

## Face Identification and Recognition Using AI

Dr. A. Senthil Kumar<sup>1</sup>, R. Anusha<sup>2</sup>, K. Arshitha<sup>2</sup>, S. M. Zeelan Basha<sup>2</sup>, P. Lavanya<sup>2</sup>

<sup>1</sup>B.E., M.E., PGDVLISI., DISM., Ph.D. (IITR), PDF. (TUT.SA), Senior PDF. (VSD. TUV, EUROPE)  
 Department of EEE, Principal, Sanskrithi school of engineering, Puttparthi, Andhra Pradesh, India

<sup>2</sup>Department of ECE, Sanskrithi School of Engineering, Puttparthi, , Andhra Pradesh, India

### ABSTRACT

Nowadays we are using cab system very much because it can fulfil our needs quickly. The most priorable problems are accidents by drowsiness of the drivers, environmental conditions, vehicle conditions etc. Improper cab driver behaviour leads to increase the probability of accidents occurring. Detecting the driver's behaviour is one of interested research area. An idea of this project is proposed by AI, it is an three level process i.e., face detection and face recognition using viola jones algorithm.

Keywords : Viola jones algorithm, eigenface method, Face Detection, Haar-features, Ada-boost, cascading classifiers, SVM and hog, Tensor flow module, Face recognition

### Article Info

Volume 8, Issue 4

Page Number : 291-296

### Publication Issue

July-August-2021

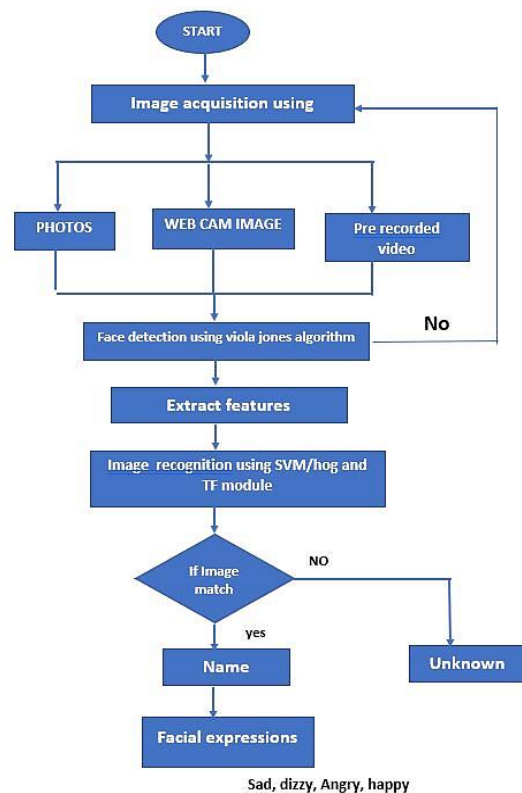
### Article History

Accepted : 10 July 2021

Published : 15 July 2021

## I. INTRODUCTION

FACIAL analytics is the most important research topic nowadays. It is used to extract useful information about pose, key-points, gender, expression, age, identity, etc. It is adopted in many application areas such as law enforcement, face biometrics for payments, self-driving vehicles, active authentication on devices, etc. it is challenging to identify the human faces due to varying light effects and aging factor of human beings. The classification techniques are embracery used over large data sets of human faces to learn rich and compact representations of faces and train the model accordingly to perform face recognition of human beings. Face recognition is a process that involves the following four steps such as



- Face detection: In this process, one or more faces are discovered in an image and mark with a bounding box.
- Feature extraction: The facial features are extracted for the next face recognition task
- Face Alignment: By considering geometry and photometric of the face and the detected face is normalized.
- Face Recognition: Applying of a classifier to recognize the face by taking the existing database for matching purposes.

