# Design Method for Recognition of Filled Bubbles in OMR Sheets and Grading it using Matlab

*Project Report submitted in partial fulfilment of the requirements for the Degree of*

**BACHELOR OF TECHNOLOGY**

**IN**

**ELECTRONICS AND COMMUNICATION ENGINEERING**

**By**

**Ayush Pandey (121048)**

**Aatish Mahajan (121049)**

**Yadvir Sikarwar (121060)**

**UNDER THE GUIDANCE OF**

**Dr. Pradeep Kumar Chauhan**

**JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT**
**SOLAN – 173 234, HIMACHAL PRADESH, INDIA**
**June- 2016**

## TABLE OF CONTENTS

# DECLARATION BY THE SCHOLAR

We hereby declare that the work reported in the B-Tech. project entitled **"Design Method for Recognition of Filled Bubbles in OMR Sheets and Grading it using Matlab"** submitted at **Jaypee University of Information Technology, Waknaghat, Solan, H.P.** is an authentic record of our work carried out under the supervision of **Dr. Pradeep Kumar Chauhan**. We have not submitted this work elsewhere for any other degree or diploma.

Ayush   Aatish

Ayush Pandey   Aatish Mahajan   Yadvir Sikarwar

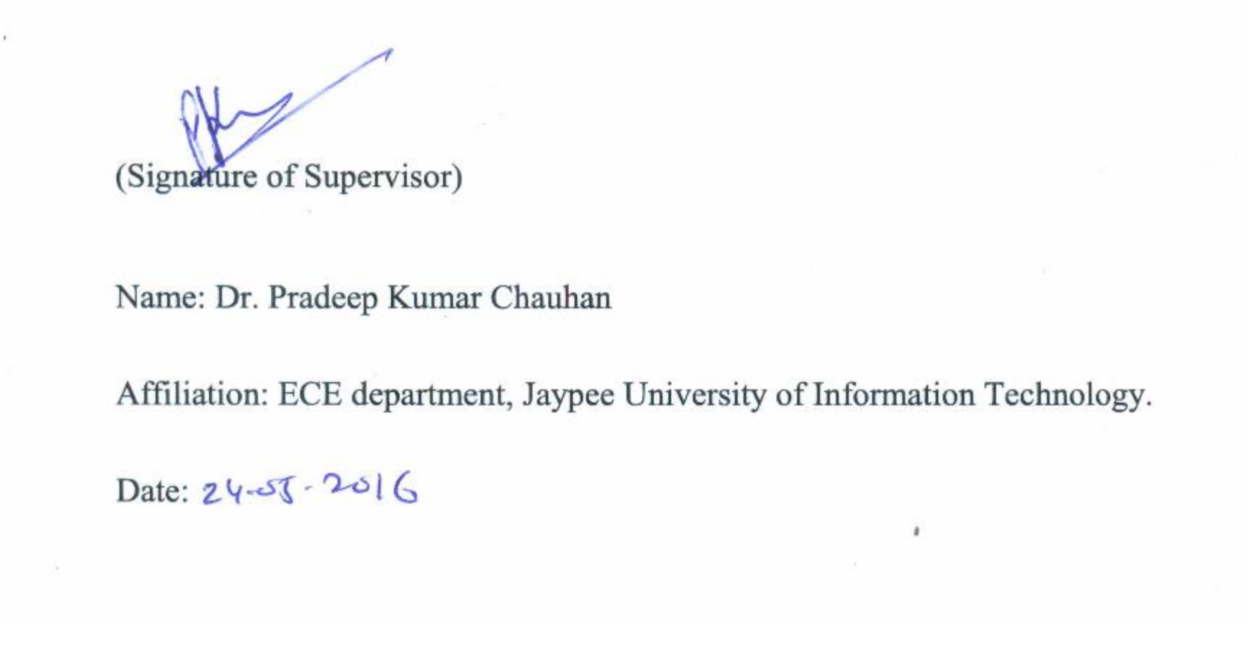
Department of Electronics and Communication

Jaypee University of Information Technology, Waknaghat, Solan, H.P.

Date: 24-05-2016

# SUPERVISOR'S CERTIFICATE

This is to certify that the work reported in the B-Tech. project entitled **"Design Method for Recognition of Filled Bubbles in OMR Sheets and Grading it using Matlab"**, submitted by **Ayush Pandey, Aatish Mahajan, Yadvir Sikarwar** at **Jaypee University of Information Technology, Waknaghat, Solan, H.P.** is a bonafide record of his / her original work carried out under my supervision. This work has not been submitted elsewhere for any other degree or diploma.



(Signature of Supervisor)

Name: Dr. Pradeep Kumar Chauhan

Affiliation: ECE department, Jaypee University of Information Technology.

Date: 24-05-2016

# ACKNOWLEDGEMENT

It is our foremost duty to express our deep regards & gratitude to our Project Guide **Dr. Pradeep Kumar Chauhan,** under whose guidance & supervision we are able to work on the project ***"Design Method for Recognition of Filled Bubbles in OMR Sheets and Grading it using Matlab".*** Any amount of gratitude towards our family and friends can never be enough for the constant support they have provided us with, we are totally indebted to them. At last it is him, to whom we owe everything, as without God's blessings this endeavours of ours would never have been a success.

# LIST OF ACRONYMS & ABBREVIATIONS

1. OMR-Optical Mark Recognition
2. MCQ-Multiple Choice Question
3. OCR- Optical Character Recognition
4. ICR -Intelligent Character Recognition
5. NCS-National Computer Systems
6. BHT- Balanced histogram thresholding method

# LIST OF FIGURES

# LIST OF TABLES

# ABSTRACT

We explore an image-processing algorithm for auto grading of answer sheets. Big organizations use OMR forms coupled with OMR software and dedicated scanner for grading multiple-choice questions. But small institutes and individual teachers cannot afford this costly setup and have to do time consuming manual grading. So we propose an application that can be used by anyone at any time to grade answer sheets and save the valuable time spent grading manually. Every year millions of students take standardized tests and they have to answer various questions asked by darkening bubbles in OMR sheets. Current solutions for evaluating these OMR sheets are expensive and need dedicated scanner, OMR software and buying customized OMR sheets. The motivation of our project comes from the idea that we could build application that will assist the instructors in auto grading these answer sheets and saving their valuable instruction time. And it is very easy to use solution without lot of steps or setup.

CHAPTER 1

# INTRODUCTION

**Optical Mark Recognition** sheets also known as OMR sheets use different form of paper that has index points on its four corners. OMR sheets are used in the examination where numbers of candidates are large and manual checking is waste of time and resources. OMR sheets are scanned and results are displayed within no time. There is no need to check each and every answer. Apart from answer sheets, OMR forms are also available. These forms are used for collecting data and are available in the form of application forms, admission forms, ballot papers, attendance sheets, questionnaires, etc.

Optical mark recognition (also called optical mark reading and OMR) is the process of capturing human-marked data from document forms such as surveys and tests.

Our algorithm is mainly divided into two parts. The first part is to apply image registration techniques to align a taken image with the template image and compute the transformation matrix. The second part is to apply the transformation matrix to center pixels of all answer circles and count the number of black pixels around the region to figure out whether the circle is filled or not. We envisage that our application would be able to do the following

i. Identify the answer in the image

ii. Detect response/s for each question and

iii. Compare student responses with reference answers. We have designed our algorithm assuming that we know the position of circles corresponding to each question.

## 1.1 Types of OMR Sheets:

The classification depends on the purpose of the examination.

**1.1.1 Multi-choice questions (MCQ test sheets):** These sheets are used by schools, colleges, universities and examination centres. They may have different types of blocks, some of which include alphabetic block, numerical (and integer) block, column matching block, column matrix, ABCD block, etc.

**1.1.2 Collecting Data:** These are used for admission forms, application forms, feedback forms, questionnaires, etc. Such types of OMR sheets have multiple blocks. Apart from OMR blocks, you will also find OCR (Optical Character Recognition) and ICR (Intelligent Character Recognition) blocks. Such forms can have blocks for photos, handwritten content and barcode too.

**1.2.3 Special purposes:** Used for attendance sheets, ballot sheets, award or mark sheets, etc.The OMR sheets can be categorized as per the design and printing process too.

**1.2.4 Coloured:** Coloured sheets are used when old technology OMR reading machines are used. It requires thick coloured paper. Generally, two colours – Orange or Red is used for the sheet. It has accurate form of checking because marks made by the candidate on the bubbles can be easily detected even if it is very light.

