CHAPTER 1

INTRODUCTION

Person Identification is a technique that is mainly used to identify criminals based on the clues given by the eyewitnesses. Based on the clues we develop an image by using the image that we have in our database and then we compare it with the images already we have. To identify any criminals we must have a record that generally contains name, age, location, previous crime, gender, photo, etc. The primary task at hand is, given still or video images require the identification of the one or more segmented and extracted from the scene, where upon it can be identified and matched. The word image is defined as an exact or analogous representation of a being or a thing. The image or monochrome image such as black and white paragraph is represented as two dimensional light intensity function f(x,y) where x and y denotes spatial co-ordinates. A digital image is an image of f (x, y) that has been digitized both in spatial co ordinate and brightness. The elements of such a digital array are called image elements, picture elements or pixels.

A novel based Bag of Words (BoW) method is proposed for robust person Identification. In our approach, a face image is partitioned into multiple blocks, dense SIFT features and SURF features are then calculated and vector quantized into different codewords on each block respectively. Finally, histograms of codeword distribution on each local block are concatenated to represent the face image. Experimental results on BIO Identification database show that only using one neutral expression frame per person for training, our method can obtain excellent face recognition results on face images with extreme expressions, variant illumination, and partial occlusions. Our method also achieves an average recognition rate of 90 % on database. First, preprocessing and detection of face from image and then extract features from the face detected to give as input to the classifier for classification. Second, model and build a classifier for classifying the image and codeword is distributed on each classified vector. Various machine learning classifiers such as Support Vector Machine (SVM), KNN, K Means classifiers are used.

1.1 Purpose

Person Identification is done through the processing of face recognition technology which is used in computer application for automatically recognizing or confirming a person from a digital image. One of simplest way to do this is by comparing selected faces image features from the face database which might be present in the organization. In a large scale organization, the main goal is to reduce the number of comparison of the image from the respective face dataset because dataset might contain huge amount of data/ faces, it should also overcome the simple problem of face recognition like lighting, brightness.

1.2 Motivation

Digital image processing deals with the manipulation of images through a digital computer.it is the subfield of signals and systems but focus particularly on images, and also focuses on developing a computer system that is able to perform processing on an image. The input of that system is digital image and the system process that image using efficient algorithms and gives an image as an output, visual information transmitted in the form of digital image is the area to be involved today he importance of computer imaging is derived from the fact that our primary sense is our visual sense, and the information that be conveyed in images has been known throughout the centuries to be extraordinary (one picture is worth a thousand words).The main motivation behind identifying the face is that growing of security systems and advance in image capture technology and to increase the recognition rate and to keep the computational complexity low.

1.3 Problem Definition

Person Identification is becoming very important, the main objective is to achieve a desirable identification accuracy with tolerable response time, increasingly electronically information the ability to achieve high accurate person identification is needed, traditional Personal Identification where based on Personal Identification Number (PIN) and/or ID card have a number of inherent disadvantages, it may be stolen, lost, forgotten or misplaced which is unable to authorize a person. Biometrics refers to identification of a person based on his/her physiological or behavioral characteristics it is more reliable and more capable in differentiating between an authorized person and fraudulent imposter , since it require that the person to be identified by a physical present at the point of Identification and relies on it.

We have used Bag OF Words model to represent the images as visual words and to extract the facial features by using feature descriptors such as SURF and SIFT and used SVM for classification which builds histograms and is evaluated with different metrics and to aim to use different combinations of feature descriptors to extract the key points to get the high possibility of accuracy.

CHAPTER 2

LITERATURE SURVEY

2.1 Biometrics

Biometrics is used in the process of authentication of a person by verifying or Identifying that a user requesting a resource is authorized user or not. It is used to describe the characteristic or a process which is used to measure the physiological and behavioral characteristic that can be used for automated recognition, its is used in computer science as a form of identification and access control, a traditional means of access control includes password or personal identification number which are identifiers that is unique to the individuals which is more reliable in verification. There are many types of biometric system like fingerprint recognition, face detection and recognition, iris recognition etc., these traits are used for human identification in surveillance system, criminal identification. Advantages of using these traits for identification are that they cannot be forgotten or lost. These are unique features of a human being which is being used widely.

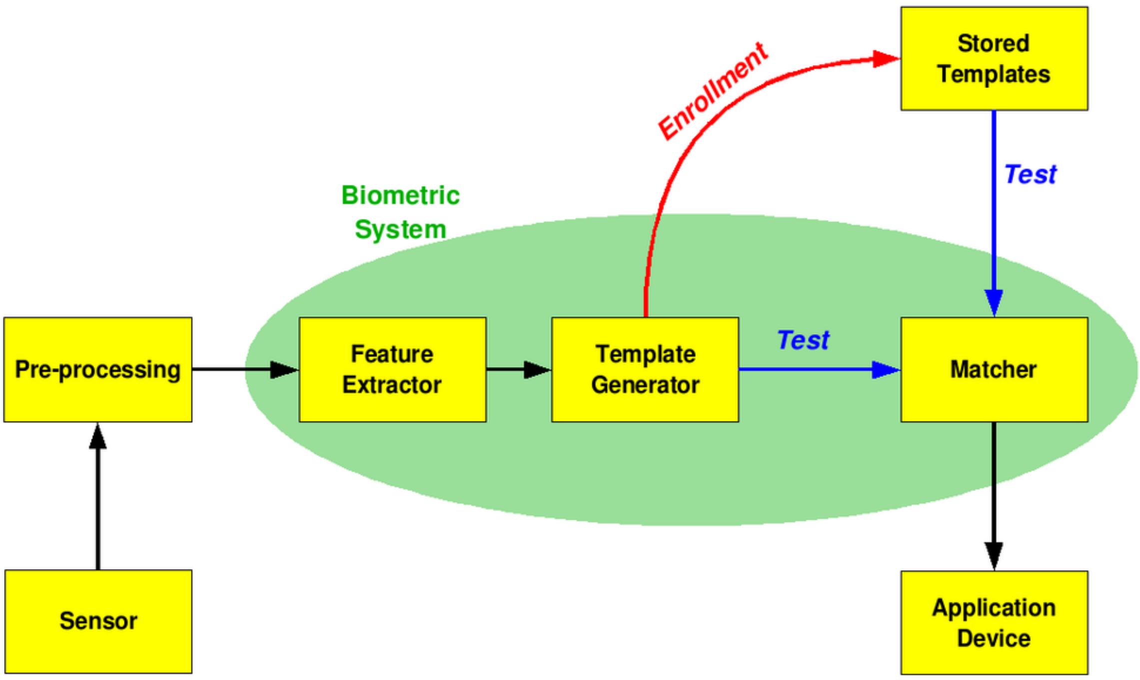


Fig 2.1 Biometric System

2.1.1 Types of Biometrics

Biometrics can be classified into two classes:

Physiological

Face Recognition: Uses an image or series of images either from a camera or photograph to recognize a person.

Fingerprint Recognition: An extremely useful biometrics technology since fingerprints have long been recognized as a primary and accurate identification method.

Hand Geometry: Hand geometry systems are commonly available in two main forms. Full hand geometry systems take an image of the entire hand for comparison while Two Finger readers only image two fingers of the hand.

Iris Face Recognition: Analysis of the iris of the eye, which is the colored ring of tissue that surrounds the pupil of the eye.

DNA: DNA has been called the "ultimate identifier", Identifies information from every cell in the body in a digital form.

Behavioral

Signature verification: The movement of the pen during the signing process rather than the static image of the signature.

Voice: Voice recognition is not the same as speech recognition, it is speaker recognition, considered both physiological and behavioral.

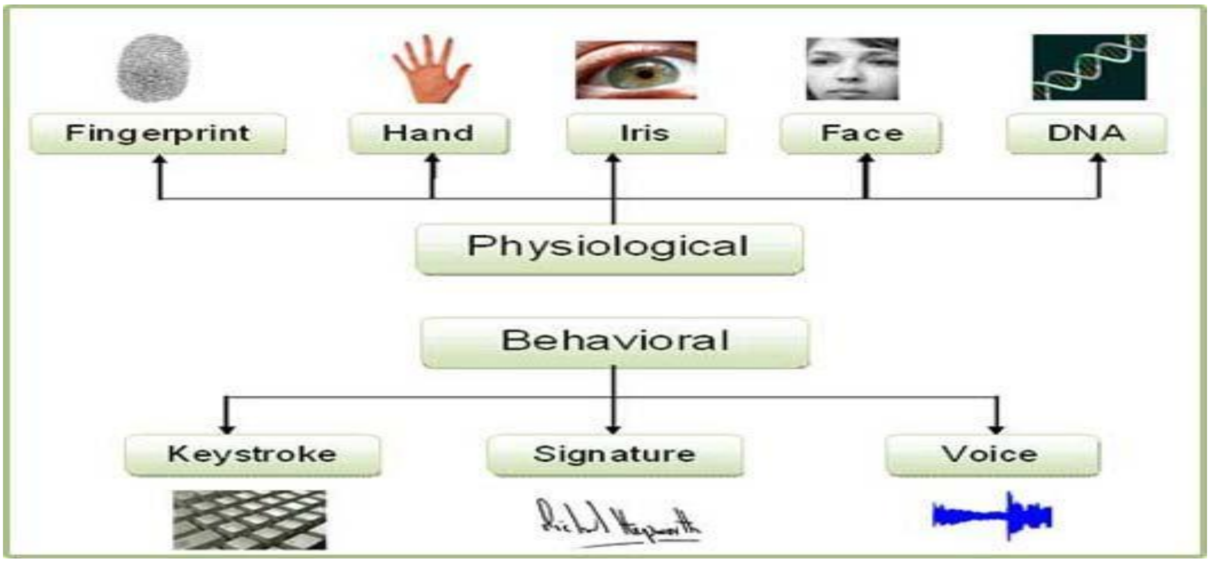


Fig 2.2 Types of Biometrics

2.2 Image Processing

Image processing is a method to convert an image into digital form and perform some operations on it, in order to get an enhanced image or to extract some useful information from it. It is a type of signal dispensation in which input is image, like video frame or photograph and output may be image or characteristics associated with that image. Usually Image Processing system includes treating images as two dimensional signals while applying already set signal processing methods to them. it is processing of images using mathematical operations by using any form of signal processing for which the input is an image, a series of images, or a video, such as a photograph or video frame ,the output of image processing may be either an image or a set of characteristics or parameters related to the image. Most image-processing techniques involve treating the image as a two-dimensional signal and applying standard signal-processing techniques to it. Image processing usually refers to digital image processing, but optical and analog image processing also are possible. Image processing are computer graphics and computer vision. In computer graphics, images are manually *made* from physical models of objects, environments, and lighting, instead of being acquired (via imaging devices such as cameras) from *natural* scenes, as in most animated movies. Computer vision, on the other hand, is often considered *high-level* image processing out of which a machine/computer/software intends to decipher the physical contents of an image or a sequence of images.

2.3 Computer Vision

Computer vision is a field that includes methods for acquiring, processing, analyzing, and understanding images and, in general, high-dimensional data from the real world in order to produce numerical or symbolic information. A theme in the development of this field has been to duplicate the abilities of human vision by electronically perceiving and understanding an image. Understanding in this context means the transformation of visual images (the input of retina) into descriptions of world that can interface with other thought processes and elicit appropriate action. This image understanding can be seen as the disentangling of symbolic information from image data using models constructed with the aid of geometry, physics, statistics, and learning theory. Computer vision has also been described as the enterprise of automating and integrating a wide range of processes and representations for vision perception. Computer vision is concerned with the theory behind artificial systems that extract information from images. The image data can take many forms, such as video sequences, views from multiple cameras, or multi-dimensional data from a medical scanner. As a technological discipline, computer vision seeks to apply its theories and models to the construction of computer vision systems.

2.4 Facial Recognition

A facial recognition system is a computer application capable of identifying or verifying a person from a digital image or a video frame from a video source. One of the ways to do this is by comparing selected facial features from the image and a facial database. It is typically used in security systems and can be compared to other biometrics such as fingerprint or eye iris recognition systems. Face recognition has become a popular area of research in computer vision, it is typically used in network security systems and access control systems but it is also useful in other multimedia information processing areas. One of its application is criminal face identification. Criminal record generally contains personal information about particular person along with the photograph. To identify any criminal we need some identification regarding particular person or persons, which are given by eyewitnesses. Based on the details given by the eyewitnesses, the further investigation would be carried out. In most cases the quality and resolution of the recorded image segments is poor and hard to identify a face. In this paper, we have classified image processing operations into three categories; low, medium and high level to process and analyze a given face. This paper presents better results than conventional methods in use relating to the face recognition process that are used in criminal identification. Recently, it has also become popular as a commercial identification and marketing tool. Some facial recognition algorithms identify facial features by extracting landmarks, or features, from an image of the subject's face. For example, an algorithm may analyze the relative position, size, and/or shape of the eyes, nose, cheekbones, and jaw. These features are then used to search for other images with matching features. Other algorithms normalize a gallery of face images and then compress the face data, only saving the data in the image that is useful for face recognition. A probe image is then compared with the face data. One of the earliest successful systems is based on template matching techniques applied to a set of salient facial features, providing a sort of compressed face representation. Recognition algorithms can be divided into two main approaches, geometric, which looks at distinguishing features, or photometric, which is a statistical approach that distills an image into values and compares the values with templates to eliminate variances.

