**Abstract**

We address the problem of robust face recognition with under sampled training data. Given only one or few training images available per subject, we present a novel recognition approach, which not only handles test images with large intraclass variations such as illumination and expression. The proposed method is also to handle the corrupted ones due to occlusion or disguise, which is not present during training. This is achieved by the learning of a robust auxiliary dictionary from the subjects not of interest. Together with the under sampled training data, both intra and interclass variations can thus be successfully handled, while the unseen occlusions can be automatically disregarded for improved recognition. Our experiments on four face image datasets confirm the effectiveness and robustness of our approach, which is shown to outperform state-of-the-art sparse representation-based methods.

**Chapter 1**

**Preamble**

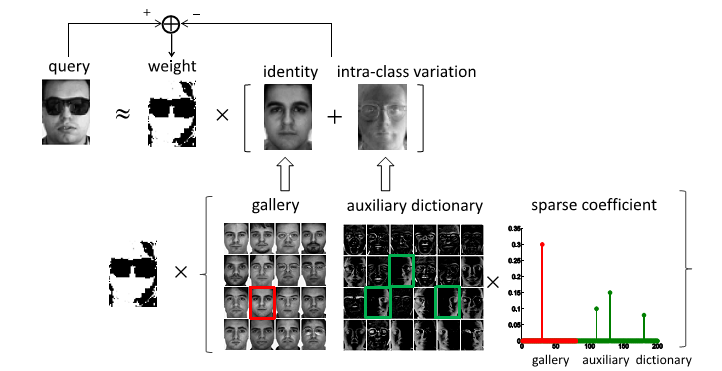
**1.1 INTRODUCTION**

A **face recognition system** is a [computer application](https://en.wikipedia.org/wiki/Application_software) capable of [identifying](https://en.wikipedia.org/wiki/Identification_of_human_individuals) or [verifying](https://en.wikipedia.org/wiki/Authentication) a person from a [digital image](https://en.wikipedia.org/wiki/Digital_image) or a [video frame](https://en.wikipedia.org/wiki/Film_frame) from a [video](https://en.wikipedia.org/wiki/Video) source. One of the ways to do this is by comparing selected [facial features](https://en.wikipedia.org/wiki/Face) from the image and a face [database](https://en.wikipedia.org/wiki/Database_management_system).

It is typically used in [security systems](https://en.wikipedia.org/wiki/Burglar_alarm) and can be compared to other [biometrics](https://en.wikipedia.org/wiki/Biometrics) such as [fingerprint](https://en.wikipedia.org/wiki/Fingerprint) or eye [iris recognition](https://en.wikipedia.org/wiki/Iris_recognition) systems Recently, it has also become popular as a commercial identification and marketing tool

FACE recognition has been an active research topic, since it is challenging to recognize face images with illumination and expression variations as well as corruptions due to occlusion or disguise. A typical solution is to collect a sufficient amount of training data in advance, so that the above intraclass variations can be properly handled. However, in practice, there is no guarantee that such data collection is applicable, nor the collected data would exhibit satisfactory generalization.

Moreover, for real-world applications, e.g : e-passport,driving license, or ID card identification, only one or very few face images of the subject of interest might be captured during the data acquisition stage. As a result, one would encounter the challenging task of undersampled face recognition. Existing solutions to undersampled face recognition can be typically divided into two categories: Patch-based methods and Generic learning from external data. For patch-based methods, one can either extract discriminative information from patches collected by different images, or utilize/integrate the corresponding classification results for achieving recognition[1].



**Figure 1.1: Sparse dictionary learning**

**Sparse dictionary learning** is a [representation learning](https://en.wikipedia.org/wiki/Representation_learning) method which aims at finding a [sparse](https://en.wikipedia.org/wiki/Sparse_matrix) representation of the input data (also known as sparse coding) in the form of a linear combination of basic elements as well as those basic elements themselves. These elements are called atoms and they compose a dictionary. Atoms in the dictionary are not required to be [orthogonal](https://en.wikipedia.org/wiki/Orthogonal_basis), and they may be an over complete spanning set. This problem setup also allows the dimensionality of the signals being represented to be higher than the one of the signals being observed. The above two properties lead to having seemingly redundant atoms that allow multiple representations of the same signal but also provide an improvement in [sparsity](https://en.wikipedia.org/wiki/Sparsity" \o "Sparsity) and flexibility of the representation.

One of the key principles of dictionary learning is that the dictionary has to be inferred from the input data. The emergence of sparse dictionary learning methods was stimulated by the fact that in [signal processing](https://en.wikipedia.org/wiki/Signal_processing) one typically wants to represent the input data using as few components as possible. Before this approach the general practice was to use predefined dictionaries (such as [fourier](https://en.wikipedia.org/wiki/Fourier_transform" \o "Fourier transform) or [wavelet](https://en.wikipedia.org/wiki/Wavelet_transform) transforms). However, in certain cases a dictionary that is trained to fit the input data can significantly improve the sparsity, which has applications in data decomposition, compression and analysis and has been used in the fields of image [denoising](https://en.wikipedia.org/wiki/Noise_reduction" \o "Noise reduction) and [classification](https://en.wikipedia.org/wiki/Image_classification), video and [audio processing](https://en.wikipedia.org/wiki/Audio_signal_processing).

**1.2 PROBLEM DEFINITION**

In particular, assumes that the objective function can be approximated by a first order Taylor expansion with a quadratic residual term. As a result, what RSC minimizes is an approximated version of the original objective function. On the other hand, our approach directly solves the optimization problem by the technique of variable substitution and the chain rule for calculating the derivatives. We note that the derivations of RSC and ours lead to similar algorithms that both iteratively solve a weighted sparse coding problem and update the weight matrix accordingly. However, our derivation guarantees the optimal solution, while the derivation of RSC might result in an approximated one. We note that RSC is extended from SRC, which requires a sufficient amount of training data (i.e., an over-complete dictionary) and thus is not able to handle under sampled recognition problem.

**1.3 General Aspects and Technology**

The name MATLAB stands for **Matrix Laboratory**. MATLAB was written originally to provide easy access to matrix software developed by the LINPACK (linear system package) and EISPACK (Eigen system package) projects.

MATLAB is a high-performance language for technical computing. It integrates computation, visualization, and programming environment. Furthermore, MATLAB is a modern programming language environment: it has sophisticated data structures, contains built-in editing and debugging tools, and supports object-oriented programming. These factors make MATLAB an excellent tool for teaching and research[2].

MATLAB has many advantages compared to conventional computer languages (e.g., C, FORTRAN) for solving technical problems. MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. The software package has been commercially available since 1984 and is now considered as a standard tool at most universities and industries worldwide.

It has powerful built-in routines that enable a very wide variety of computations. It also has easy to use graphics commands that make the visualization of results immediately available. Specific applications are collected in packages referred to as toolbox. There are toolboxes for signal processing, symbolic computation, control theory, simulation, optimization, and several other fields of applied science and engineering.

There are various tools in MATLAB that can be utilized for image processing, such as Simulink, GUI etc. Simulink contains various toolboxes and image processing toolbox is one such example. Simulink is used for simulation of various projects. GUI is another important tool in MATLAB. It can be designed either by manual programming which is tedious task or by using guide.

**1.4 OBJECTIVE**

The project address the problem of robust face recognition with undersampled training data. Given only one or few training images available per subject, we present a novel recognition approach, which not only handles test images with large intraclass variations such as illumination and expression. Our experiments on four face image datasets confirm the effectiveness and robustness of our approach, which is shown to outperform state of the art sparse representation based methods.

Design of advocate the extraction of representative information from external data via dictionary learning without assuming the prior knowledge of occlusion in query images. This framework is considered as robust auxiliary dictionary learning (RADL).

**CHAPTER 2**

**Literature SURVEY**

Literature survey is the most important step in the software development process .Before developing any tool it is very necessary to determine the time factor, economy and company strength. Once these things are satisfied, then next step are to determine which operating system and language can be used for developing the tool. Once the programmers start building the tool the programmers need lot of external support. This support can be obtained from senior programmers from book or from websites. Before building the system the above consideration are taken into account for developing the proposed system.

**2.1 RELATED WORK**

[3] The author presents **Deep face attributes recognition using spatial transformer network** that allows the face alignment is very crucial to the task of face attributes recognition. The performance of face attributes recognition would notably degrade if the fiducially points of the original face images are not precisely detected due to large lighting, pose and occlusion variations. In order to alleviate this problem, we propose a spatial transform based deep CNNs to improve the performance of face attributes recognition. In approach, first learn appropriate transformation parameters by a carefully designed spatial transformer network called LoNet to align the original face images, and then recognize the face attributes based on the aligned face images using a deep network called ClNet. To the best of our knowledge, this is the first attempt to use spatial transformer network in face attributes recognition task. Extensive experiments on two large and challenging databases (CelebA and LFWA) clearly demonstrate the effectiveness of the proposed approach over the current state-of-the-art.

[2] The author proposed **Robust dual-stage face recognition method using PCA and high-dimensional-LBP** that allows dual-stage face recognition method which utilized both holistic and local features-based recognition algorithms. In the first stage Principal Components Analysis (PCA) is utilized to recognize test image. If the confidence level test is passed, the recognition process will be terminated. Otherwise, the second stage where High Dimensional Local Binary Patterns (HDLBP) is employed will be pursued. The performance of this hybrid method is evaluated on CMU-PIE database, and we obtain improved recognition rate than PCA alone.