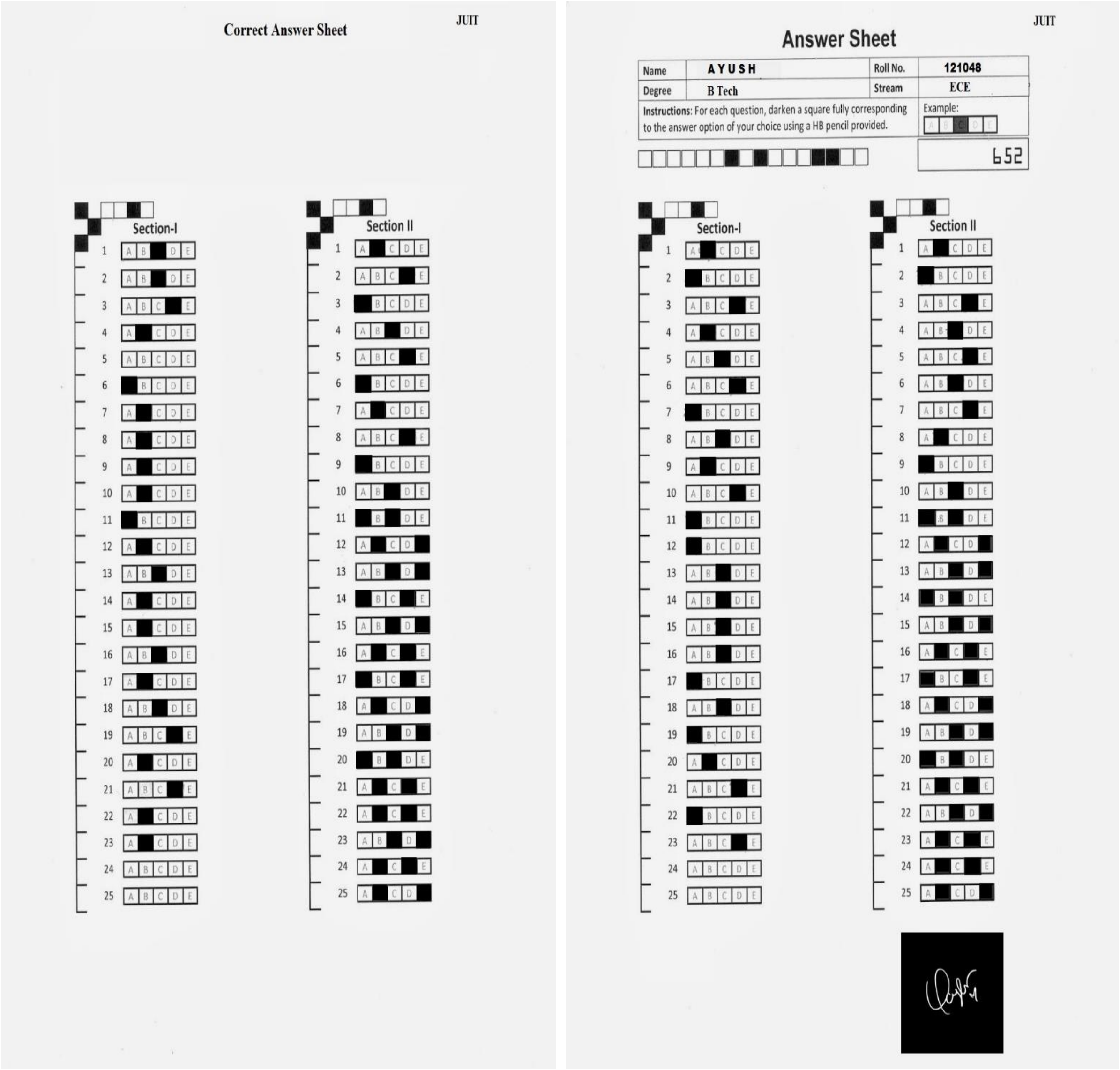
**1.2.5 Black and White:** With the advancement in the technology, it is no longer necessary to use coloured OMR sheets. One can print the black and white OMR sheets on normal A4 sized white paper. Even photocopies of these sheets are used.

**1.2.6 Bar-coded:** The bar-coded OMR sheets are usually used to mask candidate information like unique ID or roll number. This is mostly used to avoid misplacement of sheets or duplication of answers.

**1.2.7 Two-sided:** Two sided OMR sheets contain fields on both sides of the paper. This is used when all the questions cannot be fit into one side of the sheet.

**1.2.8 Other forms:** The other types of OMR sheets include lithocode, foldable, carbonless, detachable, etc. The lithocode OMR sheets are used as a substitute for barcoded sheets, while the folded sheets are used to make sure that the sheets do not get damaged during transportation. Carbonless sheets have two sheets, one is original which has to be submitted at the end of the exam and the other is a copy which can be retrieved by the student. Detachable OMR sheets are also called 2 part OMR sheets and are used to hide the identity of the students from the evaluators in case of confidential tests or exams.

Traditionally, the OMR sheets are printed on thick sheets of paper and possess a timeline in the place of index points. These have to be scanned at a high resolution for accurate results. The best thing is that such sheets can be read by the Addmen OMR software. However, you no longer need to use such sheets for examinations. With the Addmen OMR sheets, designing, printing, scanning and using OMR sheets has become easy and cost effective.



Correct Answer Sheet

JUIT

Section-I

| | | | | | |
|---|---|---|---|---|---|
| 1 | A | B | ■ | D | E |
| 2 | A | B | ■ | D | E |
| 3 | A | B | C | ■ | E |
| 4 | A | ■ | C | D | E |
| 5 | A | B | C | D | E |
| 6 | ■ | B | C | D | E |
| 7 | A | ■ | C | D | E |
| 8 | A | ■ | C | D | E |
| 9 | A | ■ | C | D | E |
| 10 | A | ■ | C | D | E |
| 11 | ■ | B | C | D | E |
| 12 | A | ■ | C | D | E |
| 13 | A | B | ■ | D | E |
| 14 | A | ■ | C | D | E |
| 15 | A | ■ | C | D | E |
| 16 | A | B | ■ | D | E |
| 17 | A | ■ | C | D | E |
| 18 | A | B | ■ | D | E |
| 19 | A | B | C | ■ | E |
| 20 | A | ■ | C | D | E |
| 21 | A | B | C | ■ | E |
| 22 | A | ■ | C | D | E |
| 23 | A | ■ | C | D | E |
| 24 | A | B | C | D | E |
| 25 | A | B | C | D | E |

Section II

| | | | | | |
|---|---|---|---|---|---|
| 1 | A | ■ | C | D | E |
| 2 | A | B | C | ■ | E |
| 3 | ■ | B | C | D | E |
| 4 | A | B | ■ | D | E |
| 5 | A | B | C | ■ | E |
| 6 | ■ | B | C | D | E |
| 7 | A | ■ | C | D | E |
| 8 | A | B | C | ■ | E |
| 9 | ■ | B | C | D | E |
| 10 | A | B | ■ | D | E |
| 11 | ■ | B | ■ | D | E |
| 12 | A | ■ | C | D | ■ |
| 13 | A | B | ■ | D | ■ |
| 14 | ■ | B | C | ■ | E |
| 15 | A | B | ■ | D | ■ |
| 16 | A | ■ | C | ■ | E |
| 17 | ■ | B | C | ■ | E |
| 18 | A | ■ | C | D | ■ |
| 19 | A | B | ■ | D | ■ |
| 20 | ■ | B | ■ | D | E |
| 21 | A | ■ | C | ■ | E |
| 22 | A | ■ | C | ■ | E |
| 23 | A | B | ■ | D | ■ |
| 24 | A | ■ | C | ■ | E |
| 25 | A | ■ | C | D | ■ |

JUIT

Answer Sheet

| | | | |
|---|---|---|---|
| Name | AYUSH | Roll No. | 121048 |
| Degree | B Tech | Stream | ECE |
| Instructions: For each question, darken a square fully corresponding to the answer option of your choice using a HB pencil provided. | | Example: A B ■ D E | |

652

Section-I

| | | | | | |
|---|---|---|---|---|---|
| 1 | A | ■ | C | D | E |
| 2 | ■ | B | C | D | E |
| 3 | A | B | C | ■ | E |
| 4 | A | ■ | C | D | E |
| 5 | A | B | ■ | D | E |
| 6 | A | B | C | ■ | E |
| 7 | ■ | B | C | D | E |
| 8 | A | B | ■ | D | E |
| 9 | A | ■ | C | D | E |
| 10 | A | B | C | ■ | E |
| 11 | ■ | B | C | D | E |
| 12 | ■ | B | C | D | E |
| 13 | A | B | ■ | D | E |
| 14 | A | B | ■ | D | E |
| 15 | A | B | ■ | D | E |
| 16 | A | B | ■ | D | E |
| 17 | ■ | B | C | D | E |
| 18 | A | B | ■ | D | E |
| 19 | ■ | B | C | D | E |
| 20 | A | ■ | C | D | E |
| 21 | A | B | C | ■ | E |
| 22 | ■ | B | C | D | E |
| 23 | A | B | C | ■ | E |
| 24 | A | B | C | D | E |
| 25 | A | B | C | D | E |

Section II

| | | | | | |
|---|---|---|---|---|---|
| 1 | A | ■ | C | D | E |
| 2 | ■ | B | C | D | E |
| 3 | A | B | C | ■ | E |
| 4 | A | B | ■ | D | E |
| 5 | A | B | C | ■ | E |
| 6 | A | B | ■ | D | E |
| 7 | A | B | C | ■ | E |
| 8 | A | ■ | C | D | E |
| 9 | ■ | B | C | D | E |
| 10 | A | B | ■ | D | E |
| 11 | ■ | B | ■ | D | E |
| 12 | A | ■ | C | D | ■ |
| 13 | A | B | ■ | D | ■ |
| 14 | ■ | B | ■ | D | E |
| 15 | A | B | ■ | D | ■ |
| 16 | A | ■ | C | ■ | E |
| 17 | ■ | B | C | ■ | E |
| 18 | A | ■ | C | D | ■ |
| 19 | A | B | ■ | D | ■ |
| 20 | ■ | B | ■ | D | E |
| 21 | A | ■ | C | ■ | E |
| 22 | A | B | ■ | D | ■ |
| 23 | A | ■ | C | ■ | E |
| 24 | A | ■ | C | ■ | E |
| 25 | A | ■ | C | D | ■ |

**FIGURE 1.1:** Correct Answer Sheet and Student Answer Sheet

# IMAGE REGISTRATION

Image registration is the process of aligning two or more images of the same scene. This process involves designating one image as the reference (also called the reference image or the fixed image), and applying geometric transformations to the other images so that they align with the reference. Images can be misaligned for a variety of reasons. Commonly, the images are captured under variable conditions that can change camera perspective. Misalignment can also be the result of lens and sensor distortions or differences between capture devices.

[8]A geometric transformation maps locations in one image to new locations in another image. The step of determining the correct geometric transformation parameters is key to the image registration process.

Image registration is often used as a preliminary step in other image processing applications. [18]For example, you can use image registration to align satellite images or to align medical images captured with different diagnostic modalities (MRI and SPECT). Image registration allows you to compare common features in different images. For example, you might discover how a river has migrated, how an area became flooded, or whether a tumor is visible in an MRI or SPECT image.For alignment of images with reference one we have used SURF detector.