## II. OPENCV

OpenCV or the Open-Source Computer Vision Library is like an eye for a machine. OpenCV is nothing but the vision to a computer. This library is used for machine-learning modules and artificial-intelligence. OpenCV's typical framework was built to give computer vision applications and to quicken machine's recognition capabilities in the business products. OpenCV is very easy to use framework

## III. VIOLA-JONES FACE DETECTION

The method proposed by Paul Viola and Michael Jones was a significant step forward in the face detection field. Similar to other previous methods, they used machine-learning algorithms to select a set of simple features which they combined into an efficient scalable classifier. Their paper, however, introduced three key innovations that enabled their detector to achieve performance boosts over previous systems. The first was the use of the "integral image" for faster feature computation. The second was the use of the AdaBoost (adaptive boosting) machine-learning algorithm as a means for quickly selecting simple and efficient classifiers. The third was a method for combining classifiers into a "cascade" to quickly eliminate background regions and focus computational attention on more promising areas of the image.

### 3.1. Haar-Like Features:

The Viola-Jones face detection method uses combinations of simple Haar-Like features to classify faces. Haar-like features are rectangular digital image features that get their name from their similarity to Haar-wavelets. Working with individual pixel intensities is computationally expensive, so these features, introduced by Papageorgiouetal. provide a method for encoding image properties in a form that can be computed much more quickly. Simple Haar-like features are composed of two adjacent rectangles, located at any scale and position within an image, and are referred to as "2-rectangle" features. The feature is defined as the difference between the sums of image intensities within each rectangle. Viola and Jones extended this set by defining similar features composed of 3 and 4 rectangles (see Figure 1). Additionally, Lienhart and Maydt defined tilted versions of these features to enrich the possible feature set in an attempt to form better classifiers. These simple features are quite coarse when compared to alternatives such as steerable filters, however, their computational efficiency more than makes up for their limitations

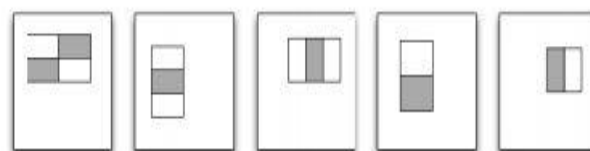


Figure 1. Various types of Haar like features

### 3.2. Integral Image:

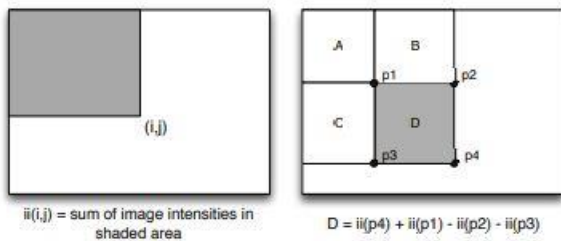
Haar-like features can be calculated extremely quickly by using an image representation called the integral image. The integral image is an application of summed-area tables; a concept introduced by Crow in 1984, usually used in computer graphics. The integral image can be defined as:

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \quad (1)$$

Where  $ii(x, y)$  is the integral image and  $i(x, y)$  is the original image intensity. The integral image can be calculated in a single pass using the following recurrences:

$$\begin{aligned} s(x, y) &= s(x, y - 1) + i(x, y) \\ ii(x, y) &= ii(x - 1, y) + s(x, y). \end{aligned} \tag{2}$$

Here's  $(x, y)$  is the cumulative row sum and we have the following base cases:  $s(x, -1) = 0$  and  $ii(-1, y) = 0$ . Using the integral image, each feature can be calculated in constant time. The sum of any rectangle can be calculated in only four memory references (see Figure 2).



**Figure 2.** Integral image and rectangle feature selection

Furthermore, because Haar-like features use rectangles that share corners, the number of total memory references needed to compute a feature can be further reduced. For example, 2, 3 and 4-rectangle features require only 6, 8 and 9 memory references respectively.