Fig 2.3 Face Recognition

2.5 Face Identification

Face Identification is a technique that is mainly used to identify criminals based on the clues given by the eyewitnesses. Based on the clues we develop an image by using the image that we have in our database and then we compare it with the images already we have. To identify any criminals we must have a record that generally contains name, age, location, previous crime, gender, photo, etc. The primary task at hand is, given still or video images require the identification of the one or more segmented and extracted from the scene, where upon it can be identified and matched. The word image is defined as an exact or analogous representation of a being or a thing. The image or monochrome image such as black and white paragraph is represented as two dimensional light intensity function f(x,y) where x and y denotes spatial co-ordinates. A digital image is an image of f (x, y) that has been digitized both in spatial co-ordinate and brightness. The elements of such a digital array are called image elements, picture elements or pixels. Biometric technologies have been evolved as an enchanting solution to perform secure identification and personal verification. The need for highly secure identification and personal verification technologies is becoming apparent as the level of security breaches and transaction fraud increases. The increasing use of biometric technologies in high security applications and beyond has created the requirement for highly dependable face recognition systems. The Face recognition system is used to verify an identity of a person by matching a given face against a database of known faces. It has become alternative to traditional identification and authentication methods such as the use of keys, ID cards and passwords. Face recognition involves computer recognition of personal identity based on geometric or statistical features that are derived from the face images. Even though human can detect and identify faces in a scene easily, building an automated system is challenging. Face recognition technology can be applied to a wide variety of application areas including access control for PCs, airport surveillance, private surveillance, criminal identification and for security in ATM transactions. In addition, face recognition system is moving towards the next-generation smart environment where computers are designed to interact more like humans. In recent years, considerable progress has been made in the area of face recognition with the development of many other useful techniques. The advances in computing technology have facilitated the development of real-time vision modules that interact with humans in recent years. Examples abound, particularly in biometrics and human computer interaction as the information contained in faces needs to be analyzed for systems to react accordingly. For biometric systems that use faces as nonintrusive input modules, it is imperative to locate faces in a scene before any recognition algorithm can be applied. An intelligent vision based user interface should be able to tell the focus of the user (i.e., where the user is looking at) in order to respond accordingly. To detect facial features accurately for applications such as digital cosmetics, faces need to be located and registered first to facilitate further processing. It is evident that face detection plays an important and critical role for the success of any face processing systems. The face detection problem is challenging as it needs to account for all the possible appearance variation caused by change in illumination, facial features, occlusions, etc. In addition, it has to detect faces that appear at different scale, pose, with in plane rotations. Often the size of the image is very large, the processing time has to be very small and usually real-time constraints have to be met. Therefore, during the last decades there has been an increasing interest in the development and the use of parallel algorithms in image processing. Face detection is attached with finding whether or not there are any faces in a given image (usually in gray scale) and, if present, return the image location and content of each face. This is the first step of any fully automatic system that analyzes the information contained in faces. This work focuses on how to make parallel computations by partitioning the image into manageable and meaningful parts for efficient calculations and results.

2.5.1 Face Identification Technique

Face identification is a term that includes several sub problems. The technique for face identification comprises of three steps: face detection, feature extraction and face recognition.

Face Detection: Face detection is defined as the process of detecting faces from images and scenes. So, the system positively identifies a certain image region as a face. This procedure has many applications like face tracking, pose estimation or compression.

Feature Extraction: Feature extraction- involves obtaining relevant facial features from the data. These features could be certain face regions, variations, angles or measures, which can be human relevant (e.g. eyes spacing) or not. This phase has other applications like facial feature tracking or emotion recognition.

Face Recognition: It is a one-to-many matching process that compares a query face image against all the template images in a face database to determine the identity of the query face. The identification of the test image is done by locating the image in the database that has the highest similarity with the test image. The identification process is a "closed" test, which means the sensor takes an observation of an individual that is known to be in the database.

2.6 Bag of Words Model

In computer vision, the bag-of-words model (BoW model) can be applied to image classification, by treating image features as words. In document classification, a bag of words is a sparse vector of occurrence counts of words; that is, a sparse histogram over the vocabulary. In computer vision, a *bag of visual words* is a vector of occurrence counts of a vocabulary of local image features. To represent an image using BoW model, an image can be treated as a document. Similarly, "words" in images need to be defined too. To achieve this, it usually includes following three steps: feature detection, feature description, and codebook generation. A definition of the BoW model can be the "histogram representation based on independent features". Content based image indexing and retrieval (CBIR) appears to be the early adopter of this image representation technique. After feature detection, each image is abstracted by several local patches. Feature representation methods deal with how to represent the patches as numerical vectors. These vectors are called feature descriptors. A good descriptor should have the ability to handle intensity, rotation, scale and affine variations to some extent. One of the most famous descriptors is Scale-invariant feature transform (SIFT) and Speeded up Robust Features (SURF). Speeded up Robust Features (SURF) is a local feature detector and descriptor that can be used for tasks such as object recognition or registration or classification or 3D reconstruction. It is partly inspired by the scale-invariant feature transform (SIFT) descriptor. The standard version of SURF is several times faster than SIFT and claimed by its authors to be more robust against different image transformations than SIFT. To detect interest points, SURF uses an integer approximation of the determinant of Hessian blob detector, which can be computed with 3 integer operations using a pre computed integral image. Its feature descriptor is based on the sum of the Haar wavelet response around the point of interest. These can also be computed with the aid of the integral image. SURF descriptors can be used to locate and recognize objects, people or faces, to make 3D scenes, to track objects and to extract points of interest. SIFT converts each patch to 128-dimensional vector. After this step, each image is a collection of vectors of the same dimension. The final step for the BoW model is to convert vector-represented patches to "code words" (analogous to words in text documents), which also produces a "codebook" (analogy to a word dictionary). A code word can be considered as a representative of several similar patches. One simple method is performing k-means clustering over all the vectors. Code words are then defined as the centers of the learned clusters. The number of the clusters is the codebook size (analogous to the size of the word dictionary).Thus, each patch in an image is mapped to a certain code word through the clustering process and the image can be represented by the histogram of the code words. The BoW model is an analogy to the BoW model in NLP, generative models developed in text domains can also be adapted in computer vision. Simple Naïve Bayes model and hierarchical Bayesian models can be used. The simplest one is Naïve Bayes classifier basic idea is that each category has its own distribution over the codebooks, and that the distributions of each category are observably different. The face category may emphasize the code words which represent "nose", "eye" and "mouth". Given a collection of training examples, the classifier learns different distributions for different categories. Since images are represented based on the BoW model, any discriminative model suitable for text document categorization can be tried, such as support vector machine (SVM). Computer vision researchers have developed several learning methods to leverage the BoW model for image related tasks, these methods can be generative and discriminative models. The confusion matrix can be used as an evaluation metric.

Fig 2.4 Bag Of Words Model

2.7 Feature Detector

A Feature Detector is an algorithm what takes an image and outputs locations i.e pixel co ordinates of significant areas in your image, the concept of feature detection refers to methods that aim at computing image abstractions and make local decisions at every image point whether there is image feature of a given type at that point or not the resulting features will be subsets of the image domain in the form of isolated points curves and connected regions.

2.8 Feature Descriptor

A feature descriptor is an algorithm which takes an image and outputs feature descriptors/feature vectors. Feature descriptors encode interesting information into a series of numbers and act as a sort of numerical that can be used to differentiate one feature from another. Ideally this information would be invariant under image transformation, so we can find the feature again even if the image is transformed in some way. An example would be SIFT, which encodes information about the local neighborhood image gradients the numbers of the feature vector and SURF.

2.9 SIFT (Scale Invariant Feature Transform)

The Scale Invariant Feature Transform (SIFT), has emerged as a cut edge methodology in general object recognition as well as for other machine vision tasks. One of the interesting features of the SIFT approach is the capability to capture the main gray-level features of an object’s view by means of local patterns extracted from a scale-space decomposition of the image. In this respect, the SIFT approach is similar to the Local Binary Patterns method with the difference of producing a view-invariant representation of the extracted 2D patterns from the given image. SIFT is based on the analysis of existing invariance-based feature detection methods at that time. SIFT has good stability and invariance. It detects local key points, which contain a large amount of information. Because of its unique advantages, it has become a popular research topic. Many researchers constantly work hard to improve. It is a potent algorithm developed by David Lowe, that dubs as both a key point detector and feature descriptor and has been ubiquitously employed in a number of computer vision tasks such as object recognition, image stitching, visual mapping etc. and is preferred for the wide gamut of standout characteristics it possesses such as its robustness to variations in scale, zoom, translation, rotation, illumination.

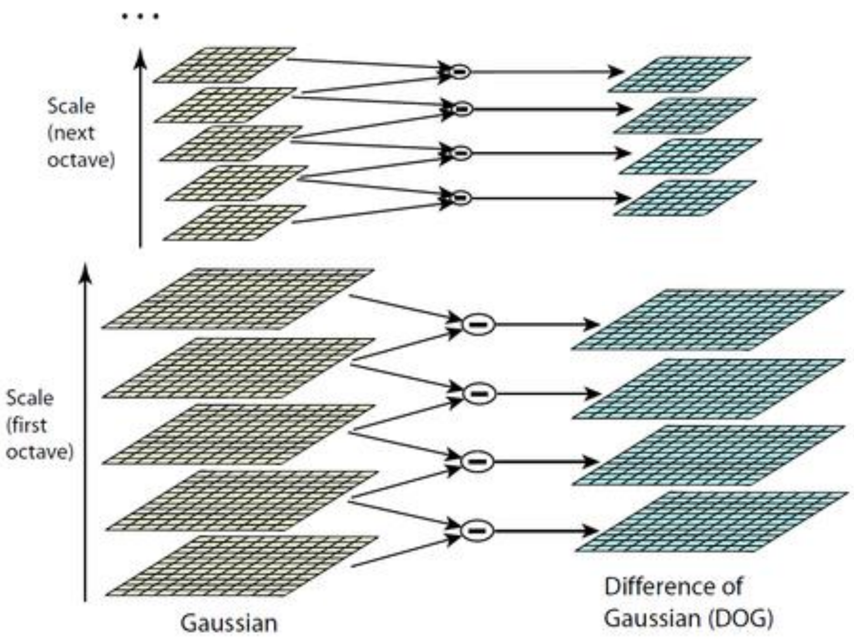


Fig 2.5 SIFT Representation

2.10 SURF (Speeded Up Robust Feature)

Speed up Robust Feature (SURF) uses nothing but multi-scale theory and the feature detector which uses Hessian Matrix, reason is that it would prove more accurate key-point and help to improve the speed of the algorithm. In Hessian Matrix uses the concepts of generalization, where in key point of an images found by assign a value for the interesting point. When compare to the SIFT algorithm SURF is quite fast. As per lot of paper one could say SURF is three times faster than SIFT. SURF (Speeded up Robust Features) is a computationally less expensive alternative to SIFT that was developed by Herbert Bay in 2006, which has demonstrated better performance and accuracy over SIFT. SURF has several standout characteristics like repeatability, distinctiveness, robustness, scale and rotation invariance, and can be computed quickly as it relies on integral images for image convolutions to reduce the computation time by employing a fast Hessian matrix based measure. Applying this algorithm on any face data base is always challenging task. Here in this paper for the normal approach, they have added two more component into the SURF algorithm .First, by using discriminant image filtering that help to maximize the key point in the SURF algorithm (named as d-surf). And in the next set they are trying to increase the further strengthen the image by using multispectral images instead of the board band of the image.

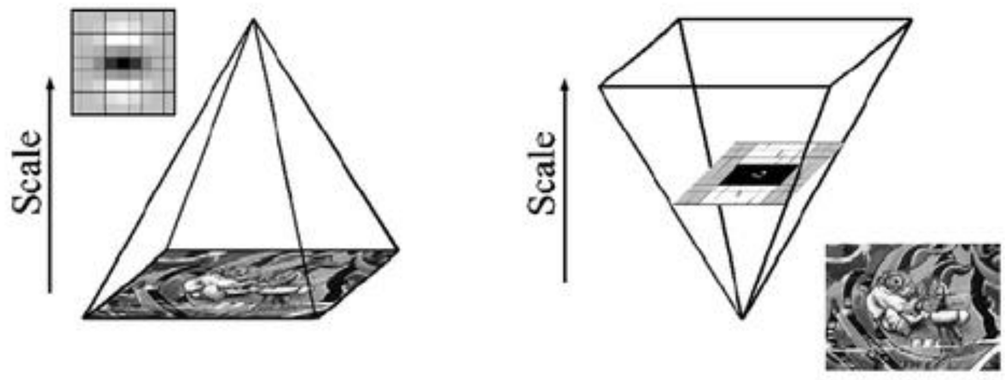


Fig 2.6 SURF Representation

2.11 Descriptor Matching

The descriptors that are generated for the query face (input face that needs to be compared against the database) need to be compared against the descriptors that are generated for the database faces so that we can match them to see if they belong to the same person. For this purpose an appropriate descriptor matcher needs to be chosen to match the faces. Many techniques are available and the most effective one is found to be FLANN (Fast Library for Approximate Nearest Neighbor) that was proposed by M.Muja and D.Lowe. FLANN finds correspondences between the descriptors of the query face and the database faces and if the matched descriptors are within a given threshold it we classify the given face under the match database face. FLANN is widely used for Face Recognition purposes when algorithms like

SIFT,SURF and so on are used to perform feature descriptor generation. It is mature, automated and very reliable in determining whether a given face matches another in the database and hence we have used this for out experiments.

2.12 OpenCV

The human brains divides the vision signal into many channels and many streams with different kinds of information into your brain. Our brain has an attention system which identifies the task in a dependent way , computer vision is just a grid of numbers, any numbers within the grid contains large components which gives less information, the aim of image processing is to help the computer to understand the content of an image , which can be achieved through the development of different algorithms for image processing.

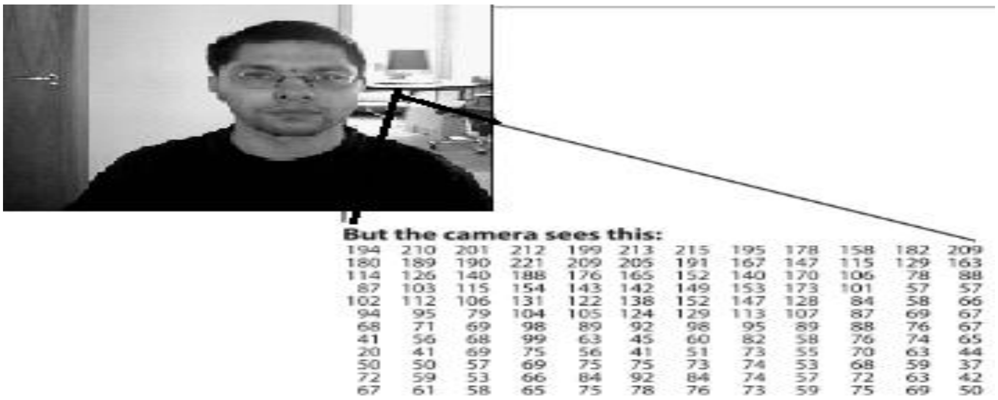


Fig 2.7 Internal Representation of Image

2.13 Limitations of Image Processing

Recognition is affected by changes in lighting, the person's hair, the age, and if the person wear glasses. It requires camera equipment for user identification, thus, it is not likely to become popular until most PCs include cameras as standard equipment.