[3] The author proposed **Real time recognition of human faces** that allows a simple yet efficient face detection and recognition system is proposed in this paper which has the capability to recognize human faces in single as well as multiple face images in a database in real time. Preprocessing of the proposed frame work includes noise removal and hole filling in color images. After preprocessing, face detection is performed by using viola jones algorithm. Detected faces are cropped out of the input image to make computation fast. SURF features are extracted from the cropped image. For face matching, putative feature matching is carried out and outliers are removed using M-estimator Sample Consensus (MSAC) algorithm. Single as well as multiple person color images from class persons of Graz 01 dataset are used to evaluate the system.

[4] The author presents **Face recognition: Perspectives from the real world Control, Automation** that allows some of our real-world deployment of face recognition (FR) systems for various applications and discuss the gaps between expectations of the user and what the system can deliver. evaluate some of the existing algorithms with modifications for applications including FR on wearable devices (like Google Glass) for improving social interactions, monitoring of elderly people in senior citizens centers, FR of children in child care centers and face matching between a scanned IC/passport face image and few live webcam images for automatic hotel/resort checkout or clearance. Each of these applications poses unique challenges and demands specific research components so as to adapt in the actual sites.

**[5]** The author proposed **Face detection and recognition in videos** that allows advancement in computer technology has made possible to evoke new video processing applications in field of biometric face detection and recognition. It has wide range of applications in human recognition, human computer interaction (HCI), behavior analysis, teleconferencing and video surveillance. Face is vital part of human anatomy that reflects prominent topographies of a person. Face detection has become popular biometric trait in recent years due to its importance in security control applications. The first step in practical face analysis systems is real-time detection of face in frames containing face and complex objects in background. In this paper a system is proposed for human face detection using Haar features and recognition using Eigen and Gabor filter in videos. Efforts are made to minimize processing time for detection and recognition process. The Eigenface method performs well in terms of computational complexity whereas Gabor filter are robust to pose changes.

[6] The author explains about **Face recognition based on convolutional neural network and support vector machine Information and Automation (ICIA)** that allows face recognition is an important embodiment of human-computer interaction, which has been widely used in access control system, monitoring system and identity verification. However, since face images vary with expressions, ages, as well as poses of people and illumination conditions, the face images of the same sample might be different, which makes face recognition difficult. There are two main requirements in face recognition, the high recognition rate and less training time. In this paper, combining Convolutional Neural Network (CNN) and Support Vector Machine (SVM) to recognize face images. CNN is used as a feature extractor to acquire remarkable features automatically. First pre-train our CNN by ancillary data to get the updated weights, and then train the CNN by the target dataset to extract more hidden facial features. Finally we use SVM as our classifier instead of CNN to recognize all the classes. With the input of facial features extracted from CNN, SVM will recognize face images more accurately. In our experiments, some face images in the Casia-Webfaces database are used for pre-training, and FERET database is used for training and testing. The results in experiments demonstrate the efficiency with high recognition rate and less training time.

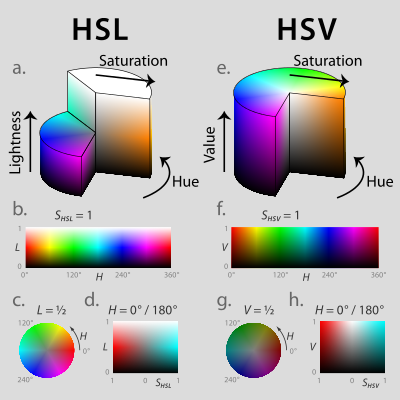
[7] The author proposed **“Adaptive discriminant learning for face recognition,” Pattern Recognition** that allows face recognition from Single Sample per Person (SSPP) is extremely challenging because only one sample is available for each person. While many discriminant analysis methods, such as Fisher faces and its numerous variants, have achieved great success in face recognition, these methods cannot work in this scenario, because more than one sample per person are needed to calculate the within-class scatter matrix. To address this problem, the Adaptive Discriminant Analysis (ADA) in which the within-class scatter matrix of each enrolled subject is inferred using his/her single sample, by leveraging a generic set with multiple samples per person. this method is motivated from the assumption that subjects who look alike to each other generally share similar within-class variations. In ADA, a limited number of neighbors for each single sample are first determined from the generic set by using kNN regression or Lasso regression. Then, the within-class scatter matrix of this single sample is inferred as the weighted average of the within-class scatter matrices of these neighbors based on the arithmetic mean or Riemannian mean. Finally, the optimal ADA projection directions can be computed analytically by using the inferred within-class scatter matrices and the actual between-class scatter matrix. The proposed method is evaluated on three databases including FERET database, FRGC database and a large real-world passport-like face database. The extensive results demonstrate the effectiveness of our ADA when compared with the existing solutions to the SSPP problem.

[8] The author presents **“Extended SRC: Undersampled face recognition via intraclass variant dictionary** sparse Representation-Based Classification (SRC) is a face recognition breakthrough in recent years which has successfully addressed the recognition problem with sufficient training images of each gallery subject. In this paper, extending SRC to applications where there are very few, or even a single, training images per subject. Assuming that the intraclass variations of one subject can be approximated by a sparse linear combination of those of other subjects, Extended Sparse Representation-Based Classifier (ESRC) applies an auxiliary intraclass variant dictionary to represent the possible variation between the training and testing images. The dictionary atoms typically represent intraclass sample differences computed from either the gallery faces themselves or the generic faces that are outside the gallery. Experimental results on the AR and FERET databases show that ESRC has better generalization ability than SRC for undersampled face recognition under variable expressions, illuminations, disguises, and ages. The superior results of ESRC suggest that if the dictionary is properly constructed, SRC algorithms can generalize well to the large-scale face recognition problem, even with a single training image per class.

[9] The author proposed **“Multi-scale patch based collaborative representation for face recognition with margin distribution optimization** small sample size is one of the most challenging problems in face recognition due to the difficulty of sample collection in many real-world applications. By representing the query sample as a linear combination of training samples from all classes, the so-called collaborative representation based classification (CRC) shows very effective face recognition performance with low computational cost. However, the recognition rate of CRC will drop dramatically when the available training samples per subject are very limited. One intuitive solution to this problem is operating CRC on patches and combining the recognition outputs of all patches. Nonetheless, the setting of patch size is a nontrivial task.

**2.2 Existing System**

Simultaneous face hallucination and face recognition, here they consider the algorithms HSV **(Hue, Saturation, Value)** and HSI separately. The major difference between these two algorithms is in the selection of the LR-HR training pairs when performing face super-resolution for face verification and identification



**Figure 2.1: HSL (a–d) and HSV (e–h).**

Above (a, e): cut-away 3D models of each. Below: two-dimensional plots showing two of a model’s three parameters at once, holding the other constant: cylindrical shells (b, f) of constant saturation, in this case the outside surface of each cylinder; horizontal cross-sections (c, g) of constant HSL lightness or HSV value, in this case the slices halfway down each cylinder; and rectangular vertical cross-sections (d, h) of constant hue, in this case of hues 0° red and its complement 180° cyan.

**A. Simulaneous face hallucination and verification (SHV)**

Given a pair of face images, the task of face verification is to verify whether they belong to the same person or not. In our framework for simultaneous face hallucination and verification, we will first use the singular values of face images as a global scale-invariant feature vector for their representation according to Equation. We define a similarity function, SIM, to measure the similarity between an input query and the claimed identity in the gallery database, based on singular values only, as follows:

**hallucination and identification (SHI)**

For simultaneous face hallucination and identification, Q faces that are the most similar to theinput LR face image are first searched from a gallery database of LR-HR pairs based on its singular values, as shown in Fig

**B. Simultaneous face**

Suppose that these Q faces belong to M distinct subjects, where M < Q, and for each of the M identities, the corresponding mapping models are learned and used to super-resolve the query input. Therefore, M hallucinated HR face images I hj, j  , , ...,M, 1 2 for the LR query are generated using the face hallucination algorithm introduced in Section III.A. Then, the differences between each of the M hallucinated HR face images and the corresponding HR face images of the jth distinct subject, j  , , ...,M, 1 2 in the gallery database are computed using the eigenface method.

**ADVANTAGES**

* Improve the performance for robust face recognition under the scenario of undersampled face recognition
* Modeling of intraclass variation
* Performs face super-resolution for face verification and identification
* Measure the similarity between an input query and the claimed identity in the gallery database.

**DISADVANTAGES**

* Only one or few non occluded training images are available for each subject of interest.
* The learned dictionaries cannot guarantee the recognition performance.
* The method cannot identify the faces which are covered with mask.
* Do not provide the security for hackers.

**2.3 proposed system**

Face recognition is a technology that can be applied and implemented in many parts of today’s society. Some areas of applications are

• Biometrics: Using a face as a biometric is proved to be a successful approach since it is the way humans recognize each other.

• Identification Systems: A face could be used to examine if a person exist or not in a list of individuals. Based on that information the system can allow respectively deny access.

• Law Enforcement: Face recognition technology can be used to increase performance of surveillance and law enforcement.

There have been great achievements and progress in this particular field of study, but there are many challenges left to overcome. Still today, low accuracy is one of the main drawbacks of face recognition. Also considering that this technology can be applied in several important areas is making it an appropriate technology to develop.

The purpose of this project is to create an face recognition algorithm that can recognize faces in manipulated images.