**3. 3 Feature Selection:**

The set of possible features in a given sub-window is huge. For example, there are over 160,000 features in any given  $24 \times 24$ -pixel window using only 2, 3 and 4-rectangle features. This set is much larger than the number of pixels in the window and is many times overcomplete, however, Viola and Jones hypothesized (and discovered through experimentation) that a very small number of critical features can be combined to form an effective classifier. Using a large set of hand-labelled training data and a possible feature set, many different types of machine-learning techniques can be

used to train a classification function. The Viola Jones method uses a variation of the AdaBoost algorithm, formulated by Freund and Schapire in, to select a small set of critical features to form an effective classifier. For best results, the training data should contain images over a range of lighting conditions and facial properties (e.g., skin colour, glasses, facial hair).

**3.4. Attentional Cascade:**

Perhaps the most important innovation of the Viola Jones method is the advent of the attentional cascade. The idea is to focus first on the removal of negative regions of the image while including almost all of the positive ones. A simple version of this idea was implemented by Rowley. Their method was to use two neural networks, one complex network that focused on small regions of the image and detected faces with a low false positive rate, and a simpler faster network that focused on larger regions but had higher false positive rates. The algorithm first uses the faster neural network to select regions of interest before running the slower, more complex network to pick out the faces. The Viola-Jones method extends this approach to a 38-stage cascade of classifiers in order to minimize total computation. The initial stages are composed of fewer, simpler features and are created by adjusting AdaBoost to favour false-positives and maximize detection rates. The later stages of the cascade use more complex classifiers to further reduce overall false-positive rates. Sub-windows of the image are only accepted as faces if they have passed every stage of the cascade. Running the cascade in this specific order allows many regions to be discarded early in the cascade so they will not require any further processing. After the training phase is complete, we have a cascade of individual classifiers referred to as levels. Each level is made up of a threshold value and a set of features with corresponding success and failure at the threshold value. To run the cascade, we simply iterate through each level, calculating feature sums. If the feature sum is within the threshold, the success value is added to a

level sum. Otherwise, the failure value is added. After the level sum calculation is complete, if it is within the level threshold, the image passes the level. A face is detected if the image successfully passes every level of the cascade.

**3.5. Detection:**

After the training process is complete, the detection algorithm is fairly straightforward. One simply has to scan all possible sub-windows of an image at a range of scales, running the cascade on each window. If a sub-window passes the final level of the cascade, then the sub-window likely contains a face. It is important to note that the detector is invariant to small changes in scale and location, so there will often be many hits centred around each face. To filter out these duplicated results, overlapping hits can be averaged together to form a single detection. Clearly, running the cascade on every possible sub-window is computationally expensive. Thus, the amount we adjust scale and position between tested sub-windows (scanning factor) must be adjusted to achieve a balance between speed and accuracy. Viola and Jones achieved best results using a scaling factor of 1.25, and a scanning factor proportional to the current scale.

**IV. K NEAREST NEIGHBOR CLASSIFIER (KNN)**

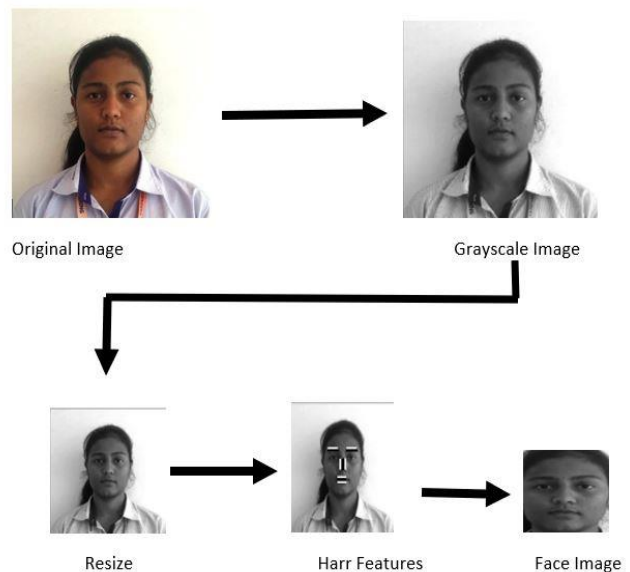
The K nearest neighbour (KNN) classifier is an extension of the simple nearest neighbour (NN) classifier system. The nearest neighbour classifier works based on a simple nonparametric decision. Each query image Iq is examined based on the distance of its features from the features of other images in the training database. The nearest neighbour is the image which has the minimum distance from the query image in the feature space. The distance between two features can be measured based on one of the distance functions such as, city block distance d1, Euclidean distance d2 or cosine distance dcos:

