[2] RKDF Institute of Science and Technology, Bhopal, India

[3] RKDF Institute of Science and Technology, Bhopal, India

DOI: https://doi.org/10.26438/ijcse/v7i5.11171120 | Available online at: www.ijcseonline.org

Accepted: 18/May/2019, Published: 31/May/2019

Abstract- Viola Jones introduced an efficient method to detect the face rapidly within an image. There are different calculations and strategies utilized for face location; here Viola Jones algorithm is utilized to for face recognition in a picture. This algorithm is accustomed to recognizing and finding the human face independent of its size, circumstance and environment. The face discovery is a strategy that distinguishes the human face and disregarding whatever else, similar to trees, bodies and structures. This algorithm is utilized to discover a programmed human face structure dataset. In this paper we implement Viola Jones algorithm to detect the face, nose, mouth and eye. This paper for the most part tends to the structure of facial acknowledgment programming which falls into a huge gathering of advancements known as biometrics. It has been a standout amongst the most fascinating and significant research fields.

Keywords- Viola-Jones, face detection, Haar feature, Adaboost, Integral Image.

I. INTRODUCTION

The human face poses even more problems than other objects since the human face is a dynamic object that comes in many forms and colors [1]. Viola– Jones calculation is powerful. It has a high detection rate. The calculation is quick enough to be actualized progressively applications. What's more, in the event that we streamline our actualized code, we can make it run a lot quicker. A face recognition calculation is planned by giving input as certain faces and non faces and a training classifier. A training classifier is something which distinguishes a face from the sources of info. We train a classifier utilizing faces and non faces and once the preparation is done, the information that we have can be utilized to identify faces from a picture. Human Computer Interaction (HCI) could greatly be improved by using emotion, pose, and gesture recognition, all of which require face and facial feature detection and tracking [2]. To place this in more straightforward words, we demonstrate a few pictures of face to an alien who has no past information of what the human face is. We demonstrate approximately hundreds or thousands of human faces and reveal to it that it is a human face. What's more, comparably we again demonstrate approximately hundreds or thousands of non faces (or non human faces) and reveal to it that these are non-faces. When that outsider is prepared to recognize the highlights, we give it an info picture and it is then ready to characterize it as a face or a non face.

II. VIOLA-JONES DETECTION ALGORITHM

The characteristics of Viola–Jones algorithm which make it an effective detection algorithm are:

- Robust – very high detection rate (true-positive rate) & very low false-positive rate always.
- Real time – For practical applications at least 2 frames per second must be processed.
- Face detection only

The algorithm has four stages:

1. Haar Feature Selection
2. Creating an Integral Image
3. Adaboost Training
4. Cascading Classifiers

1. Haar Features

Haar Features are essentially like the convolution kernels, which are utilized to recognize the presence of the highlights and features in a picture. Consider the accompanying picture demonstrating the Haar highlights utilized in Viola-Jones face detection algorithm.

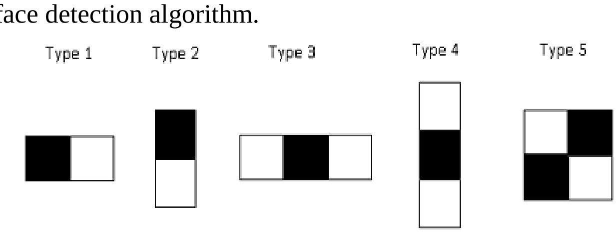

Figure 1. Haar Features

A dark locale is replaced by +1 and the white area is replaced with - 1. When we apply this mask to an input picture, we simply subtract the pixel esteems under white district from the pixel esteems under dark area and the yield will be a single value. So what we comprehend from this is, all these Haar features have a type of similarity to some facial highlights or qualities of countenances. In this way, Haar features speak to certain attributes of a face. Viola-Jones algorithm utilizes a 24x24 sub-window from a picture, and it calculates these features everywhere throughout the picture.