**ADVANTAGES**

* Combines the benefits of existing authentication system to form 3 levels of secure password.
* Provides security against hackers and robot attacks.
* Unauthorized users cannot access the system**.**
* Low power consumption.
* Easy to implement.
* Non intrusive
* Cheap technology.
* Verification time is about five seconds.
* Very high accuracy.
* Is the most economical biometric PC user authentication technique.
* Confidential Data will be saved from thefts.

**CHAPTER 3**

**SYSTEM REQUIREMENT SPECIFICATION**

A software requirement specification (SRS) is a comprehensive description of the intended purpose and environment for software under development. The SRS describes what the software will do and how it will be expected to perform. An SRS minimizes the time and effort required by developers to achieve desired goals and also minimizes the development cost. A good SRS defines the way in which an application interacts with the system hardware. Parameters such as computational speed, response time, availability, portability, maintainability, security and speed of recovery from adverse events are evaluated.

**3.1 SOFTWARE REQUIREMENT:**

Software requirement deals with defining software resource requirements and prerequisites that need to be installed on a computer to provide optimal functioning of an application. These requirements or pre-requisites are generally not included in the software installation package and need to be installed separately before the software is installed. Typical platform include computer architecture, operating system or programming languages and their run time libraries software required is specified below.

**SOFTWARE DESCRIPTION**

* Operating system : Windows XP/7.
* Coding Language : MATLAB
* Tool : MATLAB R 2012

**3.2 HARDWARE REQUIREMENTS:**

Hardware requirements and pc requirements are specified below.

**HARDWARE REQUIREMENTS**

**HARDWARE DESCRIPTION**

* System : Pentium IV 2.4 GHz.
* Hard Disk : 40 GB.
* Floppy Drive : 1.44 Mb.
* Monitor : 15 VGA Colour.
* Mouse : Logitech.
* Ram : 512 Mb.
* Speed : 1.1Ghz

**3.3 Feasibility study**

The feasibility of the project is analyzed in this phase. During this study an estimate is made whether the identified user needs can be satisfied using current software and hardware technologies. During system analysis the feasibility study of the proposed system is to be carried out. For feasibility analysis, some understanding of the major requirements for the system is essential.

The feasibility analysis involves three considerations such as,

1. Technical Feasibility
2. Economical Feasibility
3. Social Feasibility

**Technical Feasibility**

Technical feasibility is carried out to check the technical requirements of the system. The technical resources of the system must not impose high demand which leads to greater burden being placed on the client. The software and hardware requirements of the project are easily satisfied and the user interface is simple, hence it is technically feasible.

**Economical feasibility**

Economical feasibility considers the facts such as whether the benefits will exceed its costs and whether the resources are cost effective. This project was also considered economically feasible as the software and hardware components can easily be developed within the existing budgetary constraints. Hence this achieved because most of the technologies used are freely available.

**Social feasibility**

The aspect of social feasibility is to check the level of acceptance of the system by the user. The user has to be well trained to use the system efficiently. The user must not feel that he is threatened by the system and he must accept as it is necessarily important. The level of acceptance by the users depends on the methods that are employed to educate the user about the system and to make user familiar with it. The level of confidence must be raised so that user is also able to make some constructive criticism, which is welcomed, as the user is the final customer of the system.

**3.4 SYSTEM DESIGN**

System design is a modeling process. It is an approach to create a new system. It can be defined as transition from users view to programmers or database person’s view. The design phase acts as a bridge between the required specification and the implementation phase.

**REFRENCE IMAGE**

**CAPTURED IMAGE**

**RGB TO GRAY CONVERSION**

**RGB TO GRAY CONVERSION**

**IMAGE RESIZING**

**IMAGE RESIZING**

**IMAGE ENHANCEMENT**

**IMAGE ENHANCEMENT**

**EDGE DETECTION**

**EDGE DETECTION**

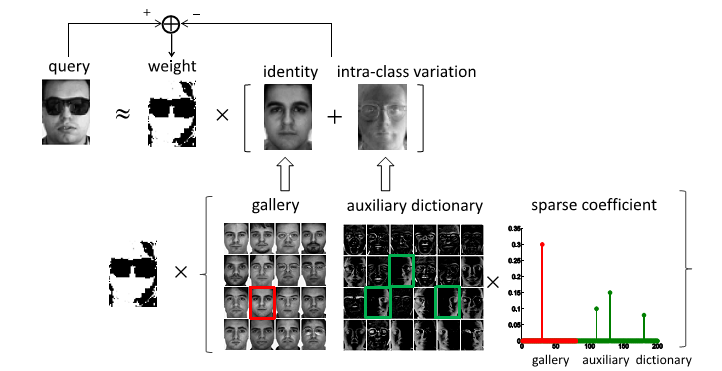
**IMAGE MATCHING**

**TIMING ALLOCATION**

**Figure 3.1 Sequence Diagram**

Above figure 3.1 shows the Sequence Diagram of “Under sampled Face Recognition via Robust Auxiliary Dictionary Learning ” (proposed algorithm)

The Block diagram above gives an overview of Under sampled Face Recognition via Robust Auxiliary Dictionary Learning. Various boxes in Block diagram are explained below:



**Figure 3.2: Robust auxiliary dictionary learning**

**3.4.1 Image Acquisition**

Generally an image is a two-dimensional function f(x,y)(here x and y are plane coordinates).The amplitude of image at any point say f is called intensity of the image. It is also called the gray level of image at that point. We need to convert these x and y values to finite discrete values to form a digital image. The input image is a fundus taken from stare data base and drive data base. The image of the retina is taken for processing and to check the condition of the person. We need to convert the analog image to digital image to process it through digital computer. Each digital image composed of a finite elements and each finite element is called a pixel.

**3.4.2 Formation of Image**

We have some conditions for forming an image f(x, y) as values of image are proportional to energy radiated by a physical source. So f(x, y) must be nonzero and finite. i.e. **0< f(x, y) < ∞.**

**3.4.3 Image Pre-Processing**

**3.4.3.1 Image Resizing/Scaling**

Image scaling occurs in all digital photos at some stage whether this be in Bayer demosaicing or in photo enlargement. It happens anytime you resize your image from one pixel grid to another. Image resizing is necessary when you need to increase or decrease the total number of pixels. Even if the same image resize is performed, the result can vary significantly depending on the algorithm.

Images are resized because of number of reasons but one of them is very important in our project. Every camera has its resolution, so when a system is designed for some camera specifications it will not run correctly for any other camera depending on specification similarities. so it is necessary to make the resolution constant for the application and hence perform image resizing.

**3.4.3.2 RGB to GRAY Conversion**

Humans perceive colour through wavelength-sensitive sensory cells called cones. There are three different varieties of cones, each has a different sensitivity to electromagnetic radiation (light) of different wavelength. One cone is mainly sensitive to green light, one to red light, and one to blue light. By emitting a restricted combination of these three colours (red, green and blue), and hence stimulate the three types of cones at will, we are able to generate almost any detectable colour. This is the reason behind why colour images are often stored as three separate image matrices; one storing the amount of red (R) in each pixel, one the amount of green (G) and one the amount of blue (B). We call such colour images as stored in an RGB format.

In grayscale images, however, we do not differentiate how much we emit of different colours, we emit the same amount in every channel. We will be able to differentiate the total amount of emitted light for each pixel; little light gives dark pixels and much light is perceived as bright pixels. When converting an RGB image to grayscale, we have to consider the RGB values for each pixel and make as output a single value reflecting the brightness of that pixel. One of the approaches is to take the average of the contribution from each channel: (R+B+C)/3. However, since the perceived brightness is often dominated by the green component, a different, more "human-oriented", method is to consider a weighted average, e.g.: 0.3R + 0.59G + 0.11B.

**3.4.4 Image Enhancement**

Image enhancement is the process of adjusting digital images so that the results are more suitable for display or further analysis. For example, we can eliminate noise, which will make it more easier to identify the key characteristics.

In poor contrast images, the adjacent characters merge during binarization. We have to reduce the spread of the characters before applying a threshold to the word image. Hence, we introduce “**POWER- LAW TRANSFORMATION**” which increases the contrast of the characters and helps in better segmentation. The basic form of power-law transformation is

**s = cr γ,**

where r and s are the input and output intensities, respectively; c and γ are positive constants.

A variety of devices used for image capture, printing, and display respond according to a power-law. By convention, the exponent in the power-law equation is referred to as gamma. Hence, the process used to correct these power-law response phenomena is called gamma correction. Gamma correction is important, if displaying an image accurately on a computer screen is of concern.

In our experimentation, γ is varied in the range of 1 to 5. If c is not equal to ’1’, then the dynamic range of the pixel values will be significantly affected by scaling. Thus, to avoid another stage of rescaling after power-law transformation, we fix the value of c = 1. With γ = 1, if the power-law transformed image is passed through binarization, there will be no change in the result compared to simple binarization. When γ > 1, there will be a change in the histogram plot, since there is an increase of samples in the bins towards the gray value of zero. Gamma correction is important if displaying an image accurately on computer screen is of concern.

**3.4.5 Edge Detection**

Edge detection is the name for a set of mathematical methods which aim at identifying points in a digital image at which the image brightness changes sharply or, more technically, has discontinuities or noise. The points at which image brightness alters sharply are typically organized into a set of curved line segments termed edges.

The same problem of detecting discontinuities in 1D signal is known as step detection and the problem of finding signal discontinuities over time is known as change detection. Edge detection is a basic tool in image processing, machine vision and computer envisage, particularly in the areas of feature reveal and feature extraction.