$$d1(x, y) = \sum_{i=1}^N |xi - yi|$$

$$d2(x, y) = \sqrt{\sum_{i=1}^N |xi - yi|^2}$$

$$d \cos(x, y) = 1 - \frac{\vec{x} \cdot \vec{y}}{|x| \cdot |y|}$$

K nearest neighbour algorithm uses K closest samples to the query image. Each of these samples belongs to a known class Ci. The query image Iq is categorized to the class CM which has the majority of occurrences among the K samples. The performance of the kNN classifiers highly related to value of the k, the number of the samples and their topological distribution over the feature space. Appendix A describes this through a series of experiments. Many approaches are introduced to improve the performance of the kNN systems using wavelet techniques, Cluster-Based Trees and Tolerant rough sets and so on. In this paper we show ensemble-based techniques can be used to improve the performance of the system.



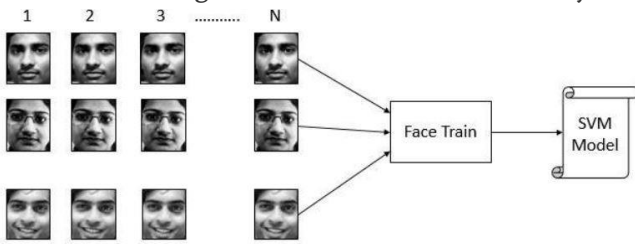
An image in the test set is recognized by assigning its label of the closest point in the learning set, where Euclidean distances between two pixels are measured in image space. The Euclidean distance is used to determine the closeness between the data points in K-NN. It is assigned between all pixels in a dataset.

**V. HOG+Linear SVM**

$$h = \sqrt{hx^2 + hy^2}$$

$$\theta = \arctan \frac{hy}{hx}$$

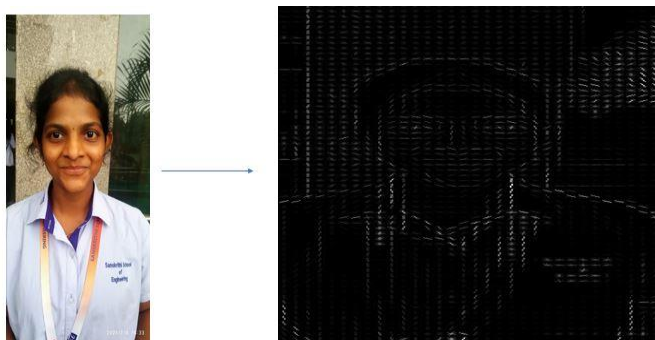
Support vector machines (SVMs) are supervised machine learning models that divide and classify data.



SVMs are widely used for applications such as face detection, classification of images, handwriting recognition, etc. An SVM model can be considered as a point space wherein multiple classes are isolated using hyperplanes.

**Histogram of Oriented Gradients (HOG):**

A HOG is a feature descriptor generally used for object detection. HOGs are widely known for their use in pedestrian detection. A HOG relies on the property of objects within an image to possess the distribution of intensity gradients or edge directions. Gradients are calculated within an image per block. A block is considered as a pixel grid in which gradients are constituted from the magnitude and direction of change in the intensities of the pixel within the block.