2.14 Need of Security in Image Processing

Computer security system has also seen the application of face recognition tools. To prevent any sort of intruder from changing files or transacting with others when the authorized individual leaves the computer terminal for a short time, users are continuously authenticated, checking that the individual in front of the computer screen or at a user is the same authorized person who logged in. Since growing technology security is of utmost concern. With the increase in cybercrime, providing only network security is not sufficient. Security provided to images like blue print of company projects, secret images of concern to the army or of company's interest, using image steganography and stitching is beneficial. As the text message is encrypted using AES algorithm and embedded in a part of the image the text message is difficult to find. More over since the secret image is broken down into parts and then sent to the receiver. Many techniques have been developed to do so like digital, visual cryptography were used, In spatial domain the image pixels in the spatial domain are arranged in order to incorporate the data to be embedded.

Fig 2.8 Face as security measure

CHAPTER 3

SYSTEM REQUIREMENT SPECIFICATION

3.1 Scope

The scope of the Person Identification is that it works very well under constrained conditions in case of presence of noise and illumination changes does not rely on modality and produces robust results. Considering all the requirements, identification system that use face recognition seems to be more potential for wide-spread application. It mainly focuses on identifying the right person from the preprocessed stored image with given static image or video frame.

There are several areas where identification would be useful.

- Social Interaction: In Social interactions identifying a person places a major role , where face images can be captured from a distance without touching the person identified.
- Crime Deterrent: In Crime purpose the faces images are captured by recording and helps to identify a right person.

3.2 Hardware Requirement

PC with minimum 250GB Hard Disk

Minimum 1 GB RAM

1.7 Ghz processor

Keyboard

Mouse

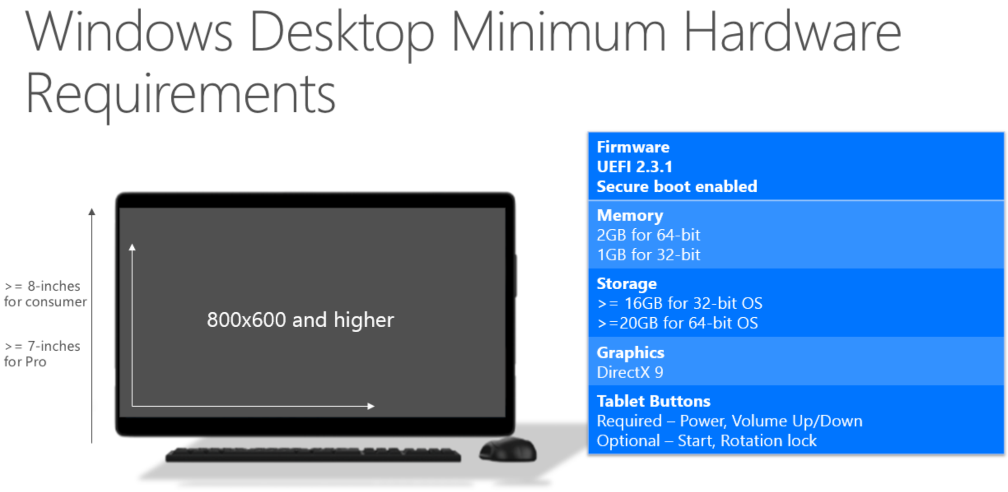


Fig 3.1 Hardware requirement specifications

3.3 Software Requirement

Microsoft Visual Studio 2010 or higher

OpenCV 2.4.5 or higher

ASP.NET code for GUI

3.4 General Description

3.4.1 Product Perspective

It is basically person identification system, where unknown and known images are stored in file as input the system tries to match the images, provides user interface to compute with different descriptors and classifies produces output.

3.4.2 User Characteristics

With the GUI user is able to compute the bag of words with SURF and Bag of words with SIFT model, based on the input through the file the images are matched and descriptors found are displayed on the image with a graph displaying the number of keypoints extracted.

3.4.3 General Constraints

The constraint on Bag of model is that during the codebook generation the semantic information is lost due to the low level features extracted and through the clustering visual features.

3.4.4 Assumptions

The bag words model is mainly used in the classification problem to classify the given image and tag with the specified object, we are using combination of both BOW and the different descriptors for feature extraction which resulted in better results.

3.5 Specific Requirements

FE-1: OPENCV Libraries should be properly imported in visual studio.

FE-2: Images are taken through the file input.

FE-3: Training and testing is done the images.

FE-4: Extracted features are given as input to the codebook generation.

FE-5: For binning the values are in the powers of 2 to plot the histogram.

FE-6: Histograms represents the small patches of the images.

3.6 Functional Requirements

Feature Detection – System should be able to detect the features from the images.

Feature Representation- System should be able to represent the features

Clustering- System should be able to cluster the images with equal partitioning.

Codebook- System should be able to generate the codebook with similar features extracted from the images.

3.7 Non- Functional Requirements

3.7.1 Performance Requirements

Since each feature is matched with all the extracted features the computation is tie consuming.

3.7.2 Software quality attributes

The system should be user friendly to do computation.

3.8 Gantt Chart

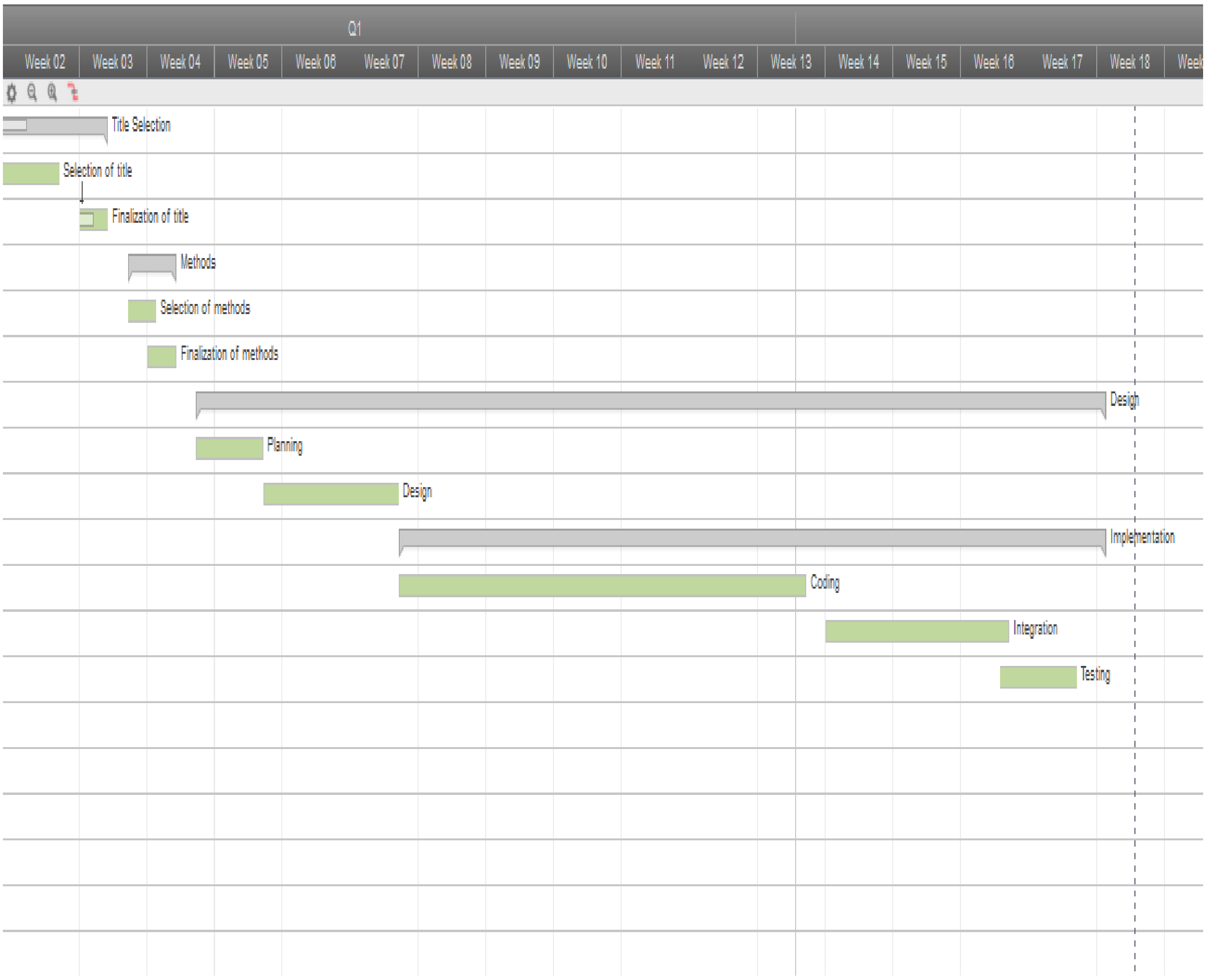


Fig 3.2 Gantt chart

CHAPTER 4

SYSTEM DESIGN

4.1 Block Diagram

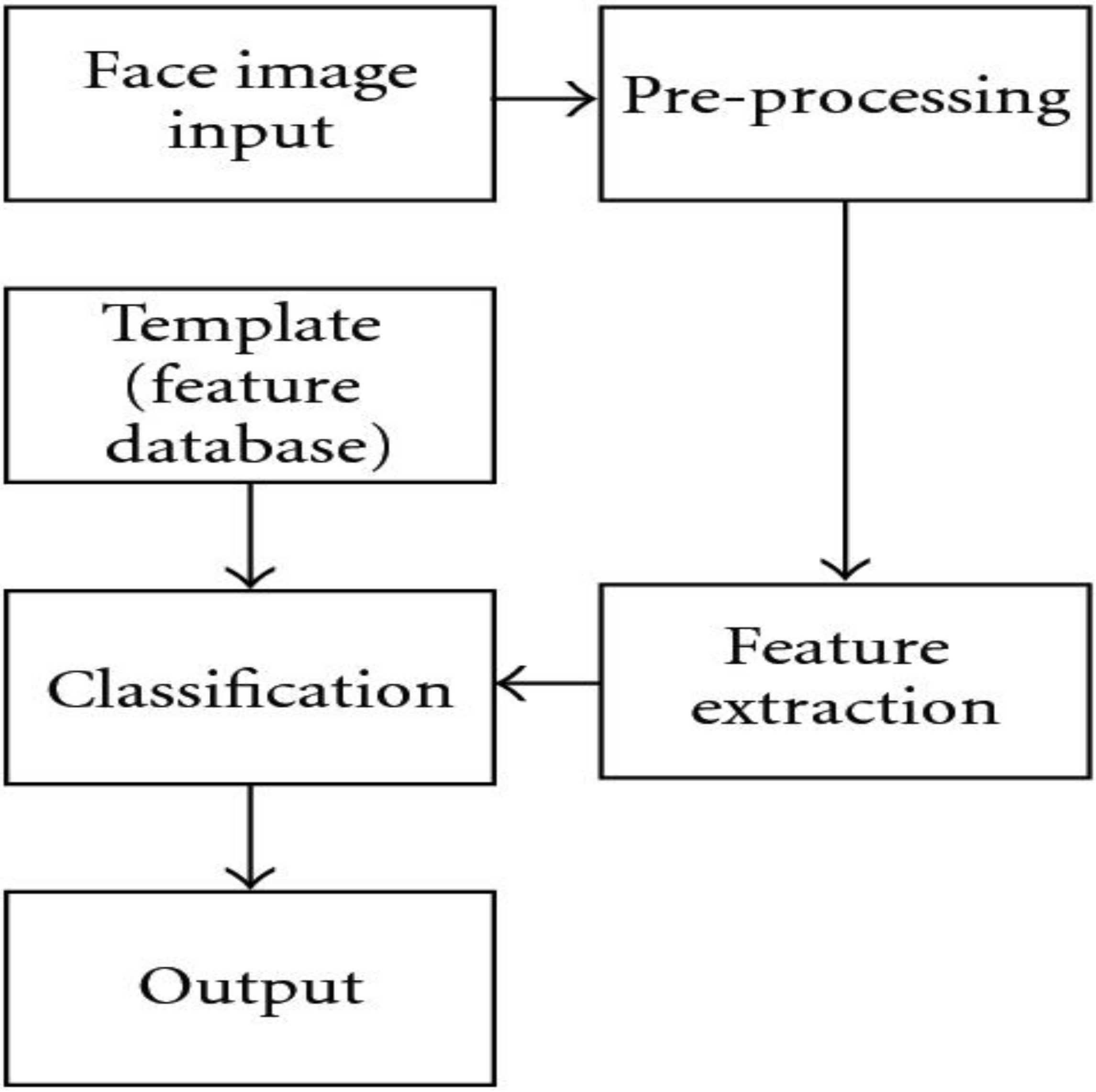


Fig 4.1 Block Diagram

4.2 Architectural Diagram

A schematic representation of the architecture for the “Person Identification System”