Let us say you have this  feature which is of 2 pixel dimension. This feature is applied at the upper left pixel of the input picture and the value is determined. At that point it is moved ahead appropriate by one unit (pixel) and again the value is determined. And so on this 2 pixel feature is moved over the whole input picture till we end up achieving base corner pixel of the picture. At that point we increment the extent of this feature. We make it 2 pixels white and 2 pixel dark, bringing about a 4 pixel size component. We apply this component to the picture once more by moving it pixel by pixel and we get the qualities. Further, we make it 4 pixels white and 4 pixel dark, and again apply it to the picture to get values. So also same thing is finished by taking the various features and is applied to the picture to get values. Thinking about all the variety of size, position of every one of these highlights, we end up figuring about 160,000+ features in this 24x24 window, as each single kind of feature is rehashed everywhere throughout the picture in all scales, sizes and positions; while everything joined, we have a numerous combinations. Presently for such a case, we have to assess a large set of features for each 24x24 sub-window in any new picture. For ongoing face detection this doesn't look viable. Along these lines, we take out the repetitive features (the highlights which are not valuable) and select just those features which are exceptionally helpful for face recognition. This is finished by Adaboost.

2. Adaboost

Adaboost disposes of all the excess features and it limits it down to a few a large numbers of features that are extremely helpful. Presently before going towards Adaboost, we should discuss a significant idea called Integral Image. As we probably am aware, in the algorithm, each and every time we have to total up every one of the pixels in black area and total up every one of the pixels in the white area. At whatever point we need to register the total of any locale state dark area, this is computationally wasteful if figuring for constant since it will be a long procedure for a huge number of features. So Viola-Jones has thought of a thought which tackles this issue. The fundamental Idea behind integral picture is, we do not need to sum up all the pixels, rather, and we use the corner values of this patch and do a simple mathematical calculation which is stated below

Figure 2. Applying Haar Feature

In an integral image the value at pixel (x,y) is the sum of pixels above and to the left of (x,y). Let's say this is a given input image and we want to calculate the integral image.

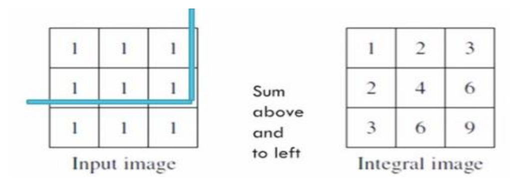

Figure 3. Integral Image Calculation

To get a new value at a particular pixel we sum up all the pixel values falling in top and left region. The resultant is an Integral Image. The integral image allows for the calculation of sum of all pixels inside any given rectangle using only four corner values of the rectangle.
Consider this example.

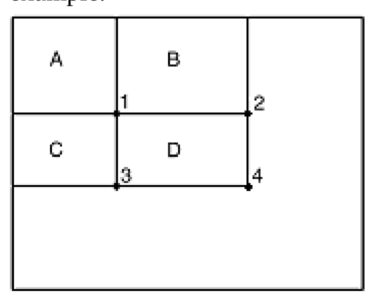

Figure 4. Sum of pixels in rectangle

The Integral sum inside rectangle D, we can compute as : ii(4) + ii(1) - ii(2) - ii(3).

Using the integral image representation one can compute the value of any rectangular sum in constant time.

3. Cascading

For every 24x24 window, we need to evaluate 2500 features that we obtain after performing Adaboost and take a linear combination of all those 2500 outputs to detect whether it exceeds a certain threshold or not, and then decide whether a face is detected or not. Now even though an image is detected in an input image, an excessive large amount of evaluated sub-windows would still be negative.

In simpler terms, instead of using these 2500 features all the time on every single 24x24 window, we use cascades. Out of 2500 features, we make sets of features. Let's say first 10 features are kept in one classifier and then next 10-20 features in another classifier and so on. Though the complexity is increased, but the advantage is that, when we apply this cascade on a certain window of 24x24 size on an given image, based on the output of the first classifier, we can check if it is a face or not. These 2500 features are arranged in a cascading manner as shown below.

The processed solid classifier is additionally called a detector. The essential key of Viola-Jones Face detection algorithm is to filter the detector ordinarily through a similar picture each time with another size.

Let's state we have an input picture of 640X480 pixels resolution, we have to move this 24x24 window all through the picture.

For each 24x24 window, we have to assess 2500 features that we get in the wake of performing Adaboost and take a linear combination of all 2500 yields to distinguish whether it surpasses a specific threshold or not, and afterward choose whether a face is recognized or not. Presently despite the fact that a picture is distinguished in an input picture, an unreasonable expansive measure of assessed sub-windows would even now be negative.