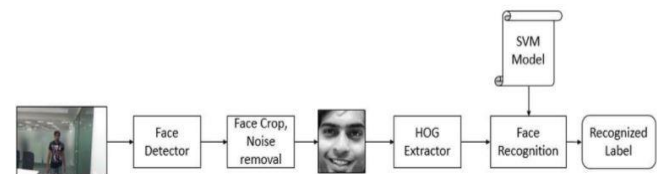
**Hog feature sample face**

In the current example, all the face sample images of a person are fed to the feature descriptor extraction algorithm; i.e., a HOG. The descriptors are gradient vectors generated per pixel of the image. The gradient for each pixel consists of magnitude and direction, calculated using the following formulae:

In the current example, Gx and Gy are respectively the horizontal and vertical components of the change in the pixel intensity. A window size of 128 x 144 is used for face images since it matches the general aspect ratio of human faces. The descriptors are calculated over blocks of pixels with 8 x 8 dimensions. These descriptor values for each pixel over 8 x 8 block are quantized into s9 bins, where each bin represents a directional angle of gradient and value in that bin, which is the summation of the magnitudes of all pixels with the same angle. Further, the histogram is then normalized over a 16 x 16 block size, which means four blocks of 8 x 8 are normalized together to minimize light conditions. This mechanism mitigates the accuracy drop due to a change in light. The SVM model is trained using a number of HOG vectors for multiple faces.

**Face Recognition**

The recognition of a face in a video sequence is split into three primary tasks: Face Detection, Face Prediction, and Face Tracking. The tasks performed in the Face Capture program are performed during face recognition as well. To recognize the face obtained, a vector of HOG features of the face is extracted. This vector is then used in the SVM model to determine a matching score for the input vector with each of the labels. The SVM returns the label with the maximum score, which represents the confidence to the closest match within the trained face data.



The task of calculating matching scores is exceptionally heavy to compute. Hence, once detected and identified, the labelled face in an image

needs to be tracked to reduce the computation in future frames until the face eventually disappears from the video. Of all the available trackers, the Camshift tracking algorithm is used since it produces the best results with faces.

## VI. SCOPE FOR FUTURE ENHANCEMENT

Can be implemented in fully-automated electronic vehicles for driver assistance system.

- The proposed idea can be enhanced by indicating the present situation of the cab driver to the responsible person.
- If there is any complication in the emotional status of the driver then immediately the information should be given to the nearby vehicle with the help of vehicle to vehicle communication.
- This system is developed which can act according to the emotions of that person and supports him through the speech delivering which can change his mood.

## VII. REFERENCES

- [1]. Proceedings of the Fifth International Conference on Communication and Electronics Systems (ICCES 2020) IEEE Conference Record # 48766; IEEE Xplore ISBN: 978-1-7281-5371-1
- [2]. Proceedings of the International Conference on Intelligent Computing and Control Systems (ICICCS 2020) IEEE Xplore Part Number:CFP20K74-ART; ISBN: 978-1-7281-4876-2
- [3]. National Center for Statistics and Analysis. Distracted Driving 2014, Traffic Safety Facts; National Highway Traffic Safety Administration's National Center for Statistics and Analysis, U.S. Department of Transportation: Washington, DC, USA, 2016.
- [4]. E. Ericsson, "Variability in urban driving patterns," *Transportation Res. Part D*, vol. 5, pp. 337-354, 2000
- [5]. Proceedings of the International Conference on Intelligent Computing and Control Systems (ICICCS 2020) IEEE Xplore Part Number:CFP20K74-ART; ISBN: 978-1-7281-4876-2
- [6]. 2009 Canadian Conference on Computer and Robot Vision Real-Time Viola-Jones Face Detection in a Web Browser Theo Ephraim Tristan Himmelman Kaleem Siddiqi School of Computer Science & Centre for Intelligent Machines McGill University.

### Cite this article as :

Dr. A. Senthil Kumar, R. Anusha, K. Arshitha, S. M. Zeelan Basha, P. Lavanya, "Face Identification and Recognition Using AI", *International Journal of Scientific Research in Science and Technology (IJSRST)*, Online ISSN : 2395-602X, Print ISSN : 2395-6011, Volume 8 Issue 4, pp. 291-296, July-August 2021. Available at doi : <https://doi.org/10.32628/IJSRST218447>  
Journal URL : <https://ijsrst.com/IJSRST218447>