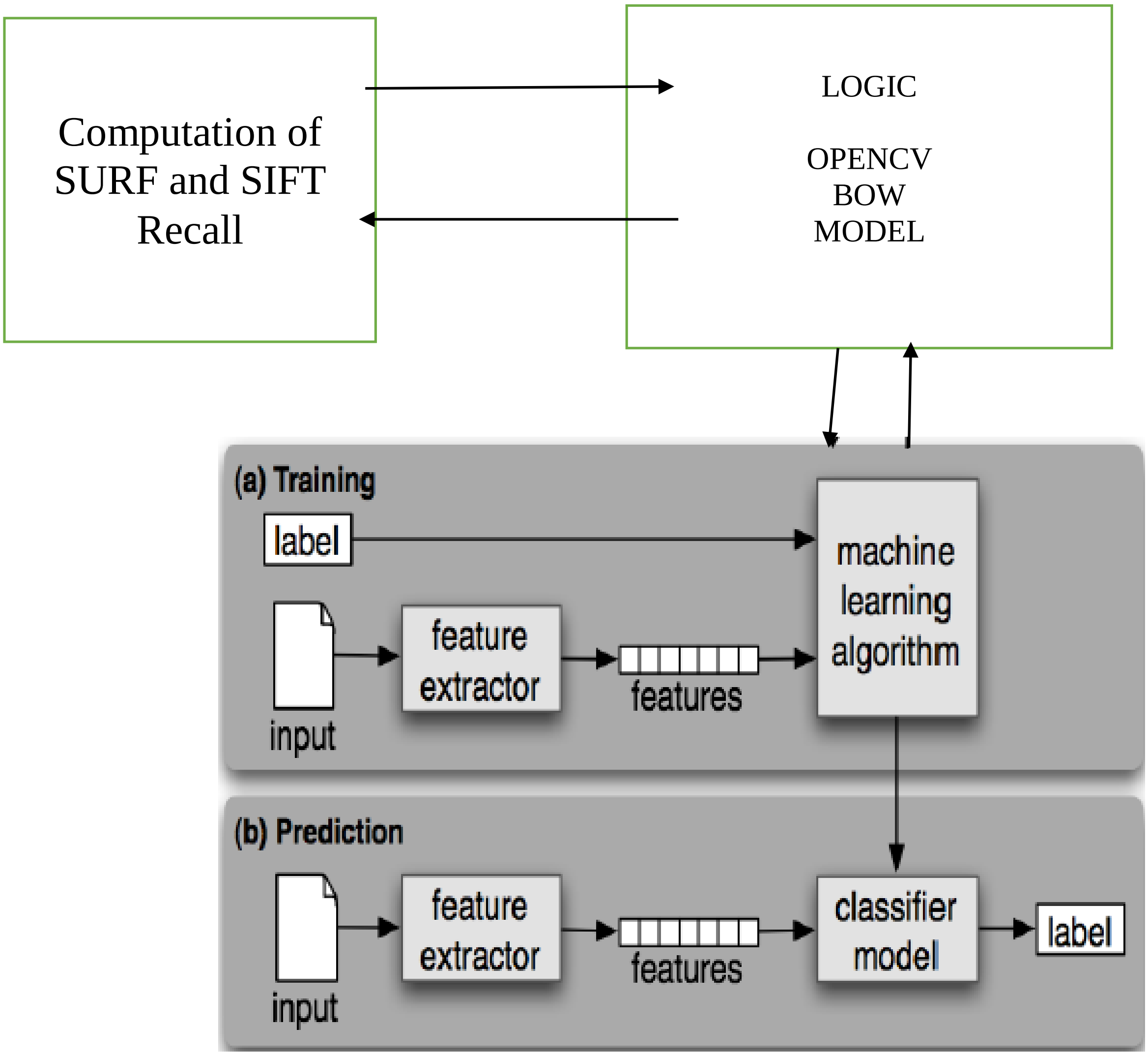


Fig 4.2 Architectural Diagram

This system is GUI based , where user can click the button to compute the SURF and SIFT with Bag of words and all computation is done in the code , freature extraction from the images , finding descriptors generating codebook displaying it as histograms visual words of image patches and evaluating through performance metrics and displaying recall result.

4.3 Use Case Diagram

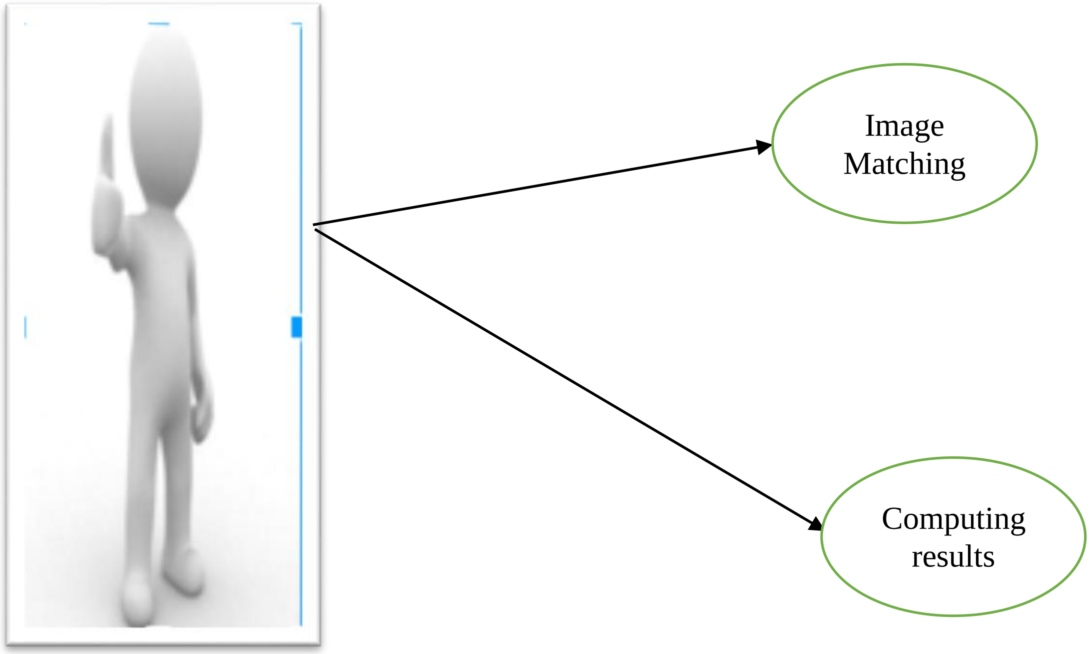


Fig. 4.3 Use case diagram

CHAPTER 5

DETAILED DESIGN

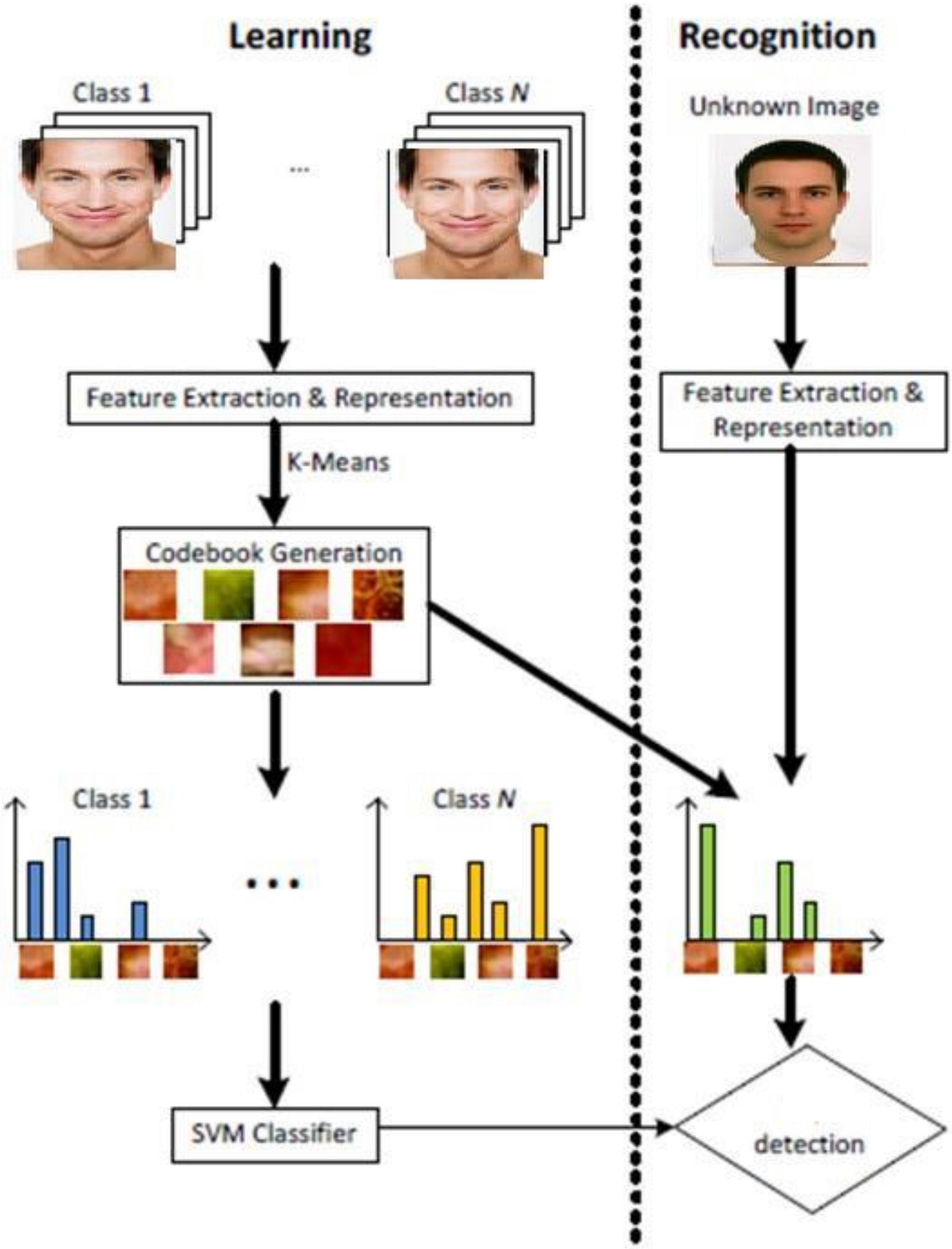


Fig 5.1 Steps in bag of words

The system can be divided mainly into 3 steps

Feature Detection and interest points

Feature Description and clustering

Codebook generation and classification

Feature Detection

Fig 5.2 Feature Detection

In Bag of Words model the images features are extracted and local patches from the images are found and stored.

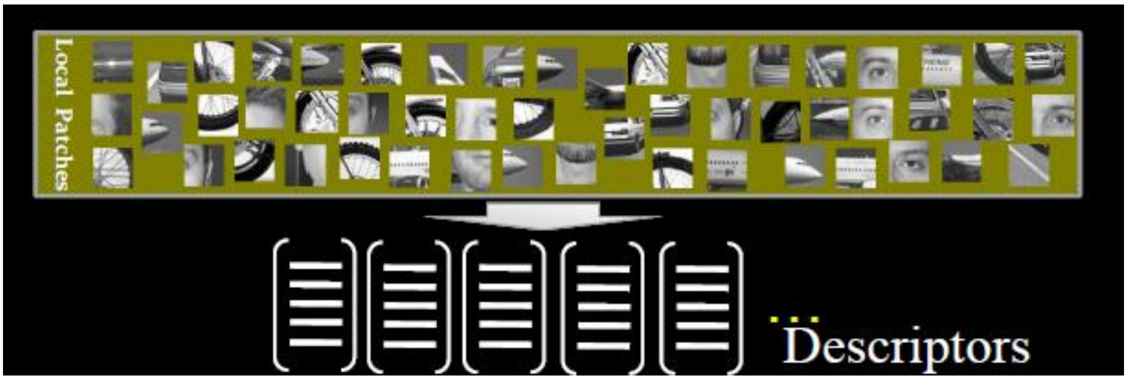


Fig. 5.3 Feature Descriptors

Next local point descriptors are found for that model and description.

Feature Representation and Clustering

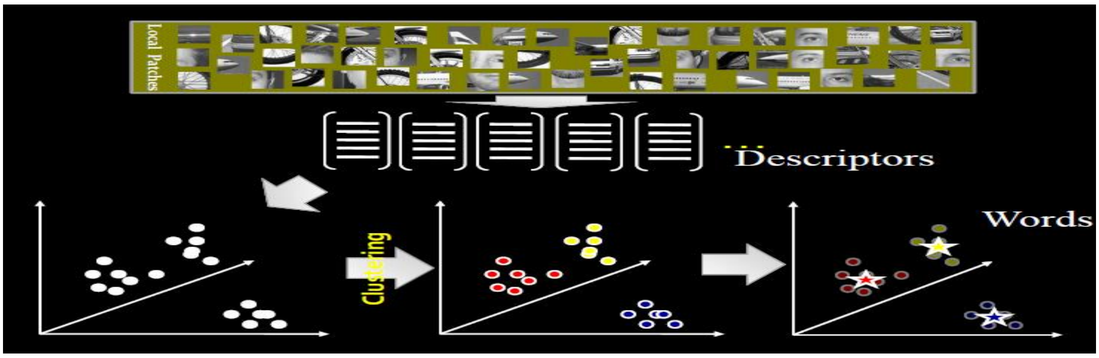


Fig. 5.4 Feature representation and clustering

The extracted features are represented and clustering is done to find the visual words.

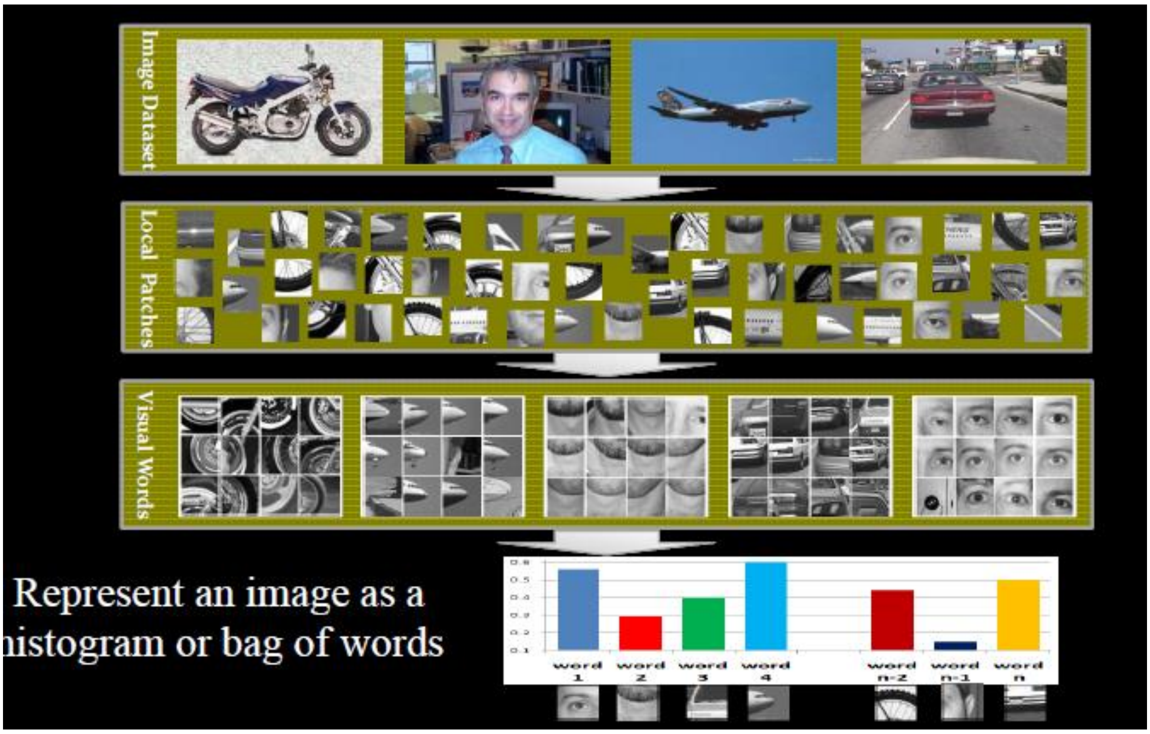


Fig.5.5 Bag of words model as whole

These visual words are represented as histograms or bag of words.