In more straightforward terms, rather than utilizing these 2500 features all the time on each and every 24x24 window, we use falls. Out of 2500 features, we make sets of features. Let's state initial 10 features are kept in one classifier, at that point next 10-20 includes in another classifier, etc. In spite of the fact that the multifaceted nature is expanded, however the preferred standpoint is that, when we apply this course on a specific window of 24x24 size on a given picture, in view of the yield of the main classifier, we can check on the off chance that it is a face or not. These 2500 highlights are masterminded in a falling way as appeared as follows.

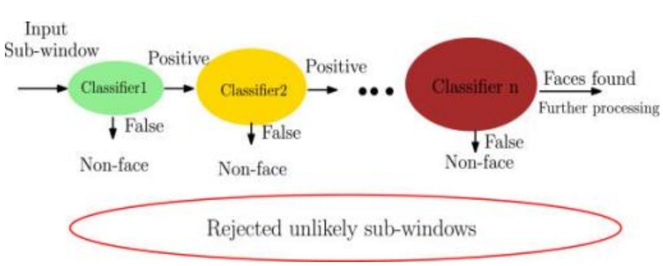

Figure 5. Cascading

With this an input can be rejected quickly from first or first few stages instead of evaluating all the 2500 features. In real time when we need to detect faces, this gives us a lot of computational advantage.

III. RESEARCH METHODOLOGY

The main objectives are as follows:

- Face Detection step is to give the coordinates of x, y, w, h which makes a rectangle box in the picture to show the location of the face or we can say that to show the region of interest in the image.
- Mouth Detection step is to give the coordinates of x, y, w, h which makes a rectangle box in the picture to show the location of the mouth or we can say that to show the region of interest in the image.
- Nose Detection step is to give the coordinates of x, y, w, h which makes a rectangle box in the picture to show the location of the nose or we can say that to show the region of interest in the image.
- Eye Detection step is to give the coordinates of x, y, w, h which makes a rectangle box in the picture to show the location of the eye or we can say that to show the region of interest in the image.

IV. CONCLUSION

Our work on Face recognition has many applications in the real world, few of them are:

- Easy people tagging
- Gaming
- Price comparison
- Making mental notes
- Identifying TV shows
- Augmented reality
- Image search
- Solving Sudoku puzzles
- Security
- Revealing more about you in public

REFERENCES

[1]. Muller, N., Magaia, L. and Herbst B.M. Singular value decomposition, eigenfaces, and 3D reconstructions. SIAM Review, Vol. 46 Issue 3, pp. 518–545. Dec. 2004.
[2]. Bradski, G. Computer vision face tracking for use in a perceptual user interface. Intel Technology Journal, 2nd Quarter, 1998.
[3]. Viola, Jones: Robust Real-time Object Detection, IJCV 2001 See pages 1, 3.
[4]. "Viola-Jones' face detection claims 180k features" Retrieved 2017-06-27.
[5]. R.Szeliski, Computer Vision, algorithms and applications, Springer
[6]. Viola, Jones: Robust Real-time Object Detection, IJCV 2001 See page 11.
[7]. Face Detection and Tracking using the KLT algorithm
[8]. Tej Pal Singh, "Face Recognition by using Feed Forward Back Propagation Neural Network", International Journal of Innovative Research in Technology & Science, vol.1, no.1,
[9]. N.Revathy, T.Guhan, "Face recognition system using back propagation artificial neural networks", International Journal of Advanced Engineering Technology, vol.3, no. 1, 2012.

[10]. H.Schneiderman and T. Kanade,"A statistical method for 3D object detection applied to faces and cars," IEEE conference on Computer vision and pattern recognition, 13-15 June 2000, Hilton Head Island, pp. 746–751.
[11]. Ling-Zhi Liao, Si-Wei Luo, and Mei Tian "Whitened faces Recognition With PCA and ICA" IEEE Signal Processing Letters, vol. 14, no. 12, pp1008-1011, Dec. 2007.
[12]. Paul Viola and Michael J. Jones. Rapid Object Detection using a Boosted Cascade of Simple Features. IEEE CVPR, 2001.
[13]. Kumaran U, Automated Real-Time Face Recognition and Tagging, IJSRCSEIT Volume 4 Issue 6 ISSN : 2456-3307,2018
[14]. Numitha M N1, Taha Noorain1, Face Recognition and IoT Based Smart Lock Access System, IJSRCSEIT | Volume 4 | Issue 6 | ISSN : 2456-3307,2018