**3.4.5.1 Edge detection techniques**

Different colours have different brightness values of particular colour. Green image has more bright than red and blue image or blue image is blurred image and red image is the high noise image.

Following are list of various edge-detection methods:-

* Sobel Edge Detection Technique
* Perwitt Edge Detection
* Roberts Edge Detection Technique
* Zero cross Threshold Edge Detection Technique
* Canny Edge Detection Technique

In our project we use “CANNY EDGE DETECTION TECHNIQUE” because of its various advantages over other edge detection techniques

**3.4.5.2 Canny Edge Detection**

The Canny Edge Detector is one of the most commonly used image processing tools detecting edges in a very robust manner. It is a multi-step process, which can be implemented on the GPU as a sequence of filters. Canny edge detection technique is based on three basic objectives.

* **Low error rate**

All edges should be found, and there should be no spurious responses. That is, the edges must be as close as possible to the true edges.

* **Edge point should be well localized:-**

The edges located must be as close as possible to the true edges. That is, the distance between a point marked as an edge by the detector and the centre of the true edge should be minimum.

* **Single edge point response**

The detector should return only one point for each true edge point. That is, the number of local maxima around the true edge should be minimum. This means that the detector should not identify multiple edge pixels where only a single edge point exist.

The essence of Canny’s work was in expressing the preceding three criteria mathematically and then attempting to find optimal solution to these formulations, in general, it is difficult to find a close form solution that satisfies all the preceding objectives. However, using numerical optimization with 1-D step edges corrupted by additive while Gaussian noise led to the conclusion that a good approximation to the optimal step edge detector is the first derivative of Gaussian.

**CHAPTER 4**

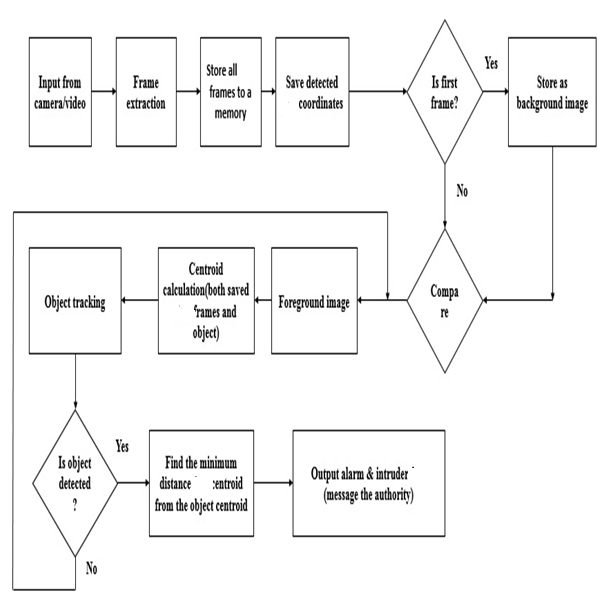
**IMPLEMENTATION**

Implementation is the carrying out, execution, or practice of a plan, a method, or any design, idea, model, specification, standard or policy for doing something. As such implementation is the action that must follow any preliminary thinking in order for something to actually happen.

**4.1 List of modules:**

* Video acquisition
* store the car frames & co-ordinates
* construction
* object detection
* post processing
* Feature selection
* Object representation
* Trajectory plot
* Video analytics processing

Figure below shows the block diagram (flow chart) for the proposed system. The problem is divided into different modules. First the video input is given to the system which is from a CCTV camera in the case of real time. But here the test is done using videos of real time situation. The videos given as input are of the same resolution as the video from a CCTV camera.



**Fig 4.1: Block diagram for proposed system**

In the second stage the video is extracted to frames for analyzing the video frame by frame at a time single frames are analyzed.

In the third stage we check for any cars in the incoming frames and if any save that frame to a memory location and keeps the co-ordinates of the cars in an array.

The next stage checks whether the frame is first frame or not. If it is first frame store it as background image after a conversion from RGB to gray. And the next coming frames are also converted to gray scale and compared with the stored background image. If there is any change is detected. That is any new object is come in next frames (moving or stationary), that will save as the foreground image. If there is no change the result of background subtraction will be zero. i.e., we will get a black image.

So now after change detection we have to ignore the moving objects and concentrate on stationary objects which are not in the scene before. So we avoid the motion changes using some thresholding method. And find the stationary objects. We avoid stationary objects of small area. Because of chance of presence of noise. And find the objects of some range of area. After this operation the resulting image will be a binary image showing objects detected as white and all other as black.

Now we calculate the centroids of objects including the cars which are saved earlier to a particular location.

Then we have to track the stationary objects in order to detect when it is abandoned. So we keep an eye on each object is alone without the presence of its owner for a particular time delay, the object is termed as abandoned, and when the object is detected as abandoned we find the minimum distance car centroid from the object centroid in order to find the car of the owner.

And finally we makes an alarm or pass a message to the authority subject to the detection of the abandoned object and displays the frame from the video in which the car of the owner is marked inside a box.

And if there is no object is detected as abandoned the process is repeated looking for any change in the incoming frames. i.e. looking for a foreground image. This process repeats.

**4.1.1 Video Acquisition**

This processing unit is the process of importing the video from a video stream and capture into sequence frames.

1. **Video Stream –** This method receives a streaming video from a file or a CCTV camera. Currently, the following video file formats are supported, mp4, avi, bmp and others. By far, this paper works with only one video from one camera or one video file at a time.
2. **Sequence Frame** – After the program reads the video file, it takes and processes each image by querying frames from the video file.
3. **Capture Image Displaying** – Creating a window in which the captured images from camera will be shown on that window.

**4.1.2 Store the Car frames & co-ordinates**

This section saves the car in each frames and face co-ordinates also. The system object used to detect face is “**vision.CascadeObjectDetector”** System object. And it is explained in detail below.

1. **vision.CascadeObjectDetector:-**

Detect objects using the Viola-Jones algorithm

**4.1.3 Construction**

detector = vision.CascadeObjectDetector creates a System object, detector, that detects objects using the Viola-Jones algorithm. The Classification Model property controls the type of object to detect. By default, the detector is configured to detect faces.

Detector = vision.CascadeObjectDetector(MODEL) creates a System object, detector, configured to detect objects defined by the input string, MODEL. The MODEL input describes the type of object to detect. There are several valid MODEL strings, such as 'FrontalCarCART', 'UpperBody', and 'ProfileCar'. See the [ClassificationModel](http://in.mathworks.com/help/vision/ref/vision.cascadeobjectdetector-class.html" \l "bs_hhwg-4) property description for a full list of available models.

detector = vision.CascadeObjectDetector(XMLFILE) creates a System object, detector, and configures it to use the custom classification model specified with the XMLFILEinput. The XMLFILE can be created using the [trainCascadeObjectDetector](http://in.mathworks.com/help/vision/ref/traincascadeobjectdetector.html) function or OpenCV (Open Source Computer Vision) training functionality. You must specify a full or relative path to the XMLFILE, if it is not on the MATLAB path.

detector = vision.CascadeObjectDetector (Name, Value) configures the cascade object detector object properties. You specify these properties as one or more name-value pair arguments. Unspecified properties have default values.

1. **To detect a feature:**
2. Define and set up your cascade object detector using the constructor.
3. Call the step method with the input image, I, the cascade object detector object, detector, points PTS, and any optional properties. See the syntax below for using the step method.

Use the step syntax with input image, I, the selected Cascade object detector object, and any optional properties to perform detection.

BBOX = step (detector, I) returns BBOX, an M-by-4 matrix defining M bounding boxes containing the detected objects. This method performs multiscale object detection on the input image, I. Each row of the output matrix, BBOX, contains a four-element vector, [x y width height], that specifies in pixels, the upper-left corner and size of a bounding box. The input image I, must be a grayscale or true color (RGB) image.

BBOX = step (detector, I, roi) detects objects within the rectangular search region specified by roi. You must specify roi as a 4-element vector, [x y width height], that defines a rectangular region of interest within image I. Set the 'UseROI' property to true to use this syntax.

**4.1.4 Object Detection**

Object detection is the process of finding out the area of interest as per user’s requirement. Here we have proposed the algorithm for object detection using frame difference method (One of the background subtraction algorithms). Steps are given as:

* 1. Read all the image frames generated from the video, which are stored on a variable or storage medium.
  2. Convert them into greyscale image using rgb2gray ( ) from coloured image.
  3. Store the first frame as background image
  4. Calculate the difference as |frame[i]-background frame|
  5. If the difference is greater than a threshold (rth), then the value is considered to be the part of foreground otherwise background (no change is detected)
  6. Update the value of i by incrementing with one.
  7. Repeat the step c to d up to the last image frame.
  8. End the process.

**4.1.5 Post Processing:**

The detected object in the previous phase may lead to have a problem of connectivity and it may also have some holes which may be useless for object representation. Therefore here we need to have some post processing which will reduce the problem of handling holes and the connectivity of pixels within object region. Mathematical morphological analysis is one of post processing approach which leads to enhance the segmented image in order to improve the required result. In the proposed method we have used the erosion and dilation iteratively so that an object will clearly appear in foreground while the rest useless blobs will be removed. Morphological operations are useful to obtain the useful components from the image. These components may be the object boundary, region, shape and skeleton etc.