## 2.1 Intensity-based vs feature-based

Image registration or image alignment algorithms can be classified into intensity-based and feature-based. One of the images is referred to as the reference or source and the others are respectively referred to as the target, sensed or subject images. Image registration involves spatially registering the target image(s) to align with the reference image. Intensity-based methods compare intensity patterns in images via correlation metrics, while feature-based methods find correspondence between image features such as points, lines, and contours. Intensity-based methods register entire images or sub-images. If sub-images are registered, centers of corresponding sub images are treated as corresponding feature points. Feature-based methods establish a correspondence between a numbers of especially distinct points in images. Knowing the correspondence between a numbers of points in images, a geometrical transformation is then determined to map the target image to the reference images, thereby establishing point-by-point correspondence between the reference and target images.

## 2.2 Transformation models

Image registration algorithms can also be classified according to the transformation models they use to relate the target image space to the reference image space. The first broad category of transformation models includes linear transformations, which include rotation, scaling, translation, and other affine transforms. Linear transformations are global in nature, thus, they cannot model local geometric differences between images.

The second category of transformations allows 'elastic' or 'nonrigid' transformations. These transformations are capable of locally warping the target image to align with the reference image. Nonrigid transformations include radial basis functions (thin-plate or surface splines, multiquadrics, and compactly-supported transformations), physical continuum models (viscous fluids), and large deformation models (diffeomorphisms).

There is a number of programs that implement both estimation and application of a warp-field. It is a part of the SPM and AIR programs. Transformations of coordinates via the law of function composition rather than addition**.**

Alternatively, many advanced methods for spatial normalization are building on structure preserving transformations homeomorphisms and diffeomorphisms since they carry smooth submanifolds smoothly during transformation. Diffeomorphisms are generated in the modern field of Computational Anatomy based on flows since diffeomorphisms are not additive although they form a group, but a group under the law of funciton composition. For this reason, flows which generalize the ideas of additive groups allow for generating large deformations that preserve topology, providing 1-1 and onto transformations. Computational methods for generating such transformation are often called LDDMM which provide flows of diffeomorphisms as the main computational tool for connecting coordinate systems corresponding to the geodesic flows of Computational Anatomy.

There are a number of programs which generate diffeomorphic transformations of coordinates via diffeomorphic mapping including MRI Studio and MRI Cloud.org.

### 2.2.1 Spatial vs frequency domain methods

Spatial methods operate in the image domain, matching intensity patterns or features in images. Some of the feature matching algorithms are outgrowths of traditional techniques for performing manual image registration, in which an operator chooses corresponding control points (CP) in images. When the number of control points exceeds the minimum required to define the appropriate transformation model, iterative algorithms like RANSAC can be used to robustly estimate the parameters of a particular transformation type (e.g. affine) for registration of the images.

Frequency-domain methods find the transformation parameters for registration of the images while working in the transform domain. Such methods work for simple transformations, such as translation, rotation, and scaling. Applying the phase correlation method to a pair of images produces a third image which contains a single peak. The location of this peak corresponds to the relative translation between the images. Unlike many spatial-domain algorithms, the phase correlation method is resilient to noise, occlusions, and other defects typical of medical or satellite images. Additionally, the phase correlation uses the fast Fourier transform to compute the cross-correlation between the two images, generally resulting in large performance gains. The method can be extended to determine rotation and scaling differences between two images by first converting the images to log-polar coordinates. Due to properties of the Fourier transform, the rotation and scaling parameters can be determined in a manner invariant to translation.

### 2.2.2 Single vs multi-modality methods

Another classification can be made between single-modality and multi-modality methods. Single-modality methods tend to register images in the same modality acquired by the same scanner/sensor type, while multi-modality registration methods tended to register images acquired by different scanner/sensor types.

Multi-modality registration methods are often used in medical imaging as images of a subject are frequently obtained from different scanners. Examples include registration of brain CT/MRI images or whole body PET/CT images for tumor localization, registration of contrast-enhanced CT images against non-contrast-enhanced CT images for segmentation of specific parts of the anatomy, and registration of ultrasound and CT images for prostate localization in radiotherapy.

### 2.2.3 Automatic vs interactive methods

Registration methods may be classified based on the level of automation they provide. Manual, interactive, semi-automatic, and automatic methods have been developed. Manual methods provide tools to align the images manually. Interactive methods reduce user bias by performing certain key operations automatically while still relying on the user to guide the registration. Semi-automatic methods perform more of the registration steps automatically but depend on the user to verify the correctness of a registration. Automatic methods do not allow any user interaction and perform all registration steps automatically.

### 2.2.4 Similarity measures for image registration

Image similarities are broadly used in medical imaging. An image similarity measure quantifies the degree of similarity between intensity patterns in two images. The choice of an image similarity measure depends on the modality of the images to be registered. Common examples of image similarity measures include cross-correlation, mutual information, sum of squared intensity differences, and ratio image uniformity. Mutual information and normalized mutual information are the most popular image similarity measures for registration of multimodality images. Cross-correlation, sum of squared intensity differences and ratio image uniformity are commonly used for registration of images in the same modality.

Many new features have been derived for cost functions based on matching methods via large deformations have emerged in the field Computational Anatomy including

Measure matching which are pointsets or landmarks without correspondence, Curve matching and Surface matching via mathematical currents and varifolds.

## 2.3 SURF Detector

**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.

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 precomputed integral image. [9]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 to detect the key points of the template and the answer sheet. There are several options we can choose: FAST detector, ORB detector and SURF detector. After trying all these 3 detectors, as expected we found that FAST detector is the fastest one to extract features. But SURF is the most robust one, which detects better features. Therefore, we decided to use SURF. SURF detector (Speeded Up Robust Features) is a robust local feature detector. It uses an integer approximation to the determinant of Hessian blob detector, which can be computed extremely fast with an integral image. For features, it uses sum of the Haar wavelet response around the point of interest. These also can be computed using the integral image.

**Following are stages of SURF detector:-**

1. [16]Detect interest points, use Hessian matrix approximation. Build the integral images and the scale space of image.

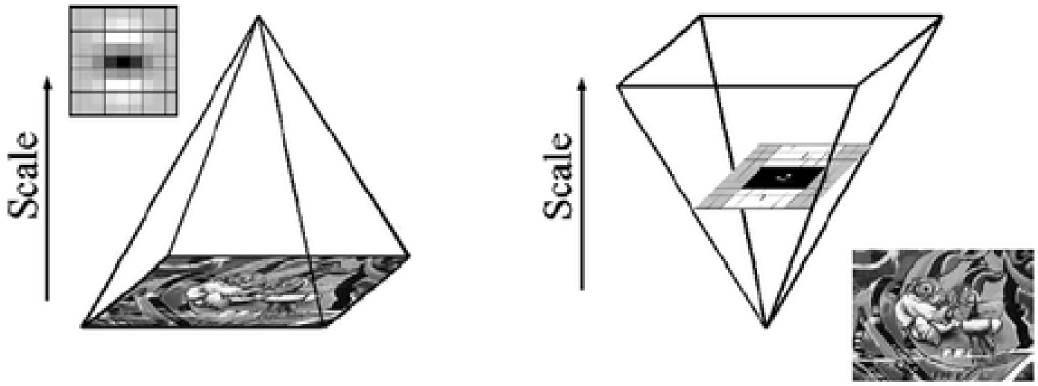
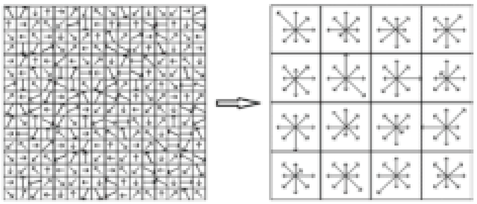
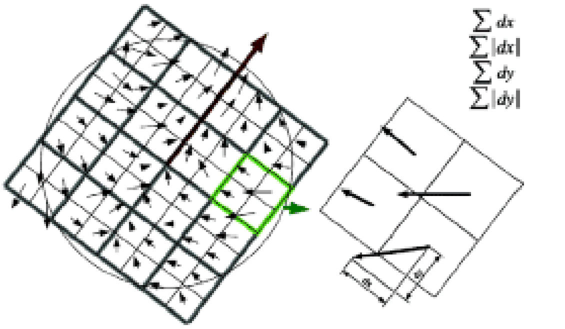


**FIGURE 2.1:** Scale Space of Image

2. Interest point description and matching, descriptor describes the distribution of the intensity content, similar to SIFT. Based on sum of Haar wavelet responses, construct a square region centered on the interest point and oriented along the orientation selected in previous section.

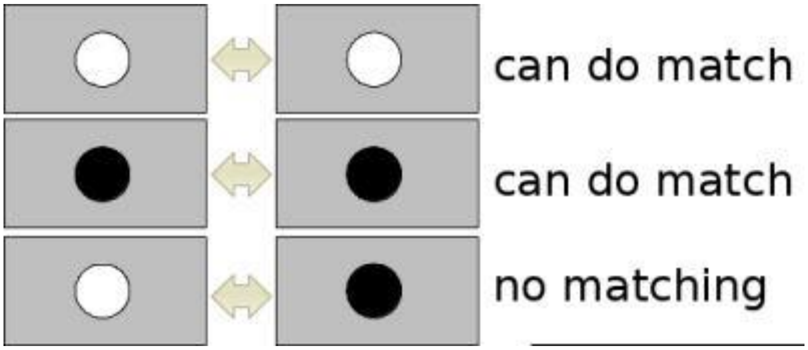
**Table 2.1:** Difference between SURF and SIFT

| | SIFT | SURF |
|---|---|---|
| Scale Space | DOG convolved with difference size images | difference size box filter convolved with integral image |
| Key points detection | Local extrema detection->Non-maximum suppression->Eliminate edge responses with Hessian matirx. | Deter mint the potential key points with Hessian matrix and Non-maximum suppression |
| Orientation | image gradient magnitudes and orientations are sampled around the key point location, using the scale of the key point to select the level of Gaussian blur for the image | A sliding orientation window of size pi/3 detects the dominant orientation of the Gaussian weighted Haar wavelet responses at every sample point within a circular neighbourhood around the interest point. |
| Descriptor | The key point descriptor allows for significant shift in gradient positions by creating orientation histograms over 4x4 sample regions. The figure shows 8 directions for each orientation histogram , with the length of each arrow corresponding to the magnitude of that histogram entry.<br><br>Figure 1: SIFT Descriptor Generation | An oriented quadratic grid with 4×4 square sub-regions is laid over the interest point (left). For each square, the wavelet responses are computed from 5×5 samples. For each field, we collect the sums dx, \|dx\|; dy and \|dy\|, computed relatively to the orientation of the grid (right).<br><br> |

For feature description, SURF uses Wavelet responses in horizontal and vertical direction (again, use of integral images makes things easier). A neighbourhood of size 20sX20s is taken around the key point where s is the size. It is divided into 4x4 sub regions. For each sub region, horizontal and vertical wavelet responses are taken and a vector is formed like this, $v = (\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|)$ . This when represented as a vector gives SURF feature descriptor with total 64 dimensions. Lower the dimension, higher the speed of computation and matching, but provide better distinctiveness of features.