Steps

Step 1: Interest points

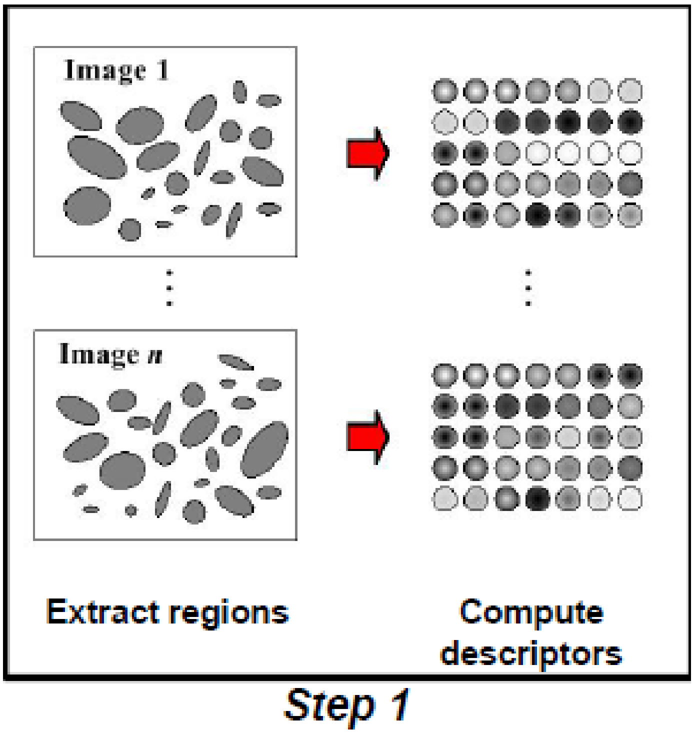


Fig.5.6 Interest points

Step 2: Finding Clusters

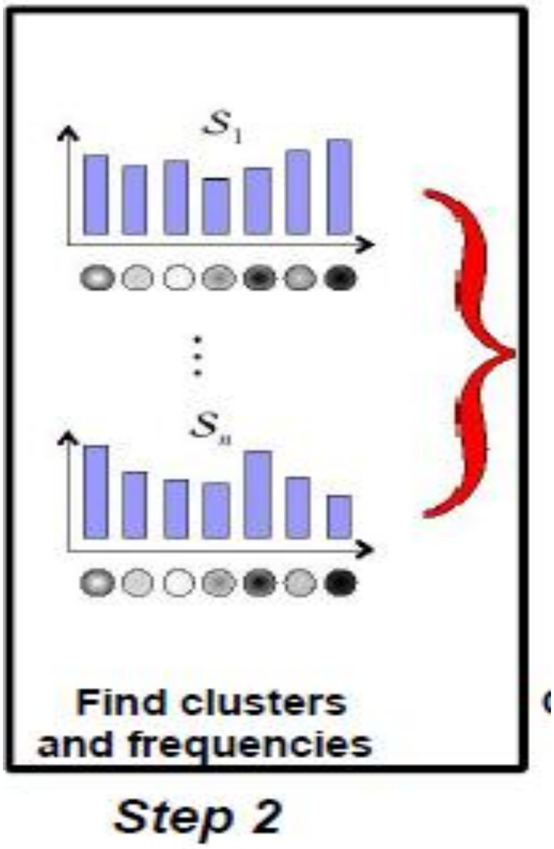


Fig. 5.7 Finding clusters

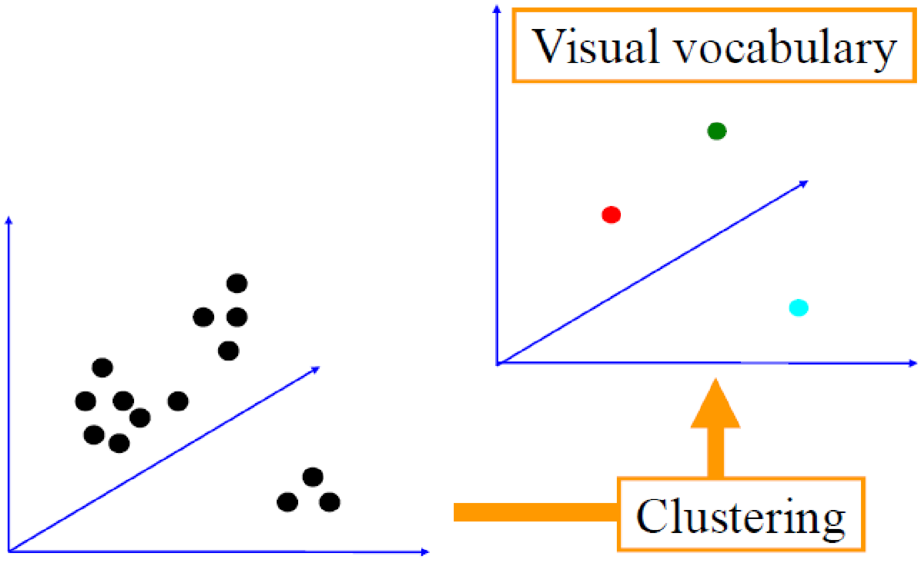


Fig.5.8 Representation of words

Step 3:

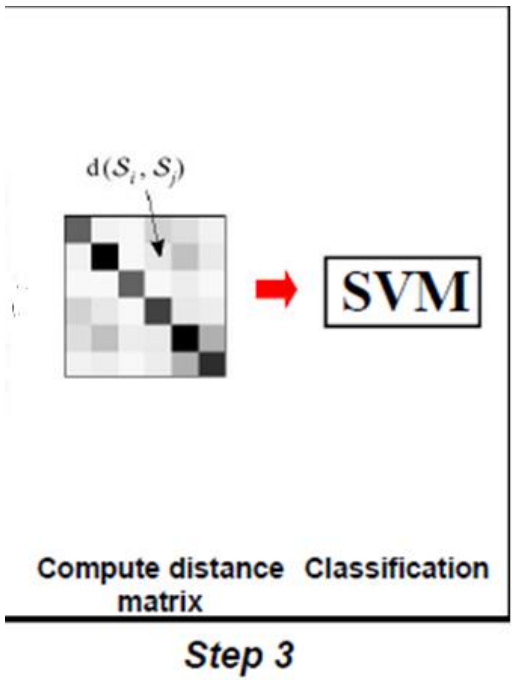


Fig. 5.9 Classification

Fig 5.10 Codewords representation

CHAPTER 6

Implementation

Implementation of any software is always preceded by important decisions regarding selection of the platform, the language to implement, and other factors. These decisions are often influenced by several factors such as the real environment in which the system will work, the speed or response time that is required, the security concerns, and other implementation specific details. This chapter gives overview of the algorithms and methods used.

6.1 Programming Language

C++ has imperative, object-oriented and generic programming features, while also providing the facilities for low level memory manipulation. It is designed with a bias for systems programming (e.g. embedded systems, operating system kernels), with performance, efficiency and flexibility of use as its design requirements. C++ has also been found useful in many other contexts, including desktop applications, servers (e.g. e-commerce, web search, SQL), performance critical applications (e.g. telephone switches, space probes) and entertainment software, such as video games

6.2 Selection of Platform

The application is developed using OpenCV on Microsoft Visual Studio 2013 in windows 8.1 operating system, needed libraries are imported and integrated with the software and linked during the compilation, GUI is also built using the visual studio and the code can run on any platform which is independent.

6.2.1 OpenCV

OpenCV is an open source C++ library for image processing and computer vision, originally developed by Intel and now supported by Willow Garage. It is free for both commercial and non-commercial use. Therefore it is not mandatory for your OpenCV applications to be open or free. It is a library of many inbuilt functions mainly aimed at real time image processing. Now it has several hundreds of image processing and computer vision algorithms which make developing advanced computer vision applications easy and efficient

Key Features

- Optimized for real time image processing & computer vision applications.
- Primary interface of OpenCV is in C++

- There are also C, Python and JAVA full interfaces
- OpenCV applications run on Windows, Android, Linux, Mac and iOS
- Optimized for Intel processors

OpenCV Modules

OpenCV has a modular structure. The main modules of OpenCV are listed below. I have provided some links which are pointing to some example lessons under each module.

Core: This is the basic module of OpenCV. It includes basic data structures (e.g.- **Mat** data structure) and basic image processing functions. This module is also extensively used by other modules like highgui, etc.

Highgui: This module provides simple user interface capabilities, several image and video codecs, image and video capturing capabilities, manipulating image windows, handling track bars and mouse events and etc. If you want more advanced UI capabilities, you have to use UI frameworks like Qt, WinForms, etc. e.g. - Load & Display Image, Capture Video from File or Camera, Write Image & Video to File

Imgproc: This module includes basic image processing algorithms including image filtering, image transformations, color space conversions and etc.

Video: This is a video analysis module which includes object tracking algorithms, background subtraction algorithms and etc.

Objdetect: This includes object detection and recognition algorithms for standard objects.

6.2.2 Microsoft Visual Studio 2013

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs for Microsoft Windows, as well as web sites, web applications and web services. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.Visual Studio includes a code editor supporting IntelliSense (the code completion component) as well as code refactoring. The integrated debugger works both as a source-level debugger and a machine-level debugger. Other built-in tools include a forms designer for building GUI applications, web designer, class designer, and database schema designer. It accepts plug-ins that enhance the functionality at almost every level—including adding support for source-control systems (like Subversion) and adding new toolsets like editors and visual designers for domain-specific languages or toolsets for other aspects of the software development lifecycle.

6.3 Algorithms

In this project we have used different algorithms

1. SIFT: Scale Invariant Feature Transform
2. SURF: Speeded up Robust Feature.
3. BOW: Bag of Words and its variants.

Overview of SIFT

The SIFT algorithm was implemented from the ground up using C++ and Visual Studio.net 2003. The two SIFT papers (Lowe 1999, Lowe 2004) were used as a reference for the structuring the algorithm. Here is the Pseudo-code of the SIFT method (Table1), where first we need to load the and then resampling is done by preprocessing and then we will process the detector mechanism of SIFT algorithm once it come out of the loop key point is evaluated and then we find the key point out of the evaluated key point (given by detector). The final result of the descriptor is saved and this key point is helpful in finding the exact match for the respective image.

Table 6.1 Steps in SIFT

(1) Parse options
(2) Load image
(3) Resample of face from an image to double size
(4) for each octave
(5) Create Gaussian blur intervals
(6) Create difference-of-Gaussian intervals
(7) Compute edges for each interval
(--) end for
(8) Search each octave for stable extrema
(9) Create keypoints at dominant orientations of extrema
(10) for each keypoint
(11) Rotate sample grid to keypoint orientation
(12) Sample region and create descriptor

(--) end for
(13) Optionally save pyramid images
(14) Save output images
(15) Save descriptors

Overview of SURF:

In 2006, three people, Bay, H., Tuytelaars, T. and Van Gool, L, published a paper called "SURF: Speeded Up Robust Features". In that they introduced a new algorithm called SURF (Speeded Up Robust Features). As name proposes, it is a speeded-up form of SIFT. In SIFT, Lowe approximated Laplacian of Gaussian with Difference of Gaussian for finding scale space. SURF approximates LoG with Box Filter as shown in the image below, a demonstration of an approximation. Advantage of this guesstimate is that, complexity with box filter can be easily designed with the help of integral images. And it can be done in equivalent to different scales. Also the SURF bank on element of Hessian matrix for both scale and location. Here is the Pseudo-code of the SURF (Table 2), which is similar to that of the SIFT the only difference is Box Filter and Haar wavelet filter of the SURF algorithm.

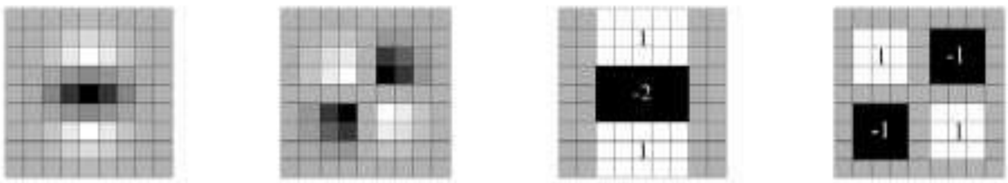


Fig 6.1 Box filter used in surf

Table 6.2 Steps in SURF

(1) Parse options
(2) Load image
(3) Resample of face from an image to double size
(4) for each octave
(5) Create Gaussian blur intervals
(6) Apply Box Filter for approximation.
(--) end for
(7) Search each octave for stable extrema
(8) for each keypoint

(9) Assign the orientation for reach key point using Haar wavelet filter
(10) Calculate weighted by the Gaussian function for each keypoint.
(11) Assign the calculated direction vector of the appropriate key point
(--) end for
(12) Save output images
(13) Save descriptors

Overview of BOW

Bow is one of the popular visual descriptors used for visual data classification. Bow is inspired by a concept called Bag of Words that is used in document classification. A bag of words is a sparse vector of occurrence counts of words; that is, a sparse histogram over the vocabulary. In computer vision, a bag of visual words of features is a sparse vector of occurrence counts of a vocabulary of local image features. Bow typically involves in two main steps. First step is obtaining the set of bags of features. This step is actually offline process. We can obtain set of bags for particular features and then use them for creating Bow descriptor. The second step is we cluster the set of given features into the set of bags that we created in first step and then create the histogram taking the bags as the bins. This histogram can be used to classify the image or video frame.