1. **Dilation:** Dilation is an increasing transform, used to fill small holes and narrow chasm in the objects.
2. **Erosion:** Erosion as morphological transformation can be used to find the contours of the objects. It is used to shrink or reduce the pixel values.

**4.1.6 Feature Selection**

The features like centroid of an object, height and width of an object are selected so that it is easy to plot the location of non-rigid body/objects with frame to frame. The proposed method evaluates the centroid of detected object in each frame. It is assumed that after the morphological operations there will not be any false object. And then a centroid of the object in two dimensional frames can be calculated as the average of the pixels in x and y coordinates belonging to the object.

**Cx=total moments in x-direction/total area------ (1)**

**Cy=total moments in y-direction/total area------ (2)**

**4.1.7 Object Representation**

Here we are using centroid and the rectangular shape to cover the object boundary to represent the object. After calculating the centroid, find the Width Wi and Height Hi of the object by extracting the positions of pixels Pxmax and Pxmin which has the maximum and minimum values of X Coordinate related to the object. Similarly, for the Y coordinates, calculate Pymax and Pymin.

**4.1.8 Trajectory Plot**

After the process of object detection using frame differencing method, the detected components are given as input to the tracking process to plot the trajectory. The frame differencing algorithm will give all the pixel values of the detected object. The centroid of the objects is calculated by using the equation. Here input will be the pixel values of the object and the output will be the rectangular area around the object. This process will calculate the centroid, height and width of the object for the purpose of trajectory plotting.

**4.1.9 Video Analytics Processing**

Segmentation is the process of detecting changes and extracting relevant changes for further analysis and qualification. Changed pixels from previous positions are referred to as "Foreground Pixels"; those that do not change are called "Background Pixels". The segmentation method used here is Background Subtraction. Image Pixels remaining after the background has been subtracted are the foreground pixels. The key factor which is used to identify foreground pixels by means of “Degree of change" in segmentation and can vary depending on the application.

The segmentation result is one or more foreground blobs. A blob is nothing but a collection of connected pixels.

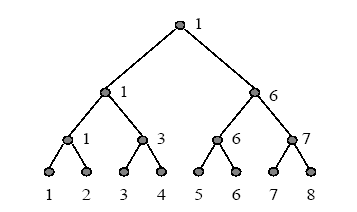
**4.2 METHODOLOGY**

**4.2.1 SVM with a binary tree classification strategy**

The first analyzed approach proposes a face detection system using linear support vector machines with a binary tree classification strategy. The result of this technique are very good as we conclude: “The experimental results show that the SVMs are a better learning algorithm than the nearest center approach for face recognition.”

General information subsection describes the basic theory of SVM for two class classification. A multi-class pattern recognition system can be obtained by combining two class SVMs. Usually there are two schemes for this purpose. One is the one-against-all strategy to classify between each class and all the remaining; The other is the one-against-one strategy to classify between each pair. While the former often leads to ambiguous classification, the latter one was used for the presented face recognition system.

A bottom-up binary tree for classification is proposed to be constructed as follows: suppose there are eight classes in the data set, the decision tree is shown in the figure below where the numbers 1-8 encode the classes. By comparison between each pair, one class number is chosen representing the “winner” of the current two classes. The selected classes (from the lowest level of the binary tree) will come to the upper level for another round of tests. Finally, the unique class will appear on the top of the tree.

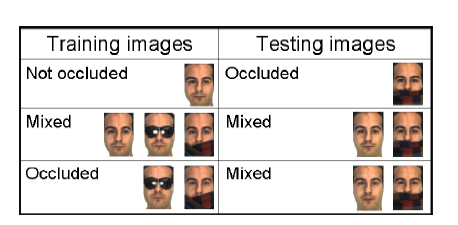


**Figure 4.2: The bottom-up binary tree used for classification**

Denote the number of classes as c, the SVMs learn  discrimination functions in the training stage, and carry out comparisons of  times under the fixed binary tree structure. If  does not equal to the power of 2, we can decompose  as where . Because any natural number (even or odd) can be decomposed into finite positive integers which are the power of 2. If  is odd, , and if  is even . It can be noticed that the decomposition is not unique, but the number of comparisons in the test stage is always .

**4.2.2 Face Detection and Recognition with Occlusions**

The next approach analyzes a more edge case, where the face is not entirely present in the analyzed image. So it is searched to derive a criterion for SVM that can be employed in the three cases defined in the figure below: not occluded, mixed and occluded. The classical criteria of SVM cannot be applied to any of the three cases, because SVM assumes all the features are visible. So a new algorithm is implemented named by the Partial Support Vector Machines (PSVM) to distinguish it from the standard criteria used in SVM.



**Figure 4.3: Occlusion cases taken into account**

The goal of PSVM is similar to that of the standard SVM – to look for a hyper plane that separate the samples of any two classes as much as possible. In contrast with traditional SVM, in PSVM the separating hyper plane will also be constrained by the incomplete data. In the proposed PSVM, the set of all possible values for the missing entries of the incomplete training sample are treated as an affine space in the feature space such that a criterion which minimizes the probability of overlap between this affine space and the separating hyper plane is designed. To model this, the angle between the affine space and the hyper plane in the formulation is incorporated. The resulting objective function is shown to have a global optimal solution under mild conditions, which require that the convex region defined by the derived criterion is close to the origin. Experimental results demonstrate that the proposed PSVM approach provides superior classification performances than those defined in the literature.

**4.2.3 PCA and SVM for Face Detection**

In this section we are describe, a novel method is proposed for eliminating most of the non-face area in gray images, so that the detection time is shortened and the detection accuracy is improved. Face area has different pixel character with most of the non-face area. By analyzing histogram distributions, it shows face and non-face area have different histogram distribution. The histogram of face areas has Gaussian-like distribution but non-face area histogram has irregular distribution. According to the histogram distribution feature, the face potential area can be choosen.

If the histogram is distributed in a small range, its mean value is a high value and if the histogram distribution is in a wide range, it has a small mean value.

The histogram of face image is a Gaussian-like distribution; the mean value is an intermediate value. By a number of tests, the histogram mean value of face potential area is choosen in a fixed range. So if the mean value of a sample area is falling in that range, this sample area is selected as a face potential area. Otherwise, it is filtered as non-face area.

Furthermore, for face detection, an algorithm combining PCV and SVM is used, it is a coarse-to-fine process to detect face region.

The processing consists of three steps:

* Step 1: face potential is selected using histogram distribution feature. Face and non-face area have different histogram distribution. The histogram of face areas has Gaussian-like distribution but non-face area histogram has irregular distribution.
* Step 2: PCA is used to decrease the dimension of face feature space. At this step, 1000 sample images of size 19×19 are trained
* Step 3: SVM is used as classifier to verify face candidate. It is trained by face and non-face samples which are represented by PCA.

**4.3 Face Recognition Technique**

There are lots of techniques for face recognition. These are:

1. Eigenfaces (Eigenfeatures).
2. Neural Networks.
3. Dynamic Link Architecture.
4. Hidden Markov Model.
5. Feature Based Matching.
6. Template Matching.
7. **Eigenfaces**

Eigenface is one of the most thoroughly investigated approaches to face recognition. It is also known as Karhunen- Loève expansion, eigenpicture, eigenvector, and principal component. Principal component analysis to efficiently represent pictures of faces. They argued that any face images could be approximately reconstructed by a small collection of weights for each face and a standard face picture (eigenpicture). The weights describing each face are obtained by projecting the face image onto the eigenpicture. There is substantial related work in multimodal biometrics. For example used face and fingerprint in multimodal biometric identification, and used face and voice. However, use of the face and ear in combination seems more relevant to surveillance applications

1. **Neural Networks**

The attractiveness of using neural networks could be due to its non linearity in the network. Hence, the feature extraction step may be more efficient than the eigenface methods.

One of the first artificial neural networks (ANN) techniques used for face recognition is a single layer adaptive network called WISARD which contains a separate network for each stored individual . The way in constructing a neural network structure is crucial for successful recognition. But it is not used for more number of persons. If the number of persons increases, the computing expense will become more demanding. In general, neural network approaches encounter problems when the number of classes (i.e., individuals) increases. Moreover, they are not suitable for a single model image recognition test because multiple model images per person are necessary in order for training the systems to “optimal” parameter setting.

1. **Graph Matching**

Graph matching is another approach to face recognition. Reference presented a dynamic link structure for distortion invariant object recognition which employed elastic graph matching to find the closest stored graph. Dynamic link architecture is an extension to classical artificial neural networks. Memorized objects are represented by sparse graphs, whose vertices are labeled with a multi-resolution description in terms of a local power spectrum and whose edges are labeled with geometrical distance vectors. Object recognition can be formulated as elastic graph matching which is performed by stochastic optimization of a matching cost function.

In general, dynamic link architecture is superior to other face recognition techniques in terms of rotation invariance; however, the matching process is computationally expensive.

1. **HIDDEN MARKOV MODELS (HMMS)**

Stochastic modeling of non stationary vector time series based on (HMM) has been very successful for speech applications. Applying this method to human face recognition. Faces were intuitively divided into regions such as the eyes, nose, mouth, etc., which can be associated with the states of a hidden Markov model. Since HMMs require a one-dimensional observation sequence and images are two dimensional, the images should be converted into either 1D temporal sequences or 1D spatial sequences.