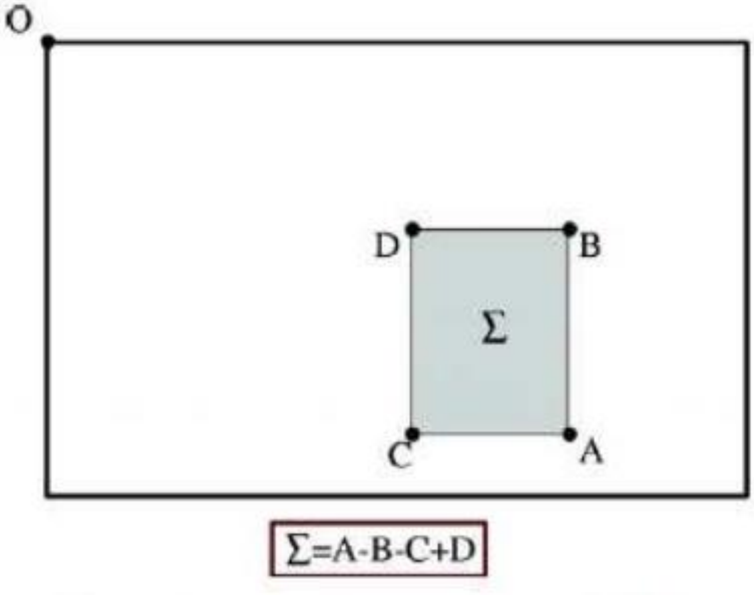
For more distinctiveness, SURF feature descriptor has an extended 128 dimension version. The sums of $d_x$ and $|d_x|$ are computed separately for $d_y < 0$ and $d_y \geq 0$. Similarly, the sums of $d_y$ and $|d_y|$ are split up according to the sign of $d_x$ , thereby doubling the number of features. It doesn't add much computation complexity. OpenCV supports both by setting the value of flag extended with 0 and 1 for 64-dim and 128-dim respectively (default is 128-dim)

Another important improvement is the use of sign of Laplacian (trace of Hessian Matrix) for underlying interest point. It adds no computation cost since it is already computed during detection. The sign of the Laplacian distinguishes bright blobs on dark backgrounds from the reverse situation. In the matching stage, we only compare features if they have the same type of contrast (as shown in image below). This minimal information allows for faster matching, without reducing the descriptor's performance.



**2.2:** Compare only with same contrast

In short, SURF adds a lot of features to improve the speed in every step. Analysis shows it is 3 times faster than SIFT while performance is comparable to SIFT. SURF is good at handling images with blurring and rotation, but not good at handling viewpoint change and illumination change.



**2.3:** Calculating sum of intensities inside rectangular region

### 2.3.1 INTEREST POINT DETECTION

An interest point is a featured location in an image. It is usually in a corner or in a T junction. For the detection it is used the Hessian-matrix approximation and integral images. They allow for fast computation of box type convolution filters. In an integral image the value of a pixel (x, y) is the sum of all the pixels in the square region within the origin and the pixel (x, y). It takes only three additions and four memory access to calculate the sum of intensities inside a rectangular region of any size. Hessian matrix based interest points. The detector is based on the Hessian matrix.[4] We detect blob-like structures at locations were the determinant is maximum. In contrast to the Hessian-Laplace detector. We rely on the determinant of the Hessian also for the scale selection. Hessian matrix, where Lxx(x, σ) is the convolution of the Gaussian second order derivative at point x and scale sigma, and Dxx is the discretised version.

### 2.3.2 Fast indexing during the matching stage

For fast indexing during the matching stage, the sign of the Laplacian (i.e. the trace of the Hessian matrix) for the underlying interest point is included. Typically, the interest points are found at blob-type structures. The sign of the Laplacian distinguishes bright blobs on dark backgrounds from the reverse situation. This feature is available at no extra computational cost as it was already computed during the detection phase. In the matching stage, we only compare features if they have the same type of contrast. Hence, this minimal information allows for faster matching, without reducing the descriptor's performance. Note that this is also of advantage for more advanced indexing methods.

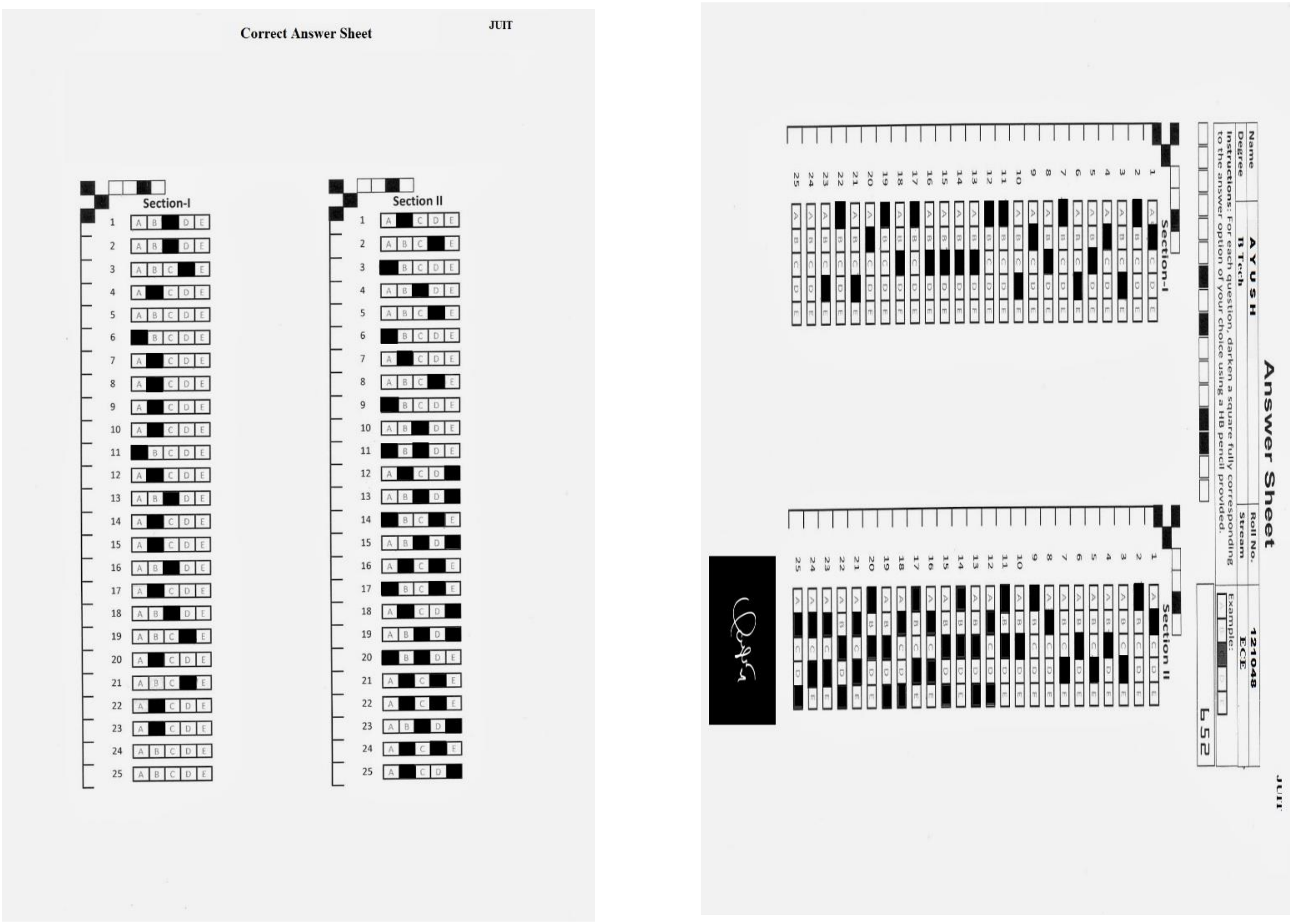

Correct Answer Sheet
JUIT
Section-I
Section II
Answer Sheet
Name
Degree
A Y U S H
B Tech
Roll No.
Stream
121048
ECE
Instructions: For each question, darken a square fully corresponding to the answer option of your choice using a HB pencil provided.
Example:
Section-I
Section II