Bag-of Words with SIFT

Let's see how can we build Bow with SIFT features.

Table6.3 Steps in BOW SIFT

1. Obtain the set of bags of features.

    i. Select a large set of images.
    ii. Extract the SIFT feature points of all the images in the set and obtain the SIFT descriptor for each feature point that is extracted from each image.
    iii. Cluster the set of feature descriptors for the amount of bags we defined and train the bags with clustered feature descriptors (we can use the K-Means algorithm).
    iv. Obtain the visual vocabulary.

2. Obtain the Bow descriptor for given image/video frame.

v. Extract SIFT feature points of the given image. vi. Obtain SIFT descriptor for each feature point. vii. Match the feature descriptors with the vocabulary we created in the first step viii. Build the histogram.

Bag-of Words with SURF

Let's see how can we build Bow with SURF features.

Fig Table 6.4 Steps in BOW with SURF

1. Obtain the set of bags of features. ix. Select a large set of images. x. Extract the SURF feature points of all the images in the set and obtain the SURF descriptor for each feature point that is extracted from each image. xi. Cluster the set of feature descriptors for the amount of bags we defined and train the bags with clustered feature descriptors (we can use the K-Means algorithm). xii. Obtain the visual vocabulary. 2. Obtain the Bow descriptor for given image/video frame. xiii. Extract SURF feature points of the given image. xiv. Obtain SURF descriptor for each feature point. xv. Match the feature descriptors with the vocabulary we created in the first step xvi. Build the histogram.

CHAPTER 7

EXPERIMENT RESULTS ON DATASET

7.1 Dataset Detail

BioID Face Dataset : The BioID Face Database has been recorded and is published to give all researchers working in the area of face detection the possibility to compare the quality of their face detection algorithms with others. It may be used for such purposes without further permission. During the recording special emphasis has been placed on "real world" conditions. Therefore the testset features a large variety of illumination, background, and face size. Some typical sample images are shown below.

Fig. 7.1 Sample images in Dataset

Description of the face database

The dataset consists of 1521 gray level images with a resolution of 384x286 pixel. Each one shows the frontal view of a face of one out of 23 different test persons. For comparison reasons the set also contains manually set eye postions. The images are labeled "BioID_xxxx.pgm" where the characters xxxx are replaced by the index of the current image (with leading zeros). Similar to this, the files "BioID_xxxx.eye" contain the eye positions for the corresponding images.

Image file format

The images are stored in single files using the portable gray map (pgm) data format. A pgm file contains a data header followed by the image data. In our case the header consists of four lines of text. In detail:

- The first line describes the format of the image data (ASCII/binary). In our files the text "P5" indicates that the data is written in binary form
- the second line contains the image width written in text form
- the third line keeps the image height also in text form
- the fourth line contains the maximum allowed gray value (255 in our images)

FACE95 Dataset:

Acquisition conditions

Using a fixed camera, a sequence of 20 images per individual was taken. During the sequence the subject takes one step forward towards the camera. This movement is used to introduce significant head (scale) variations between images of same individual. There is about 0.5 seconds between successive frames in the sequence.

Database Description

Number of individuals: 72

Image resolution: 180 by 200 pixels (portrait format)

Contains images of male and female subjects

Variation of individual's images

 Backgrounds: the background consists of a red curtain.

 Head Scale: large head scale variation

 Head turn, tilt and slant: minor variation in these attributes

 Image lighting variation: as subject moves forward, significant lighting changes occur on faces due to the artificial lighting arrangement