1. **Feature based matching.**

Geometrical feature matching techniques are based on the computation of a set of geometrical features from the picture of a face. The fact that face recognition is possible even at coarse resolution as low as 8x6 pixels when the single facial features are hardly revealed in detail, implies that the overall geometrical configuration of the face features is sufficient for recognition. The overall configuration can be described by a vector representing the position and size of the main facial features, such as eyes and eyebrows, nose, mouth, and the shape of face outline.

Geometrical feature matching based on precisely measured distances between features may be most useful for finding possible matches in a large database such as a Mug shot album. However, it will be dependent on the accuracy of the feature location algorithms. Current automated face feature location algorithms do not provide a high degree of accuracy and require considerable computational time.

1. **Template Matching**

A simple version of template matching is that a test image represented as a two-dimensional array of intensity values is compared using a suitable metric, such as the Euclidean distance, with a single template representing the whole face. There are several other more sophisticated versions of template matching on face recognition. One can use more than one face template from different viewpoints to represent an individual's face.

In general, template-based approaches compared to feature matching are a more logical approach. In summary, no existing technique is free from limitations. Further efforts are required to improve the performances of face recognition techniques, especially in the wide range of environments encountered in real world.

**6.1 Principal Components Analysis**

Finally we come to Principal Components Analysis (PCA). What is it? It is a way of identifying patterns in data, and expressing the data in such a way as to highlight their similarities and differences. Since patterns in data can be hard to find in data of high dimension, where the luxury of graphical representation is not available, PCA is a powerful tool for analyzing data. The other main advantage of PCA is that once you have found these patterns in the data, and you compress the data, i.e. by reducing the number of dimensions, without much loss of information. This technique used in image compression.

1. **Method**

Step 1: Get some data

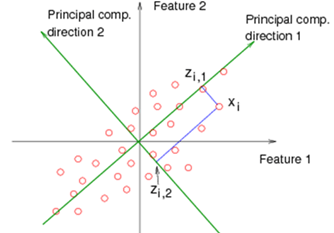
Step 2: Subtract the mean

Step 3: Calculate the covariance matrix

Step 4: Calculate the eigenvectors and eigenvalues of the co-variance matrix.

Step 5: Choosing components and forming a feature vector

Step 6: Deriving the new data set



**Figure 4.4: PCA Example**

The above figure show the how the PCA works on two different direction how the features are compared.

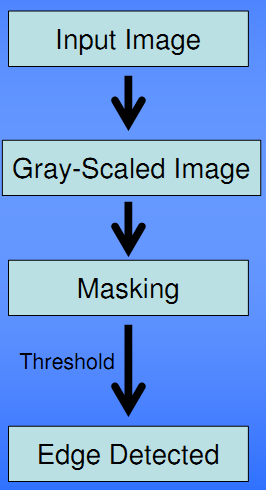
**6.2 EDGE DETECTION**

Edge detection refers to the process of identifying and locating sharp discontinuities in an image. The discontinuities are abrupt changes in pixel intensity which characterize boundaries of objects in a scene. Classical methods of edge detection involve convolving the image with an operator (a 2-D filter), which is constructed to be sensitive to large gradients in the image while returning values of zero in uniform regions. There are an extremely large number of edge detection operators available, each designed to be sensitive to certain types of edges. Variables involved in the selection of an edge detection operator include Edge orientation, Noise environment and Edge structure.

The geometry of the operator determines a characteristic direction in which it is most sensitive to edges. Operators can be optimized to look for horizontal, vertical, or diagonal edges. Edge detection is difficult in noisy images, since both the noise and the edges contain high frequency content. Attempts to reduce the noise result in blurred and distorted edges. Operators used on noisy images are typically larger in scope, so they can average enough data to discount localized noisy pixels. This results in less accurate localization of the detected edges. Not all edges involve a step change in intensity. Effects such as refraction or poor focus can result in objects with boundaries defined by a gradual change in intensity .

The operator needs to be chosen to be responsive to such a gradual change in those cases. So, there are problems of false edge detection, missing true edges, edge localization, high computational time and problems due to noise etc.

Therefore, the objective is to do the comparison of various edge detection techniques and analyze the performance of the various techniques in different conditions. There are many ways to perform edge detection. However, the majority of different methods may be grouped into two categories:

****

**Fig. 4.4: Edge detection process**

**6.3 Gradient based Edge Detection:**

The gradient method detects the edges by looking for the maximum and minimum in the first derivative of the image.

**6.4 Laplacian based Edge Detection:**

The Laplacian method searches for zero crossings in the second derivative of the image to find edges. An edge has the one-dimensional shape of a ramp and calculating the derivative of the image can highlight its location. Suppose we have the following signal, with an edge shown by the jump in intensity below: Suppose we have the following signal, with an edge shown by the jump in intensity below. .

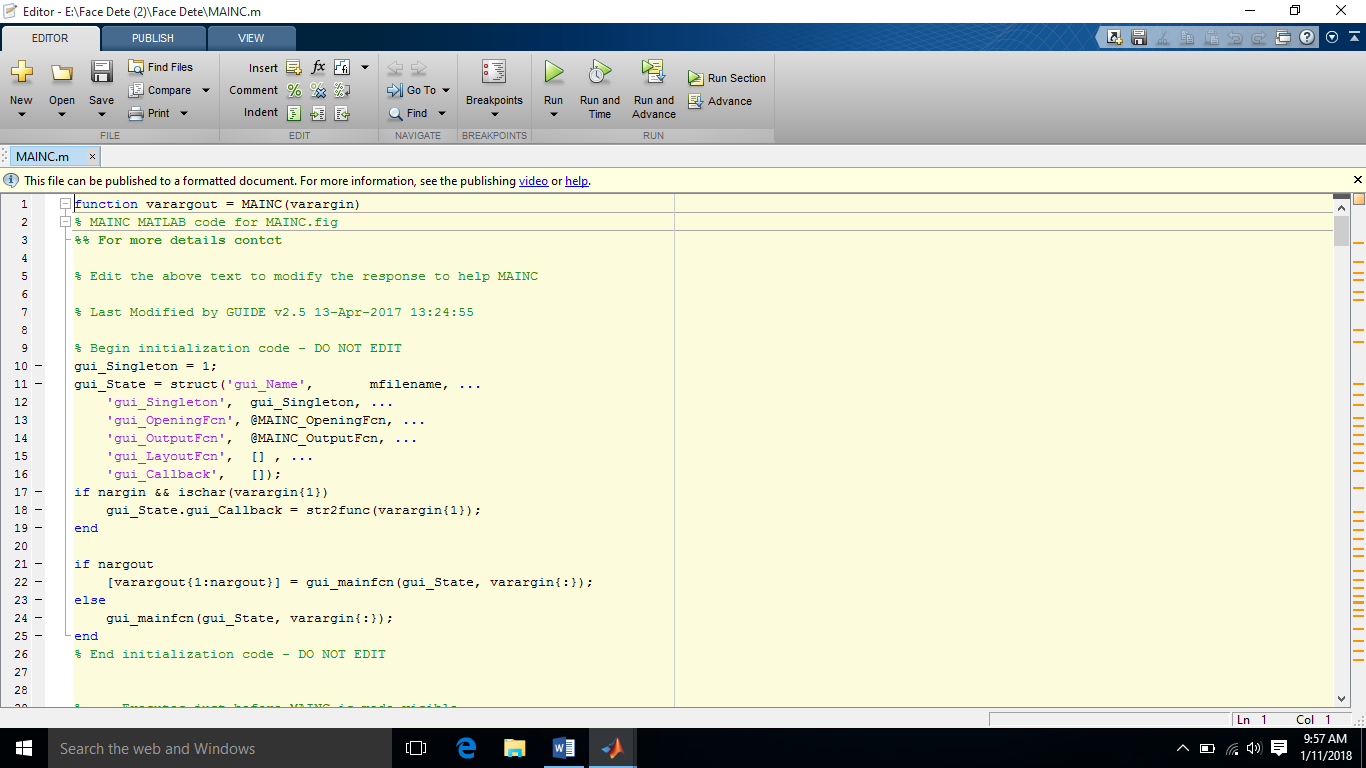
**7. FEATURE EXTRACTION**

When the input data to an algorithm is too large to be processed and it is suspected to be notoriously redundant (much data, but not much information) then the input data will be transformed into a reduced representation set of features (also named features vector). Transforming the input data into the set of features is called feature extraction. If the features extracted are carefully chosen it is expected that the features set will extract the relevant information from the input data in order to perform the desired task using this reduced representation instead of the full size input.