**FIGURE 2.4:** Original and 90 degree tilted Image

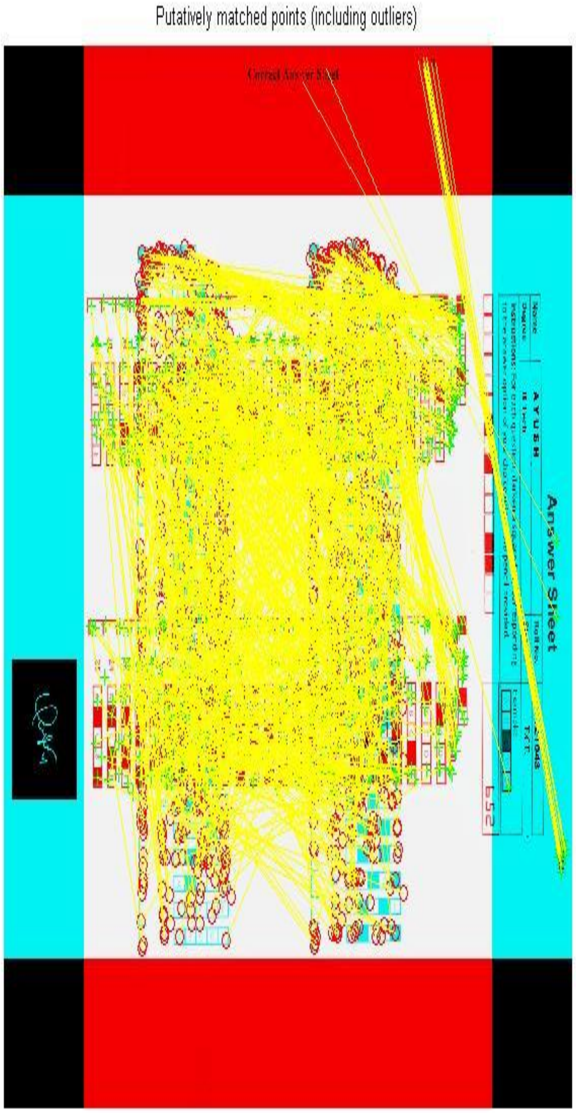


**FIGURE 2.5:** Putatively matched points (including outliers)

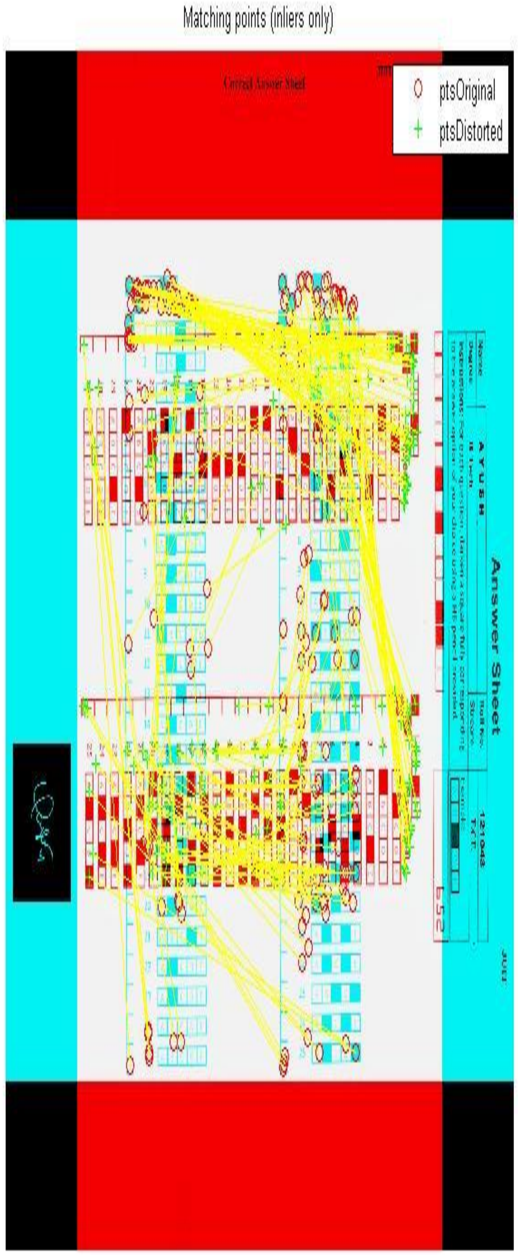
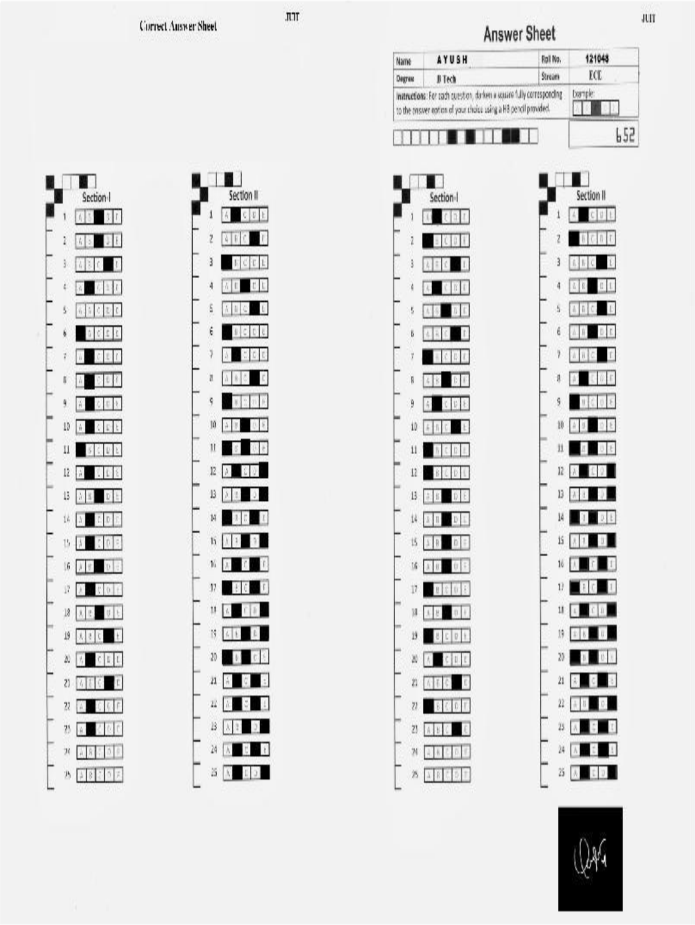


**FIGURE 2.6:** Matched points (inliers only)



Correct Answer Sheet

JUIT

Answer Sheet

JUIT

| | | | |
|---|---|---|---|
| Name | A Y U S H | Roll No. | 121048 |
| Degree | B Tech | Stream | ECE |

Instruction: For each question, darken a square fully corresponding to the answer option of your choice using a HB pencil provided.

Example:

652
Section-I
Section II
Section-I
Section II

**FIGURE 2.7:** Aligned Answer Sheet

CHAPTER 3

# OPTICAL CHARACTER RECOGNITION

**Optical character recognition (optical character reader)** (**OCR**) is the mechanical or electronic conversion of images of typed, handwritten or printed text into machine-encoded text, whether from a scanned document, a photo of a document, a scene-photo (for example the text on signs and billboards in a landscape photo) or from subtitle text superimposed on an image (for example from a television broadcast).[13]It is widely used as a form of data entry from printed paper data records, whether passport documents, invoices, bank statements, computerised receipts, business cards, mail, printouts of static-data, or any suitable documentation.

In this we have loaded the templates. We have computed the number of letters in template file. We have label and counted the number of connected component in the given text. We recognized the characters individually in the given text.

Automatic Character recognition include a number of problems which make mandatory the development of an automatic process of classifying input information according to the specific requirements imposed on such a classification. The problem of character recognition results in the automatic making of the decision on the basis of data which does not directly indicate the best of all possible decisions.

## 3.1 Binarization

The first step in Binarization is to convert a color image into a grayscale image. A grayscale digital image is an image in which each pixel is quantized exclusively the shades of neutral gray, varying from black at the weakest intensity to white at the strongest intensity.[2] To convert any color to its most approximate gray level, one must obtain the values of primary colors (Red, Green, and Blue).Then, add 30% of red, 59% of green and 11% of blue value together. Regardless the scale employed (0.0 to 1.0 or 0 to 255) the resultant number is the desired grey value. The obtained gray image is then binaries by giving the pixel blues “1” or white shade and “0” for black shade.

### 3.1.1 Algorithm: Binarization

1. Convert the color image into grayscale image.
2. Find the gray level histogram of input image F(x, y).
3. Find the valley point between two modes and select the point as threshold (T).
4. Binary image b(x,y) is defined as

$$b(x,y) = \begin{cases} 1 & for\ f(x,y) > T \\ 0 & Otherwise \end{cases}$$

[3]Then find the gray level histogram of gray scale image. The gray level histogram is composed of two dominant modes. One dominant mode corresponds to background and other is character image. Select the threshold value such that the two dominant modes meet at a valley point.

## 3.2 Character Recognition Using Template Matching

“There exists a set M of some objects which are divided into n nonintersecting subsets called object classes or characters. To each character there corresponds a specific character description x which, without restriction may be considered as multidimensional vector. The description of objects are not necessarily unique i.e. identical description may sometimes correspond to different object classes. “The problem is to design an algorithm optimal in some sense which on the basis of given description of the character would indicate the class to which it belongs. In the most general case, a reader which recognize characters must solve a problem equivalent to the calculation of the several logic function(depending on the character being distinguished), such that each function is equal to unity when and only when there is a character corresponding to this function in the field of vision of the character reader. [14]These functions must be invariant with respect to all shifts and changes in the outlines of the characters which are considered permissible for this character reader. It is possible that in the calculation of functions corresponding to various characters several actions or sequences of action will be repeated. These sequences of action yield values of elementary functions that are common to all the characters. In order to recognize a character it is necessary to determine which value the elementary function has for this character. It is natural to call these elementary function indicators. The indicators must be invariant with respect to the permissible changes in the characters. The direction of the stroke, which makes up the character, can be taken as indicators, which are invariant with respect to the forward movement and changes in the dimensions and in some cases with respect to the change in the proportion in the shape of the character. If the direction of the strokes is determined approximately, similar directions being taken as identical, then it is possible in addition to obtain variance with respect to slight rotations of a character or part of it. The directions of the stroke alone do not provide exhaustive information about the character e.g. “L” may be characterized by identical stroke directions. Additional information can be obtained about the character if the direction of is determined, not of the stroke themselves but also of the their boundaries i.e. The boundaries between the black and white fields, and if in this process account is taken as to which side of the boundary the black field is on. It is most convenient to analyze characters by means of such indicators moving along the boundary of a stroke and recording the direction of movement in the sequence in which it occurs in the character. These techniques are different from the others in that no features are actually extracted. Instead the matrix containing the image of the input character is directly matched with a set of prototype characters representing each possible class. The distance between the pattern and each prototype is computed, and the class of the prototype giving the best match is assigned to the pattern. The technique is simple and easy to implement in hardware and has been used in many commercial OCR machines. However, this technique is sensitive to noise and style variations and has no way of handling rotated characters.