 Expression Variation: some expression variation

Fig 7.2 Sample images in Dataset

7.2 Experimental Results

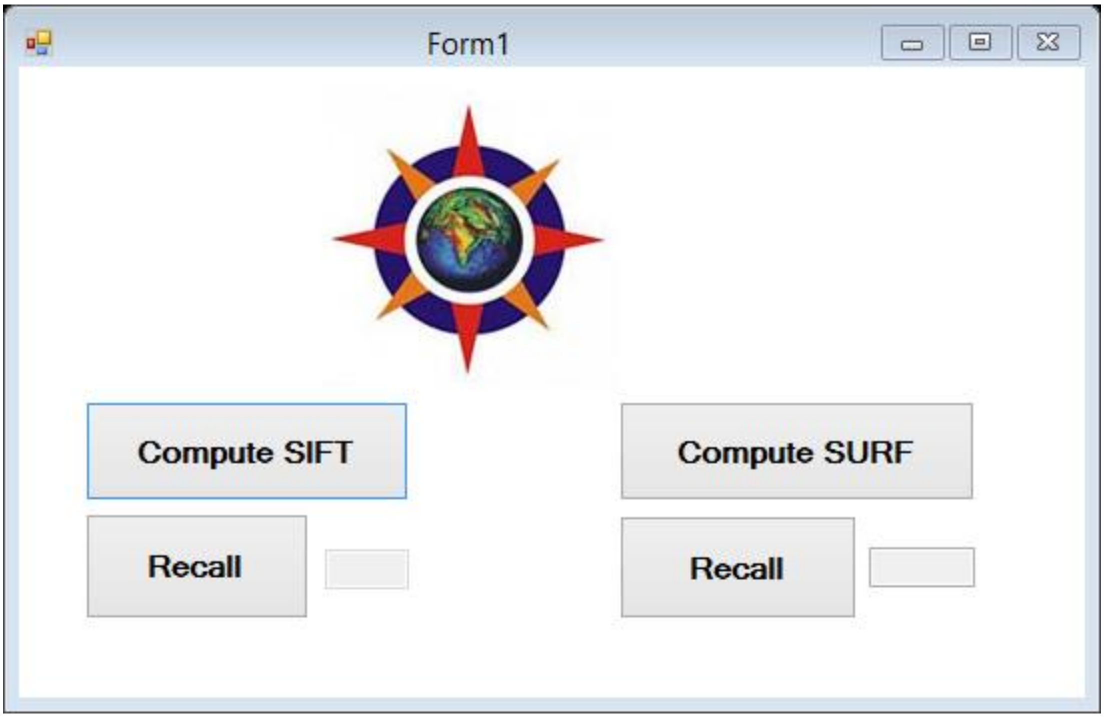


Fig. 7.3 GUI Interface

Fig. 7.4 Face Matches

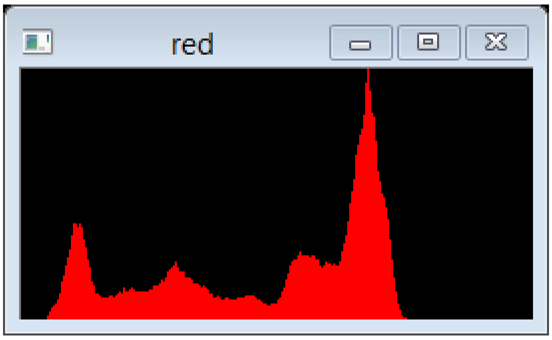


Fig. 7.5 Keypoints Graph_1

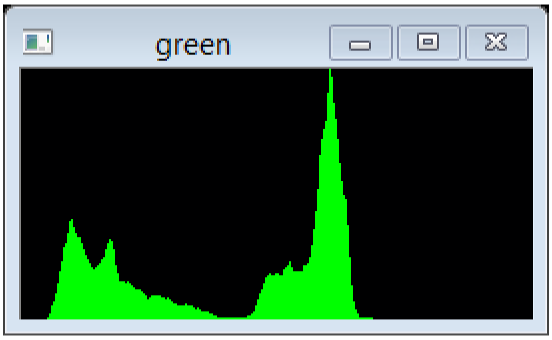


Fig. 7.6 Keypoints Graph_2

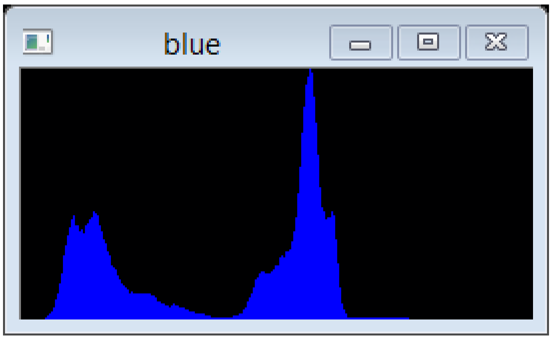


Fig. 7.7 Keypoints Graph_3

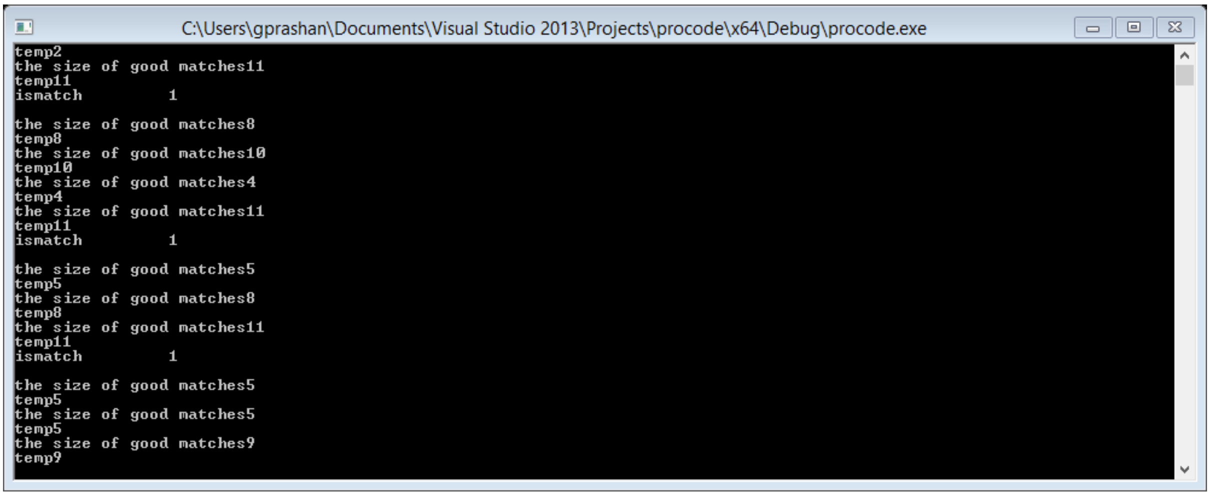


Fig. 7.8 Good Matches Result Computation_1

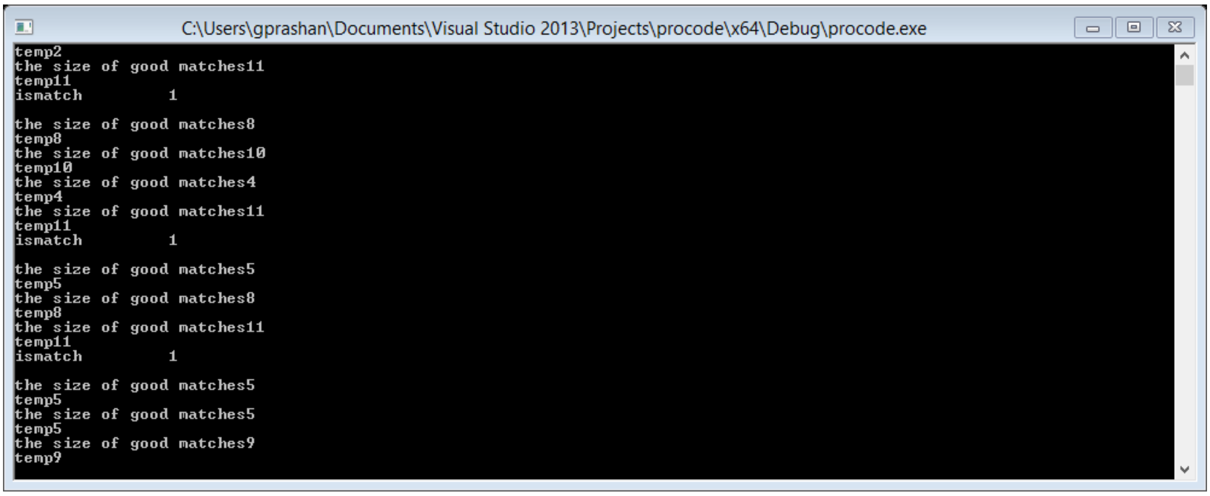


Fig. 7.9 Good Matches Result Computation_2

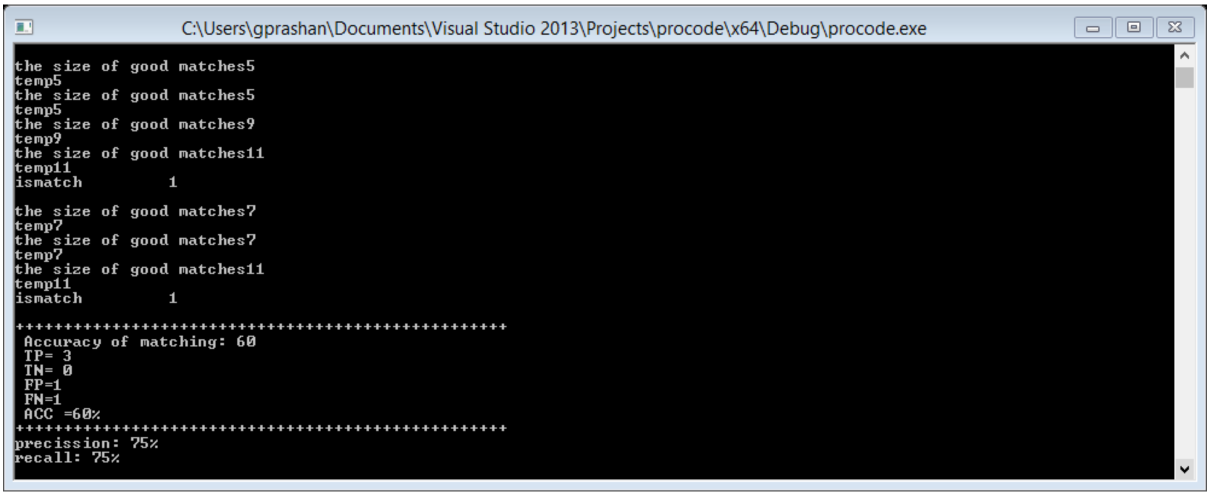


Fig. 7.10 Accuracy Results_1

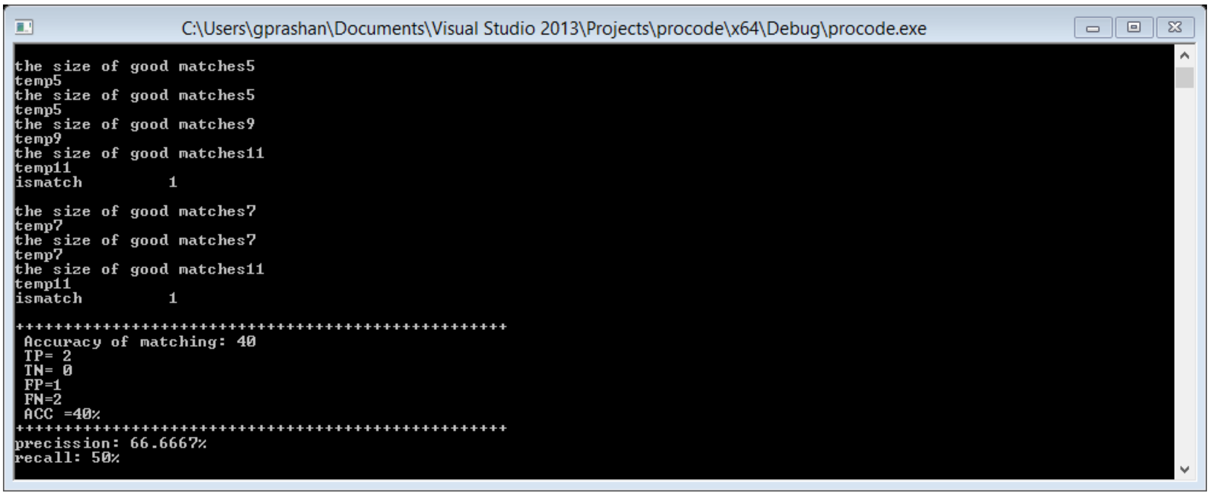


Fig. 7.11 Accuracy Results_2

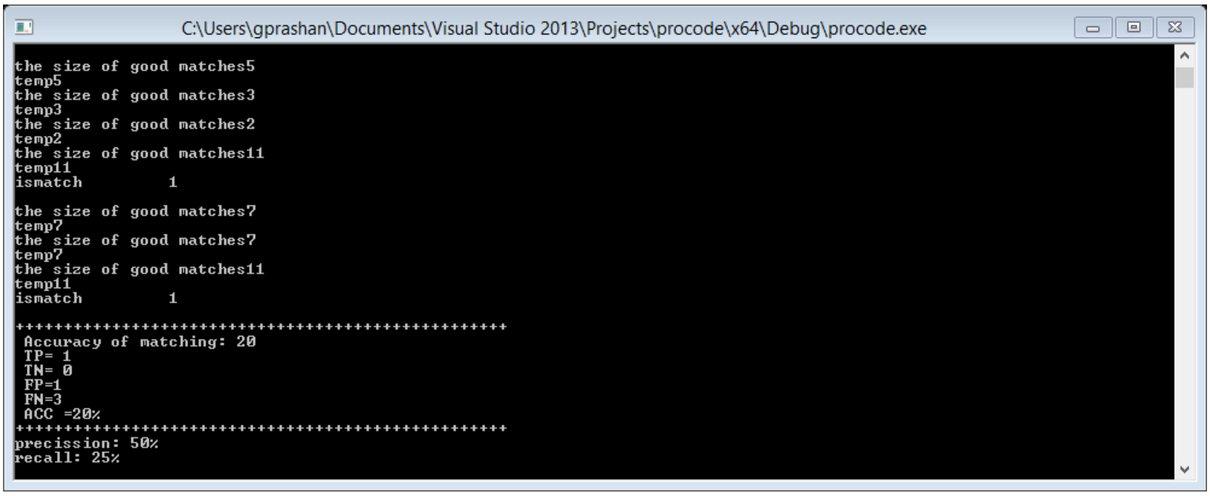


Fig. 7.12 Accuracy Results_3

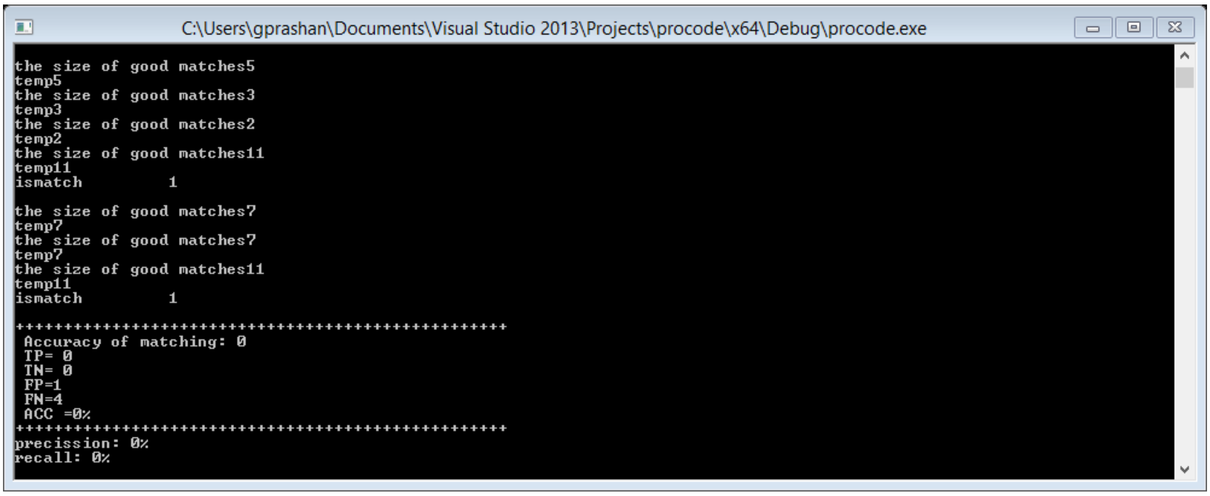


Fig. 7.13 Accuracy Results_4

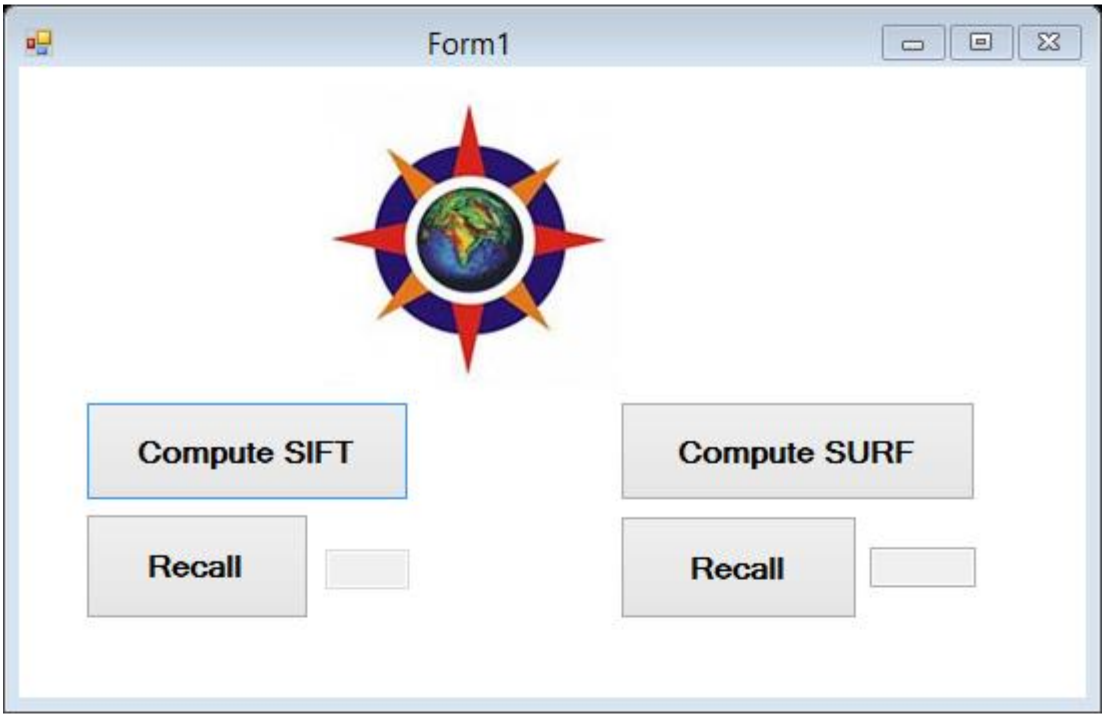


Fig. 7.14 GUI Interface for SURF

Fig. 7.15 Face matches

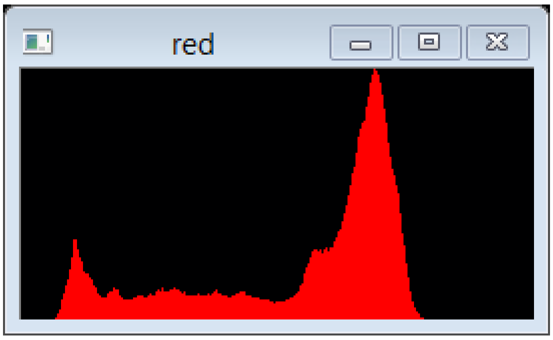


Fig. 7.16 Keypoinys graph_1

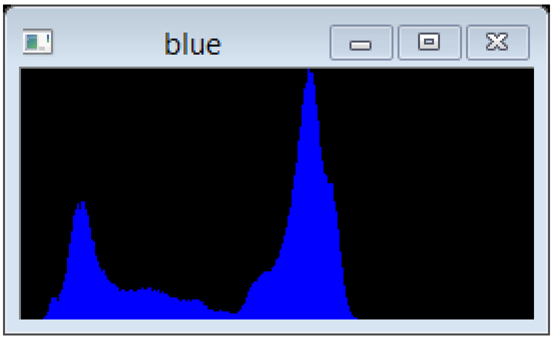


Fig. 7.17 Keypoinys graph_2

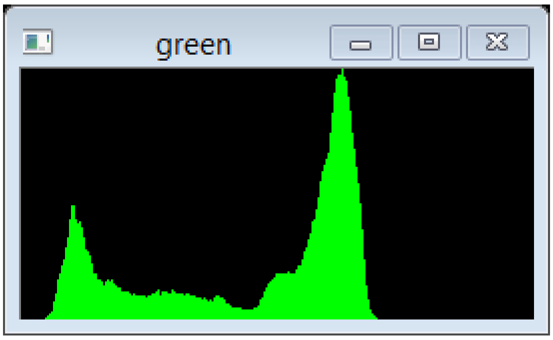


Fig. 7.18 Keypoinys graph_3

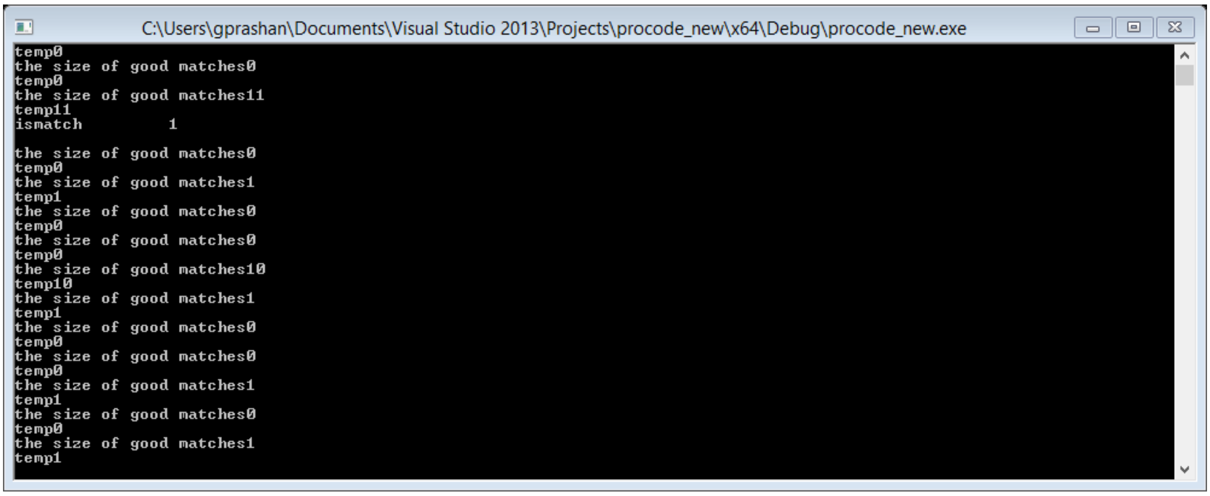


Fig. 7.19 Good matches results computation_1

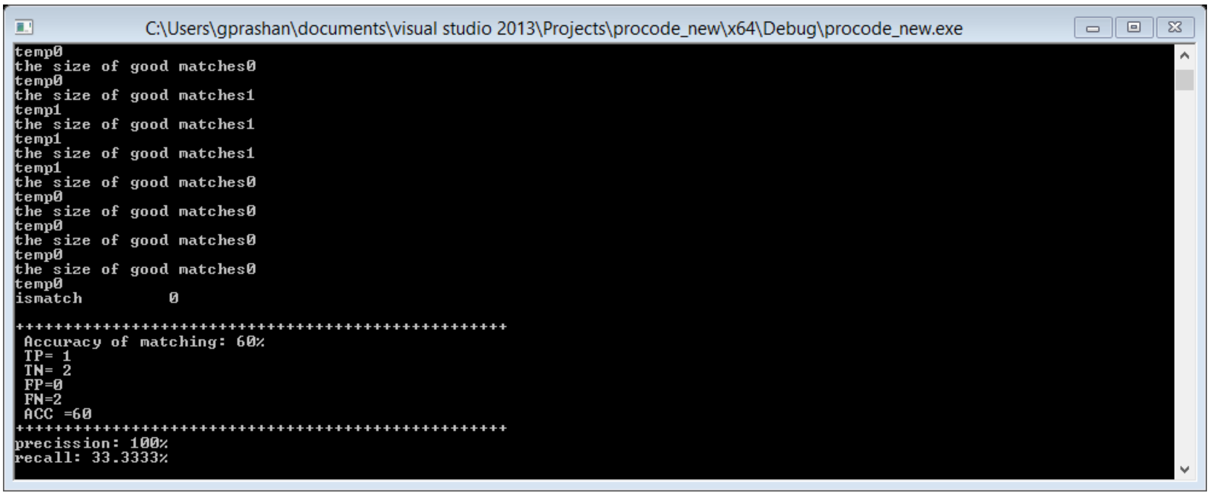


Fig. 7.20 Accuracy results_1

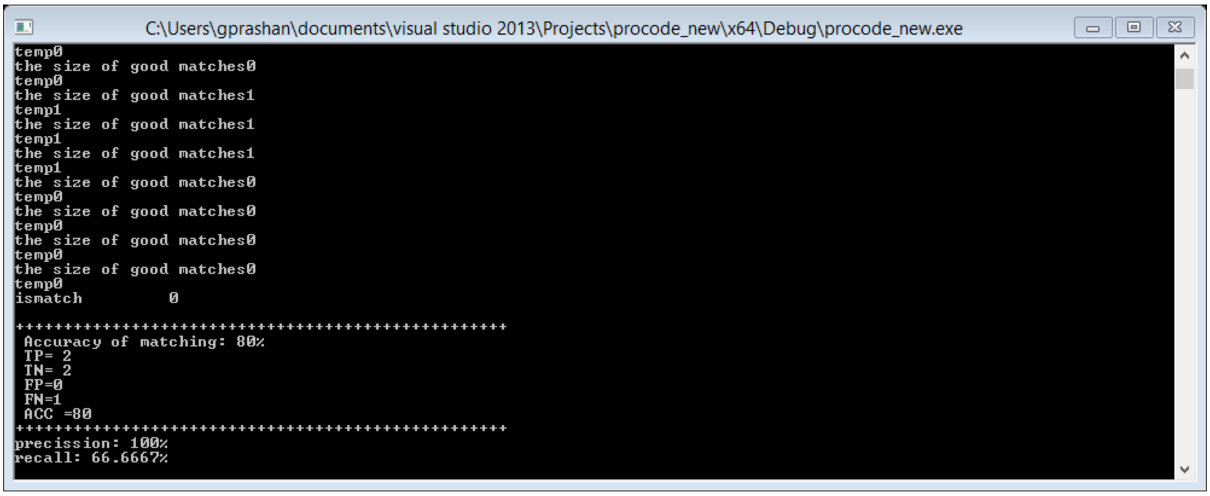


Fig. 7.21 Accuracy results_2

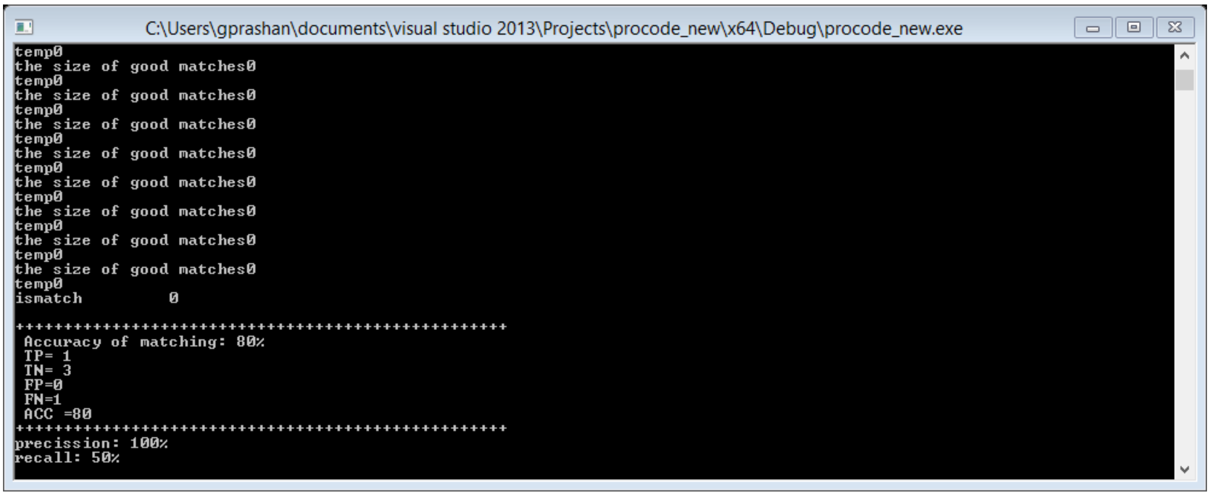


Fig 7.22 Accuracy results_3

7.3 Model Evaluation

7.3.1 Terms

On running the Classification Model for 104 images, the following Evaluation Matrix is laid.

Those are

- True Positive
- True Negative
- False Positive
- False negative
- Accuracy

- Error rate
- Sensitivity
- Specificity
- Precision
- Recall
- F1 Measure

True positives (TP): These refer to the positive tuples that were correctly labelled by the classifier. In this case the number of correct images that are correctly matched.

True negatives (TN): These are the negative tuples that were correctly labelled by the classifier. That is number of incorrect images that are matched.

False positives (FP): These are the negative tuples that were incorrectly labelled as positive. That is number of images that are wrongly matched.

False negatives (FN): These are the positive tuples that were mislabeled as negative. That is number of images that are wrongly matched.

Accuracy: The accuracy of a classifier on a given test set is the percentage of test set tuples that are correctly classified by the classifier. It is evaluated as follows,

Accuracy $= TP + TN / P + N$

Error rate or misclassification rate of a classifier, M, which
is simply 1 – accuracy(M), where accuracy(M) is the accuracy of M.

Error rate $= FP + FN / P + N$

Sensitivity is also referred to as the true positive (recognition) rate (i.e., the proportion of positive tuples that are correctly identified). Here it is proportion of correctly classified Male

Sensitivity $= TP / P$

Specificity is the true negative rate (i.e., the proportion of negative tuples that are correctly identified). That is proportion of correctly classified Female

Specificity $= TN/N$

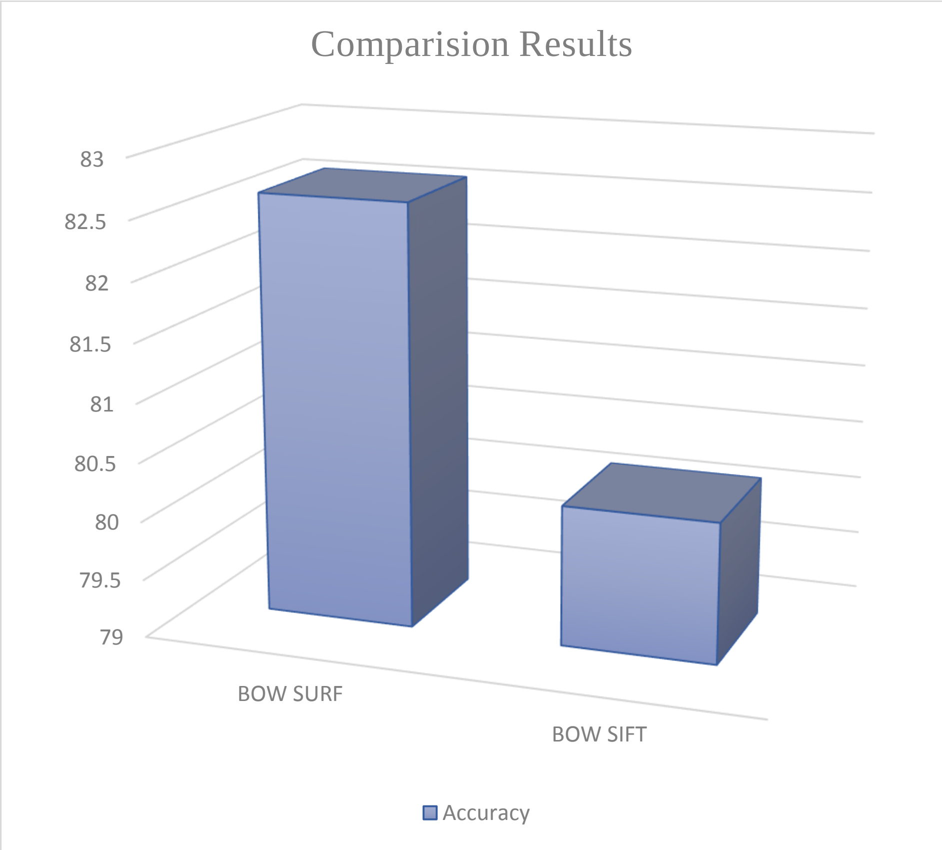
Precision can be thought of as a measure of exactness (i.e., what percentage of tuples labeled as positive are actually such)

Precision $= TP/TP+FP$

Recall is a measure of completeness (what percentage of positive tuples are labeled as such)

Recall $= TP/TP+FN$

Results


Comparision Results
83
82.5
82
81.5
81
80.5
80
79.5
79
BOW SURF
BOW SIFT
Accuracy

CHAPTER 8

CONCLUSION AND SCOPE OF THE WORK

In this project, an attempt has been made to identify person automatically. In this, we classify the images on the basis of features. When classifying the image, first face features are extracted using SURF and SIFT, once the face features are extracted ,once the features are extracted then to apply the bag of words model for representation. The proposed system has a low complexity and is suitable for real time implementations.