**CHAPTER 5**

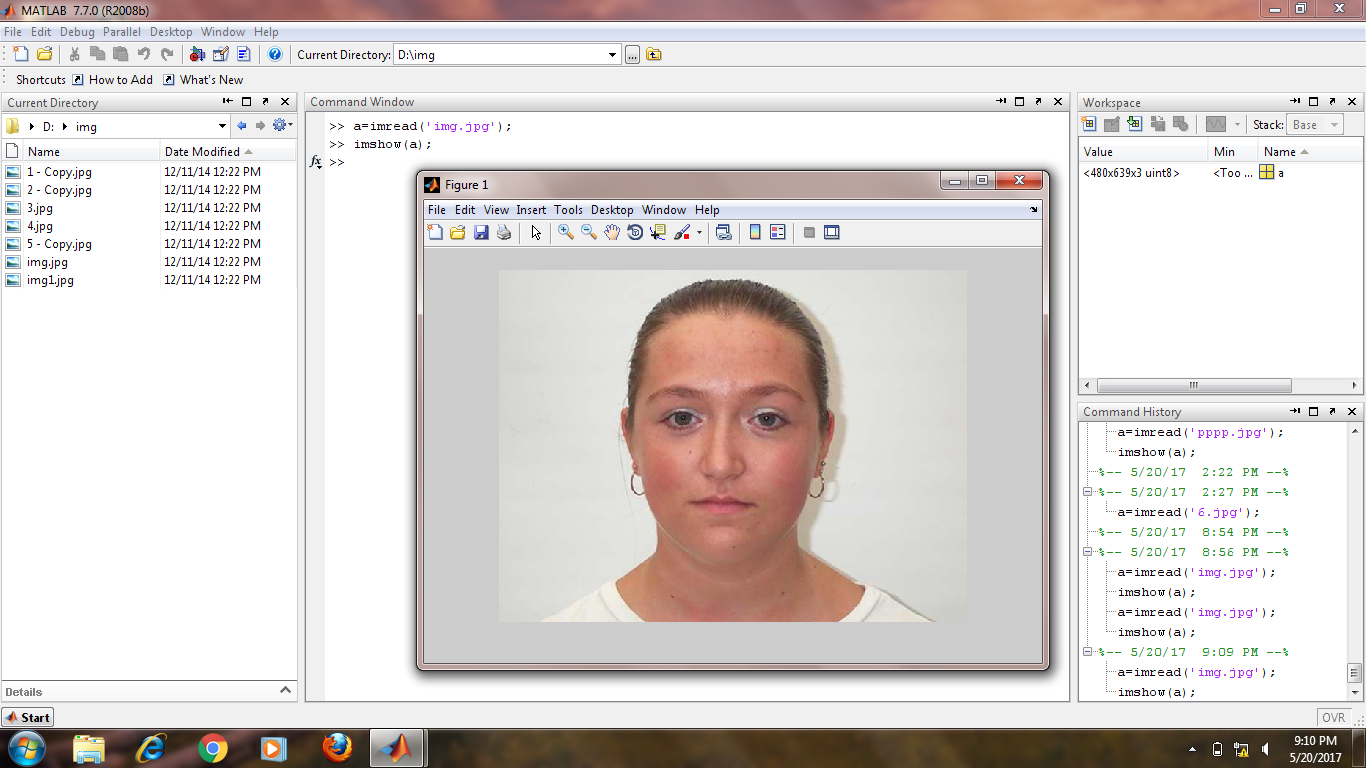
**RESULTS AND DISCUSSION**

**Snapshot 1:** In this snapshot we are using the code to get the face detection results.



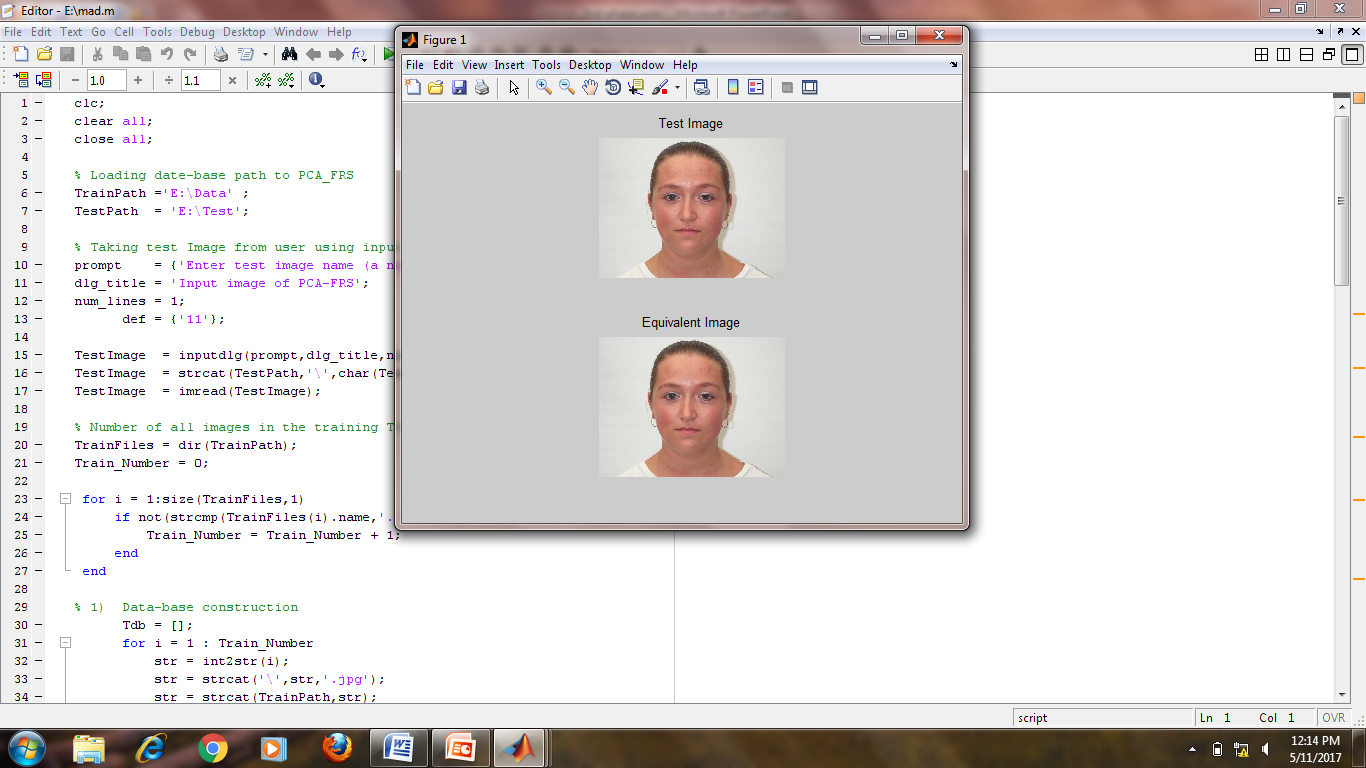
**Figure 5.1: In this figure we are running the code**