# TEMPLATE MATCHING

The used templates are given in the figure below:

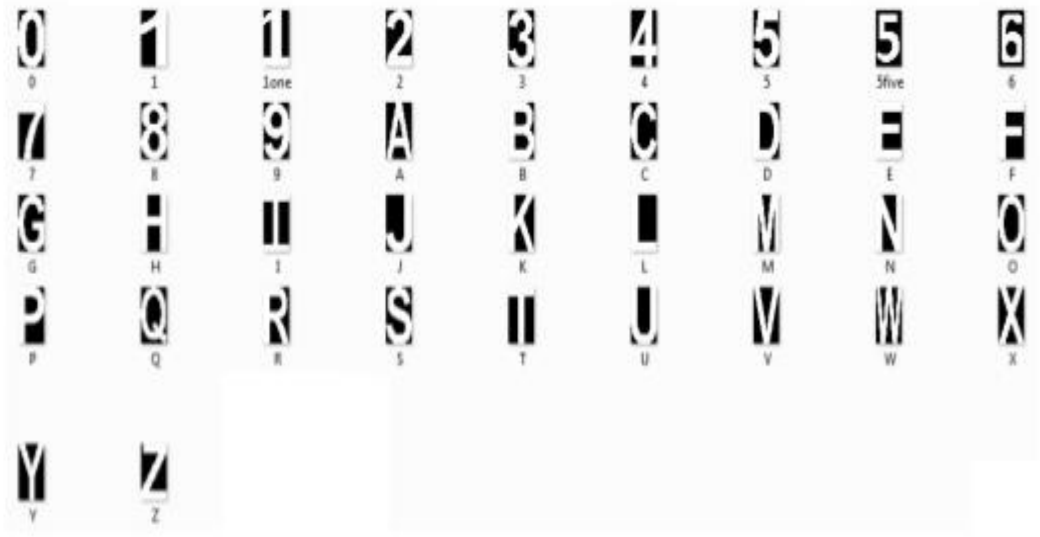
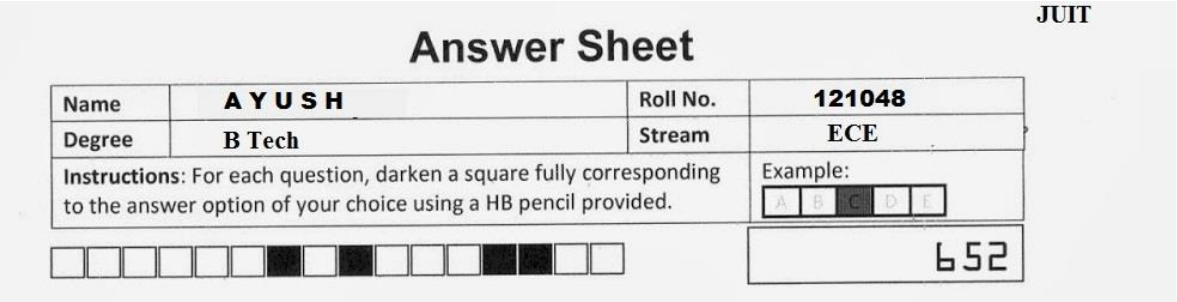


**Figure 3.1**: Used Templates in recognition



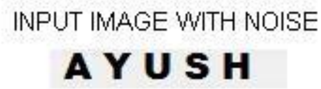
JUIT

Answer Sheet

| | | | |
|---|---|---|---|
| Name | AYUSH | Roll No. | 121048 |
| Degree | B Tech | Stream | ECE |
| **Instructions**: For each question, darken a square fully corresponding to the answer option of your choice using a HB pencil provided. | | Example: □ A □ B ■ C □ D □ E | |

□□□□□□□■□■□□□■■□□ 652

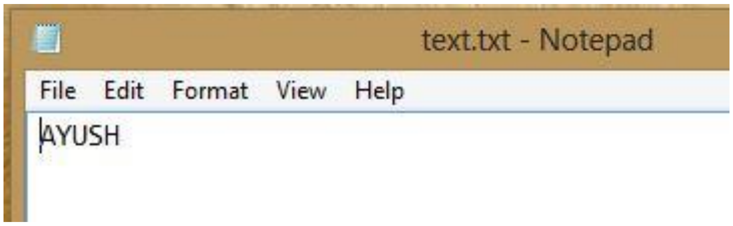
**Figure 3.2**: Image to be cropped



INPUT IMAGE WITH NOISE

AYUSH

**Figure3.3:**Input image with noise



AYUSH

**Figure3.4:**Name recognized

# Signature Matching

## 4.1 Clear Border

It suppresses structures that are lighter than their surroundings and that are connected to the image border. Use this function to clear the image border. IM can be a grayscale or binary image. For grayscale images, imclearborder tends to reduce the overall intensity level in addition to suppressing border structures. The output image, IM2, is grayscale or binary, depending on the input. The default connectivity is 8 for two dimensions, 26 for three dimensions, and conndef(ndims(BW),'maximal') for higher dimensions.

## 4.2 Edge detection

It 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 formally, has discontinuities.[5] The points at which image brightness changes sharply are typically organized into a set of curved line segments termed edges. The same problem of finding discontinuities in 1D signals is known asstep detection and the problem of finding signal discontinuities over time is known as change detection. Edge detection is a fundamental tool in image processing, machine vision and computer vision, particularly in the areas of feature detection and feature extraction

## 4.3 Edge detection methods for finding object boundaries in images

Edge detection is an image processing technique for finding the boundaries of objects within images. [6]It works by detecting discontinuities in brightness. Edge detection is used for image segmentation and data extraction in areas such as image processing, computer vision, and machine vision. Common edge detection algorithms include Sobel, Canny, Prewitt, Roberts, and fuzzy logic methods.

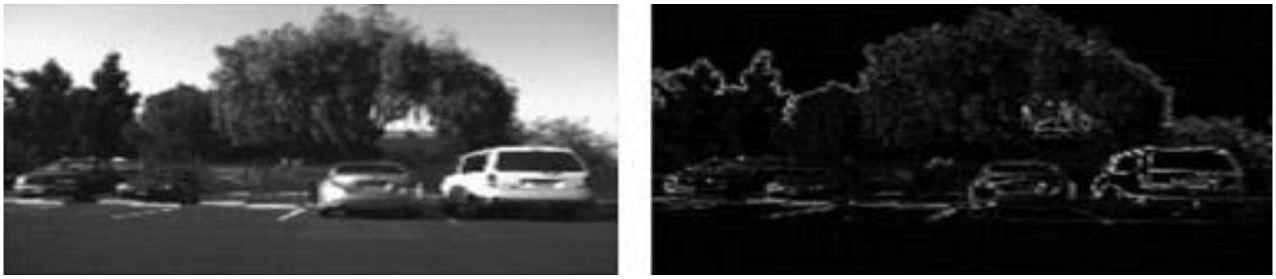


**FIGURE 4.1:** Image segmentation using the Sobel method**.**

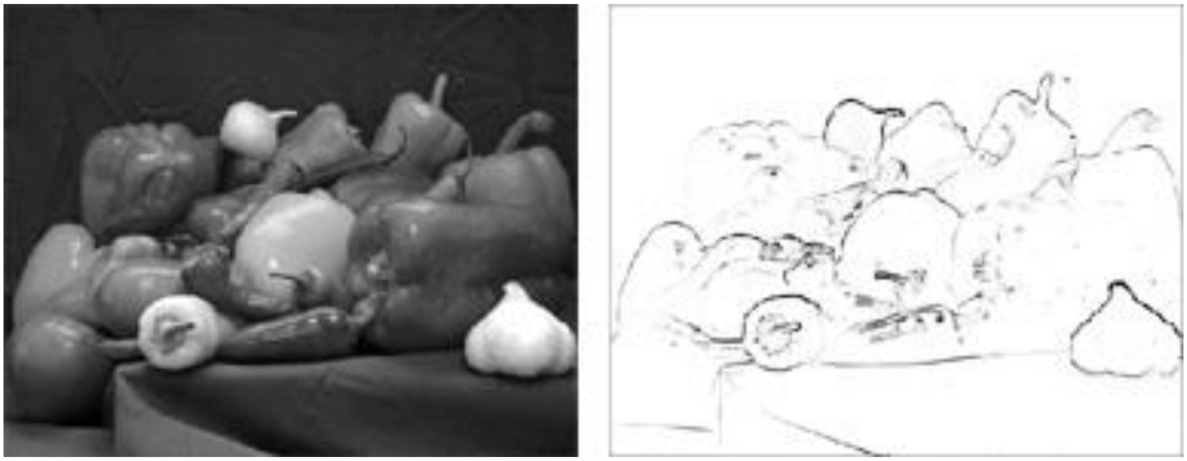


**FIGURE 4.2:** Image segmentation using the Canny method**.**

[11]The general criteria for edge detection includes:

1. Detection of edge with low error rate, which means that the detection should accurately catch as many edges shown in the image as possible
2. The edge point detected from the operator should accurately localize on the centre of the edge.
3. A given edge in the image should only be marked once, and where possible, image noise should not create false edges.

[10]To satisfy these requirements Canny used the calculus of variations – a technique which finds the function which optimizes a given functional. The optimal function in

Canny's detector is described by the sum of four exponential terms, but it can be approximated by the first derivative of a Gaussian.