However, the proposed model has some limitations. Accurately identifying the correct person through histogram representation and codebook generation. Other possibility of improvement in the proposed model is to handle facial images if some part of face is covered. The proposed model can also be extended to non-frontal facial images. Improvement in accuracy of feature extraction algorithm can also further improve the performance of proposed system.

REFERENCES

1 Cong Geng and Xudong Jiang, " FACE RECOGNITION USING SIFT FEATURES ", School of Electrical and Electronic Engineering ,Nanyang Technological University, Singapore *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 978-1-4244-5654, Oct 2009.

2. M. Bicego, A. Lagorio, E. Grosso, and M. Tistarelli, “On the use of sift features for face authentication,” Computer Vision and Pattern Recognition Workshop, 2006 Conference on, pp. 35–35, June 2006.

3. D.R. Kisku, A. Rattani, E. Grosso, and M. Tistarelli, “Face identification by sift-based complete graph topology,” Automatic Identification Advanced Technologies, 2007 IEEE Workshop on, pp. 63–68, June 2007.

4. E. Mairm G. D. Hager, D. Burschka, M. Suppa, and G. Hirzinger. “Adaptive and generic corner detection based on the accelerated segment test”,In Proceedings of the European Conference on Computer Vision (ECCV), 2010,2-5.

5. Xin Yang and Kwang-Ting (Tim) Cheng “SURF Detector on Mobile Devices”
In IEEE Trans. PAMI, 978-1-4503-1089-5, 2012.

6. Haiyan Li, Tingrong Xu, Jie Li and LiXiao Zhang, " Face Recognition based on Improved SURF ", School of computer science and technology, Soochow University, China, IEEE Transactions on Pattern Analysis and Machine Intelligence, 978-0-7695-4923-1, 2012.

7. Hamdi Jamel Bouchecha, Sebti Foufoua and Mongi Abidi, " Strengthening Surf Descriptor With Discriminant Image Filter Learning: Application To Face Recognition ", School of computer science and technology, Qatar University, US, IEEE Transactions on Pattern Analysis and Machine Intelligence, 978-1-4799-8153-3,2014.

8. Zhang Huijuan and Hu Qiong, “Fast Image Matching Based-on Improved SURF Algorithm", School of Software Engineering, Tongji University, China, IEEE Transactions on Pattern Analysis and Machine Intelligence, 978-1-4577-0321-8, 2011.

9. Charles Bibby and Ian Reid, “Fast Feature Detection with a Graphics Processing Unit Implementation", Department of Engineering Science, University of Oxford, Oxford, UK.
Evaluating the BOSS Descriptors for Face Recognition
Dept. of ISE, PESIT 2014-15 Page 101

10. Stefan Leutenegger, Margarita Chli and Roland Y. Siegwart, “BRISK: Binary Robust Invariant Scalable Keypoints", Autonomous Systems Lab, ETH Z¨urich, IEEE International

Conference on Computer Vision 978-1-4577-1102-2, 2011.

11. Xiao Tengjiao, Zhao Danpei and Shi Jun, Lu Ming, “High-speed Recognition Algorithm Based on BRISK and Saliency Detection for Aerial Images", Beihang University, Beijing, China, IEEE International Conference on Computer Vision 978-0-7695-4923-1, 2012.

12. Janez Kri_zaj, Vitomir _Struc, and France Miheli_c, “A Feasibility Study on the use of BinaryKeypoint Descriptors for 3D Face Recognition", Faculty of Electrical Engineering, University of Ljubljana, Slovenia.

13. Dagmawi Bekele, Michael Teutsch and Tobias Schuchert, “Evaluation of Keypoint Descriptors", Karlsruhe Institute of Technology, Karlsruhe, Germany, IEEE International Conference on Computer Vision 978-1-4799-2341-0, 2013.

14. H. Bay, A. Ess, T. Tuytelaars and L.Van Gool. SURF: Speeded Up Robust Features. *Computer Vision and Image Understanding*, vol.110, no.3, pp.346-359, 2008.

15 D.G. Lowe. Distinctive Image Features from Scale- Invariant Keypoints. *International Journal on Computer Vision*, vol.2, no.60, pp.91-110, 2004.

16 V. Vapnik, Statistical learning theory, *Wiley*, 1998.

17 CJC. Burges, A tutorial on support vector machines for pattern recognition, *Data Mining Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998.

18 SS. Keerthi and C-J. Lin, Asymptotic behaviors of support vector machines with Gaussian kernel, *Neural Computation*, vol. 15, no. 7, pp. 1667–1689, 2003.

19 D.A.R. Vigo, F.S. Khan, J. van de Weijer, and T. Gevers, The Impact of Color on Bag-of-Words based Object Recognition., *International Conference on Pattern Recognition (ICPR)*, pp. 1549 - 1553 August. 2010