**Snapshot 2:** this snapshot shows that we are reading the image from pc by using command line.



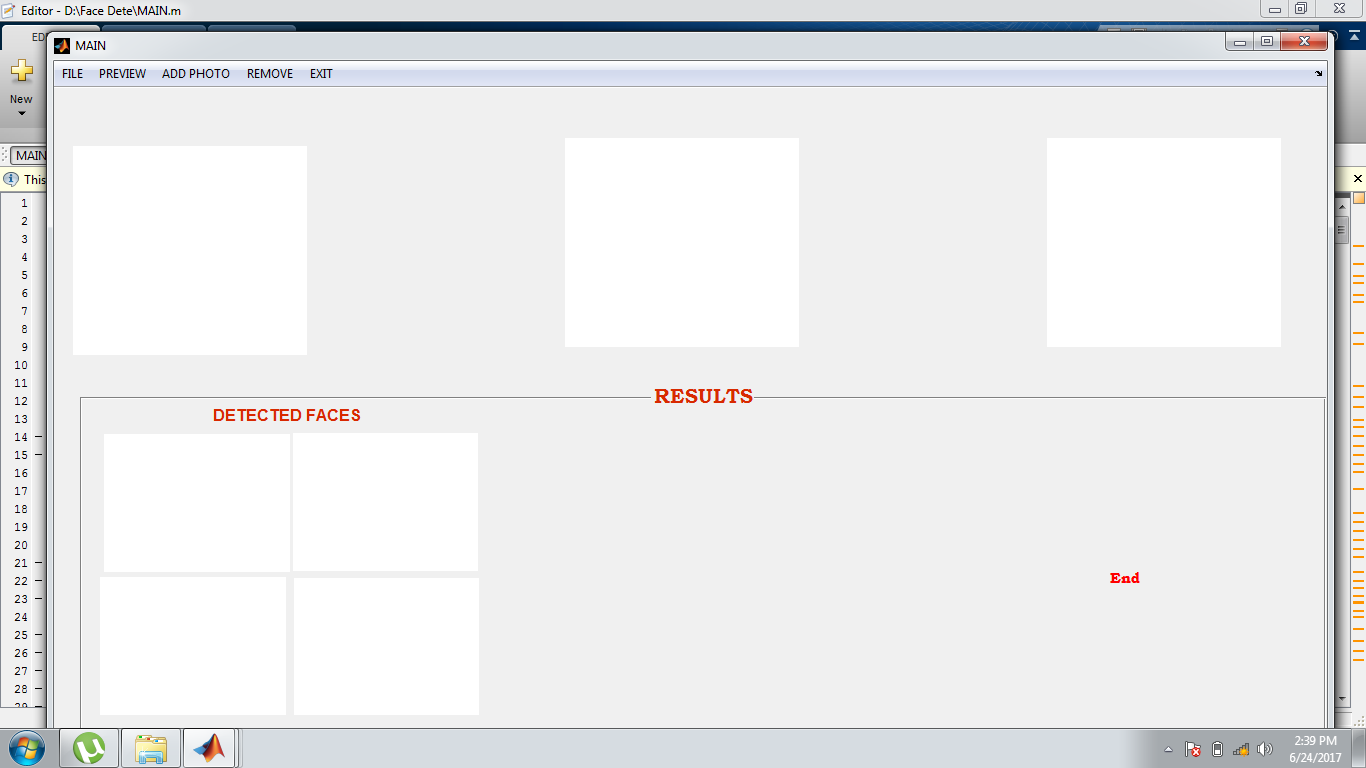
**Figure 5.2: Reading the image**

**Snapshot 3: This snapshot shows the identification of the image.**



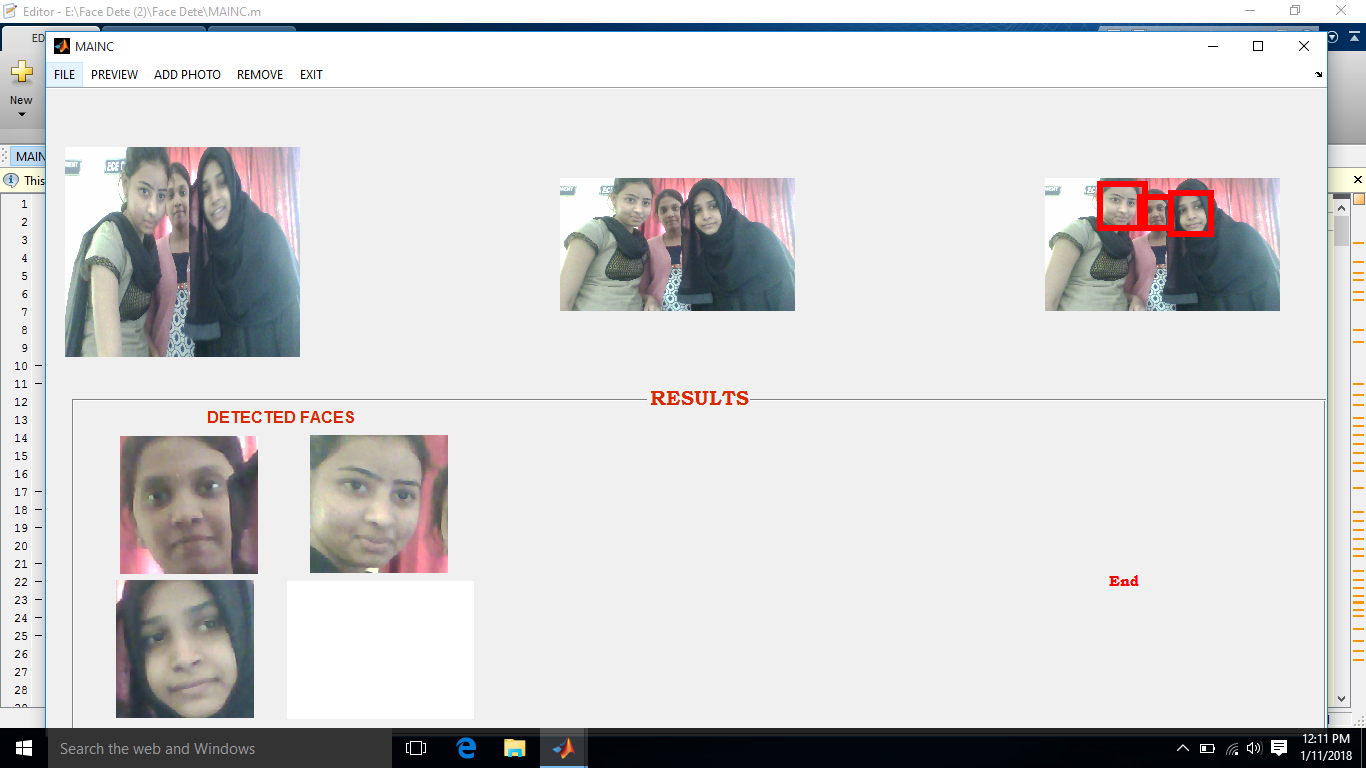
**Figure 5.3: Identification of the image**

**Snapshot 4:** In tis snapshot we are showing the face detection program in GUI



**Fig 5.4: Identification of the image**

**Snapshot 5:** In this snapshot we are showing the6 face detection through camera

****

**Figure 5.5: Face detection camera**

**CHAPTER 6**

**APPLICATIONS**

* **Used for storing and handling highly confidential data:**

In this we are storing the training data images in the gallery and we are comparing it with the test image and it is highly confidential data

* **Unauthorized transfer of data can be prohibited:**

The unauthorized people cannot have permission to access or transfer of data.

* **This security system can be applied to system applications:**

We are using the face recognition technique as locker pattern in the large industries.

* **Real time applications:**

In which the applications execute in given time such as in,

* + Military applications.
  + Atm machines.
  + Personal digital assistance.
  + Computers and laptops
* **College and Schools Computers:**

For attendance purpose recently we are using the facing recognition technique

* **Border Control/Airports:**

Facial recognition is increasingly used at airports for security and processing purposes.

* **Consumer/Residential Biometrics:**

The biometric technology is used for authentication and identification of individuals by comparing facial features from an image with a stored facial database.

* **Financial and Transactional:**
* **Justice/Law Enforcement:**
* **Bank and hospital:**

Facial recognition technology is starting to be more widely used in the banking industry as a fraud detection tool, as well as in other industries, such as to identity cheaters at casinos

**CHAPTER 7**

**CONCLUSION AND FUTURE WORK**

We presented a novel learning-based algorithm for under-sampled face recognition. We advocated the learning of an auxiliary dictionary from external data for modeling intra-class image variants of interest, and utilized a residual function in a joint optimization formulation for identifying and disregarding corrupted image regions due to occlusion. As a result, the proposed algorithm allows one to recognize occluded face images, or those with illumination and expressions variations, even only one or few gallery images per subject are available during training.

**FUTURE SCOPE**

Future work would be concentrated on improving light compensation, morphological operations and optimizing the threshold for the eigenface method. Thus, a greater precision in both detection and recognition of faces would be obtained. An intressant approach would be to combine our holistic method with a feature based method to achieve higher performance.

**REFERENCES**

[1].Lianzhi Tan ; Zhifeng Li ; Qiao Yu **Deep face attributes recognition using spatial transformer network** Information and Automation (ICIA), 2016 IEEE International Conference on 1-3 Aug. 2016

[2].Kai Liu ; Seungbin Moon **Robust dual-stage face recognition method using PCA and high-dimensional-LBP** Information and Automation (ICIA), 2016 IEEE International Conference on 1-3 Aug. 2016

[3].M Shujah Islam Sameem ; Tehreem Qasim ; Khush Bakhat **Real time recognition of human faces** Open Source Systems & Technologies (ICOSST), 2016 International Conference on 15-17 Dec. 2016

[4].Bappaditya Mandal **Face recognition: Perspectives from the real world Control, Automation,** Robotics and Vision (ICARCV), 2016 14th International Conference on 13-15 Nov. 2016

[5].S. V. Tathe ; A. S. Narote ; S. P. Narote **Face detection and recognition in videos** India Conference (INDICON), 2016 IEEE Annual 16-18 Dec. 2016

[6].Shanshan Guo ; Shiyu Chen ; Yanjie Li **Face recognition based on convolutional neural network and support vector machine Information and Automation (ICIA)**, 2016 IEEE International Conference on 1-3 Aug. 2016

[7].M. Kan, S. Shan, Y. Su, D. Xu, and X. Chen, **“Adaptive discriminant learning for face recognition,” Pattern Recognition.** vol. 46, no. 9, pp. 2497–2509, Sep. 2013

[8].W. Deng, J. Hu, and J. Guo, **“Extended SRC: Undsersampled face recognition via intraclass variant dictionary,”** IEEE Trans. Pattern Anal. Mach. Intel., vol. 34, no. 9, pp. 1864–1870, Sep. 2012.

[9].P. Zhu, L. Zhang, Q. Hu, and S. C. K. Shiu, **“Multi-scale patch based collaborative representation for face recognition with margin distribution optimization,”** in Proc. 12th Eur. Conf. Compute. Vis., 2012, pp. 822–835.

[10].X. Tan, S. Chen, Z.-H. Zhou, and F. Zhang, “Face recognition from a single image per person: A survey,” Pattern Recognit., vol. 39, no. 9, pp. 1725–1745, Sep. 2006.

[11].T. Ahonen, A. Hadid, and M. Pietikainen, “Face description with local binary patterns: Application to face recognition,” IEEE Trans. +++++Pattern Anal. Mach. Intell., vol. 28, no. 12, pp. 2037–2041, Dec. 2006.

[12].J. Zou, Q. Ji, and G. Nagy, “A comparative study of local matching approach for face recognition,” IEEE Trans. Image Process., vol. 16, no. 10, pp. 2617–2628, Oct. 2007.

[13].J. Lu, Y.-P. Tan, and G. Wang, “Discriminative multimanifold analysis for face recognition from a single training sample per person,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 1, pp. 39–51, Jan. 2013.

[14].M. Yang, L. Van Gool, and L. Zhang, “Sparse variation dictionary learning for face recognition with a single training sample per person,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Dec. 2013, pp. 689–696.

[15].M. Yang, L. Zhang, J. Yang, and D. Zhang, “Robust sparse coding for face recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2011, pp. 625–632.

[16].M. Aharon, M. Elad, and A. Bruckstein, “K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation,” IEEE Trans. Signal Process., vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[17].J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, “Robust face recognition via sparse representation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 31, no. 2, pp. 210–227, Feb. 2009.

[18].C.-F. Chen, C.-P. Wei, and Y.-C. F. Wang, “Low-rank matrix recovery with structural incoherence for robust face recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2012, pp. 2618–2625.