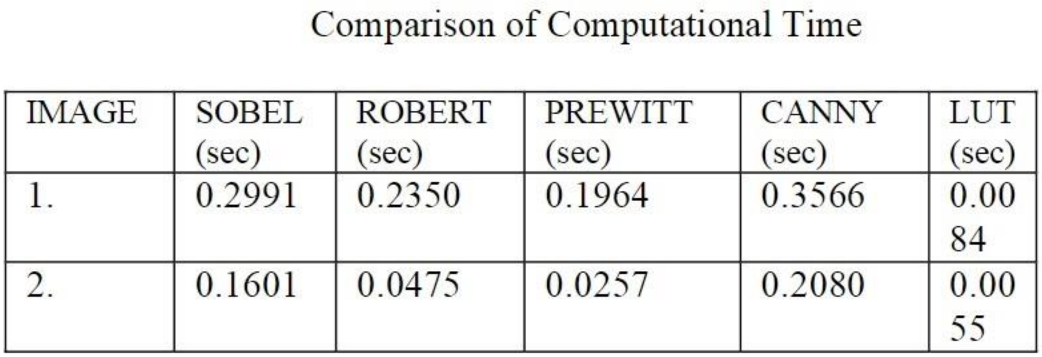
Among the edge detection methods developed so far, canny edge detection algorithm is one of the most strictly defined methods that provides good and reliable detection. Owing to its optimality to meet with the three criteria for edge detection and the simplicity of process for implementation, it becomes one of the most popular algorithms for edge detection.

## 4.4 Process of Canny edge detection algorithm

The Process of Canny edge detection algorithm can be broken down to 5 different steps:

1. Apply Gaussian filter to smooth the image in order to remove the noise
2. Find the intensity gradients of the image
3. Apply non-maximum suppression to get rid of spurious response to edge detection
4. Apply double threshold to determine potential edges
5. Track edge by hysteresis: Finalize the detection of edges by suppressing all the other edges that are weak and not connected to strong edges.

**TABLE 4.1:** Comparison of computational time



Comparison of Computational Time

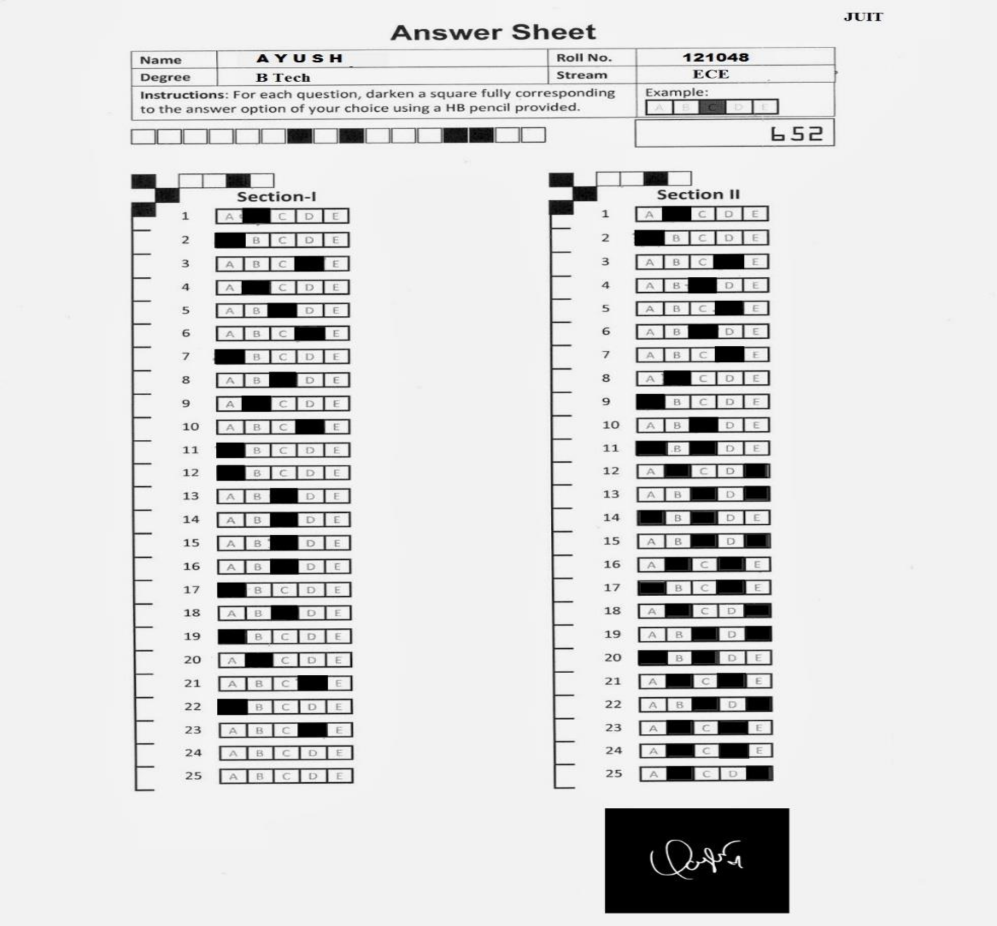
| IMAGE | SOBEL (sec) | ROBERT (sec) | PREWITT (sec) | CANNY (sec) | LUT (sec) |
|---|---|---|---|---|---|
| 1. | 0.2991 | 0.2350 | 0.1964 | 0.3566 | 0.0084 |
| 2. | 0.1601 | 0.0475 | 0.0257 | 0.2080 | 0.0055 |

## 4.5 Weiner Filtering

[12]The most important technique for removal of blur in images due to linear motion or unfocussed optics is the Wiener filter. From a signal processing standpoint, blurring due to linear motion in a photograph is the result of poor sampling. Each pixel in a digital representation of the photograph should represent the intensity of a single stationary point in front of the camera. Unfortunately, if the shutter speed is too slow and the camera is in motion, a given pixel will be an amalgram of intensities from points along the line of the camera's motion.

This is a two-dimensional analogy to G**(u,v)=F(u,v).H(u,v)** where F is the fourier transform of an "ideal" version of a given image, and H is the blurring function. In this case H is a sinc function: if three pixels in a line contain info from the same point on an image, the digital image will seem to have been convolved with a three-point boxcar in the time domain. Ideally one could reverse-engineer a Fest, or F estimate, if G and H are known. This technique is inverse filtering. It should be noted that the image restoration tools described here work in a similar manner for cases with blur due to incorrect focus. In this case the only difference is in the selection of H. The 2-d Fourier transform of H for motion is a series of sinc functions in parallel on a line perpendicular to the direction of motion; and the 2-d Fourier transform of H for focus blurring is the sombrero function, described elsewhere. In the real world, however, there are two problems with this method. First, H is not known precisely. Engineers can guess at the blurring function for a given circumstance, but determination of a good blurring function requires lots of trial and error. Second, inverse filtering fails in some circumstances because the sinc function goes to 0 at some values of x and y. Real pictures contain noise which becomes amplified to the point of destroying all attempts at reconstruction of a Fest. The goal of the Wiener filter is to filter out Noise that has corrupted a signal. It is based on a statistical approach. Typical filters are designed for a desired frequency response. However, the design of the Wiener filter takes a different approach. One is assumed to have knowledge of the spectral properties of the original signal and the noise, and one seeks the linear time-invariant filter whose output would come as close to the original signal as possible. Wiener filters are characterized by the following:

1. Assumption: signal and (additive) noise are stationary linear stochastic processes with known spectral characteristics or known autocorrelation and cross-correlation.
2. Requirement: the filter must be physically realizable/causal (this requirement can be dropped, resulting in a non-causal solution).
3. Performance criterion: minimum mean-square error (MMSE).



JUIT

Answer Sheet

| | | | |
|---|---|---|---|
| Name | AYUSH | Roll No. | 121048 |
| Degree | B Tech | Stream | ECE |
| **Instructions**: For each question, darken a square fully corresponding to the answer option of your choice using a HB pencil provided. | | Example:<br>A B ■ D E | |
| | | | 652 |

Section-I

| | | | | | |
|---|---|---|---|---|---|
| 1 | A | ■ | C | D | E |
| 2 | ■ | B | C | D | E |
| 3 | A | B | C | ■ | E |
| 4 | A | ■ | C | D | E |
| 5 | A | B | ■ | D | E |
| 6 | A | B | C | ■ | E |
| 7 | ■ | B | C | D | E |
| 8 | A | B | ■ | D | E |
| 9 | A | ■ | C | D | E |
| 10 | A | B | C | ■ | E |
| 11 | ■ | B | C | D | E |
| 12 | ■ | B | C | D | E |
| 13 | A | B | ■ | D | E |
| 14 | A | B | ■ | D | E |
| 15 | A | B | ■ | D | E |
| 16 | A | B | ■ | D | E |
| 17 | ■ | B | C | D | E |
| 18 | A | B | ■ | D | E |
| 19 | ■ | B | C | D | E |
| 20 | A | ■ | C | D | E |
| 21 | A | B | C | ■ | E |
| 22 | ■ | B | C | D | E |
| 23 | A | B | C | ■ | E |
| 24 | A | B | C | D | E |
| 25 | A | B | C | D | E |

Section II

| | | | | | |
|---|---|---|---|---|---|
| 1 | A | ■ | C | D | E |
| 2 | ■ | B | C | D | E |
| 3 | A | B | C | ■ | E |
| 4 | A | B | ■ | D | E |
| 5 | A | B | C | ■ | E |
| 6 | A | B | ■ | D | E |
| 7 | A | B | C | ■ | E |
| 8 | A | ■ | C | D | E |
| 9 | ■ | B | C | D | E |
| 10 | A | B | ■ | D | E |
| 11 | ■ | B | ■ | D | E |
| 12 | A | ■ | C | D | ■ |
| 13 | A | B | ■ | D | ■ |
| 14 | ■ | B | ■ | D | E |
| 15 | A | B | ■ | D | ■ |
| 16 | A | ■ | C | ■ | E |
| 17 | ■ | B | C | ■ | E |
| 18 | A | ■ | C | D | ■ |
| 19 | A | B | ■ | D | ■ |
| 20 | ■ | B | ■ | D | E |
| 21 | A | ■ | C | ■ | E |
| 22 | A | B | ■ | D | ■ |
| 23 | A | ■ | C | ■ | E |
| 24 | A | ■ | C | ■ | E |
| 25 | A | ■ | C | D | ■ |

**FIGURE 4.3:** Original image with signature

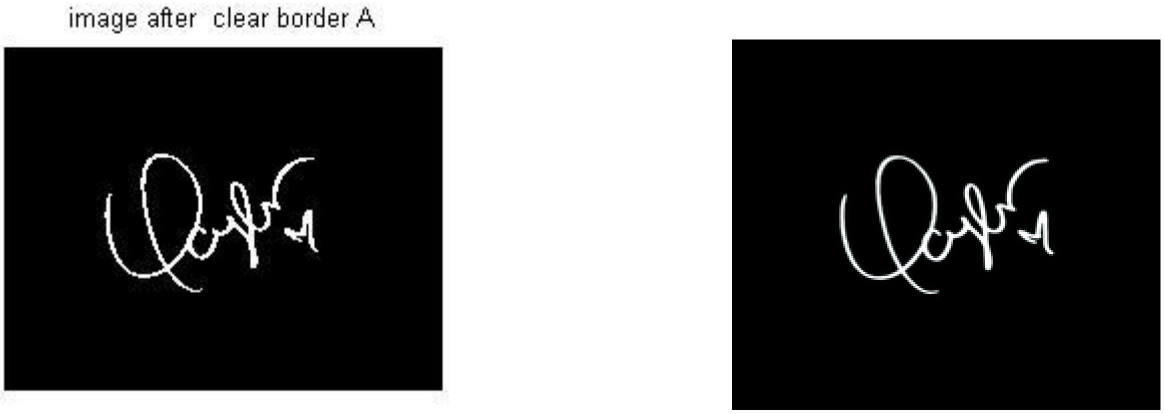


**FIGURE 4.4:** Image cropped after clearing border and Sign for matching

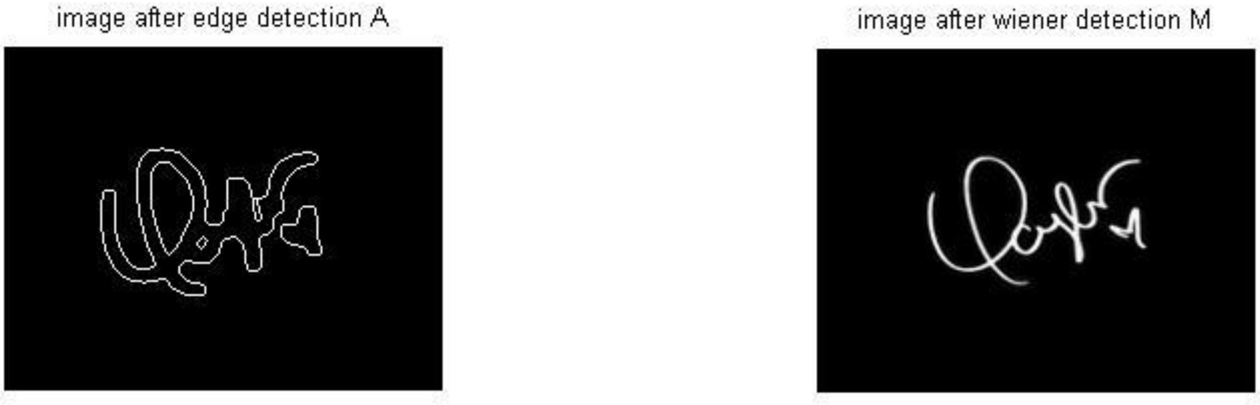


**FIGURE 4.5:** Images after edge detection

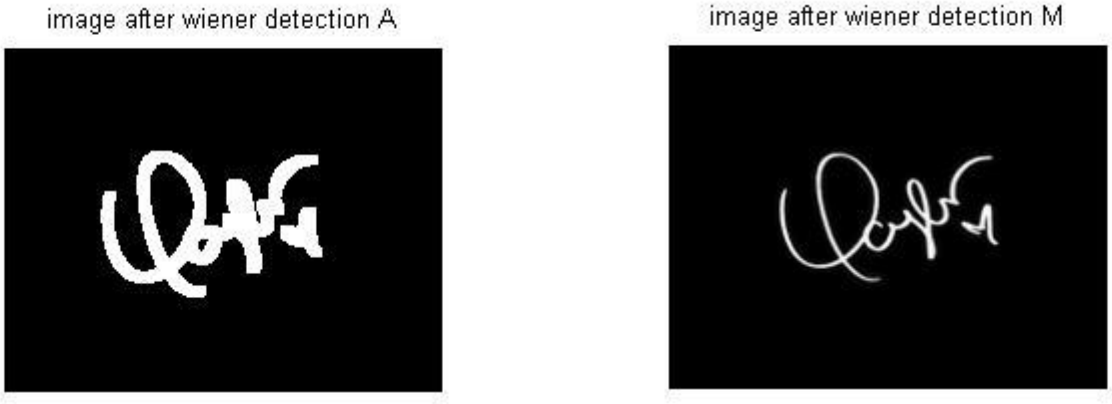


image after wiener detection A
image after wiener detection M

**FIGURE 4.6:** Images after wiener detection

CHAPTER 5

# FINDING AND RECOGNIZING FILLED BUBBLES

## 5.1 Thresholding

The simplest thresholding methods replace each pixel in an image with a black pixel if the image intensity $I_{i,j}$ is less than some fixed constant T (that is, $I_{i,j} < T$), or a white pixel if the image intensity is greater than that constant. In the example image on the right, this results in the dark tree becoming completely black, and the white snow becoming complete white.

[7]This method *weighs* the histogram, checks which of the two sides is heavier, and removes weight from the heavier side until it becomes the lighter. It repeats the same operation until the edges of the weighing scale meet.

If you get the threshold wrong the results can be disastrous

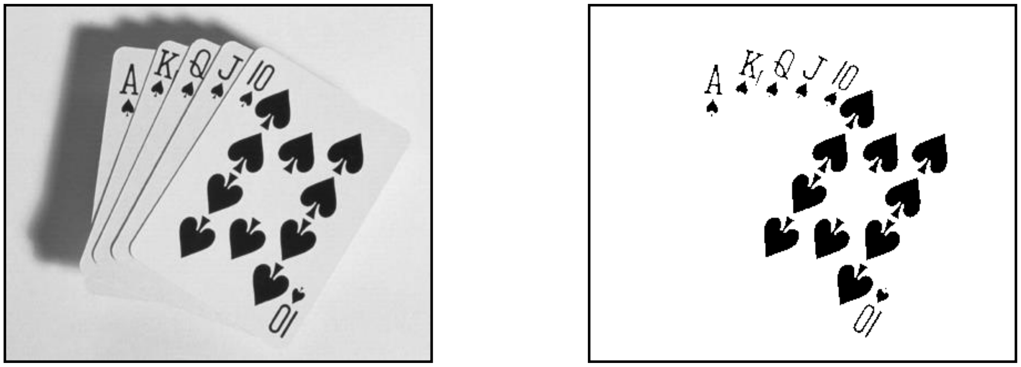


**FIGURE 5.1:** Original Image and Thresholded Image

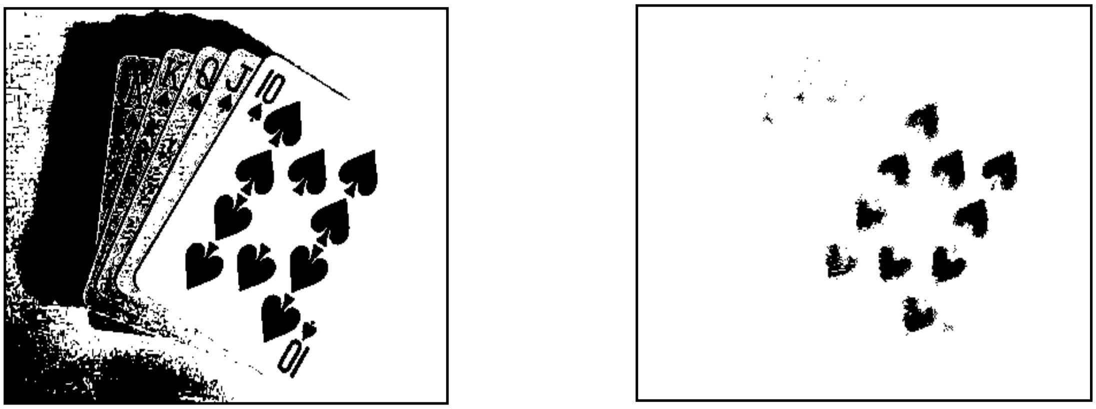


**FIGURE 5.2**: Threshold Too High and Threshold Too Low

## 5.2 Types of thresholding

### 5.2.1Basic Global Thresholding

It is based on the histogram of an image. Partition the image histogram using a single global threshold. The success of this technique very strongly depends on how well the histogram can be partitioned.

### 5.2.2 Basic Adaptive Thresholding

An approach to handling situations in which single value thresholding will not work is to divide an image into sub images and threshold these individually. Since the threshold for each pixel depends on its location within an image this technique is said to adaptive. Problem with global thresholding is that changes in illumination across the scene may cause some parts to be brighter (in the light) and some parts darker (in shadow) in ways that have nothing to do with the objects in the image. We can deal, at least in part, with such uneven illumination by determining thresholds locally. That is, instead of having a single global threshold, we allow the threshold itself to smoothly vary across the image.

### 5.2.3 Local Thresholding

The problem with global thresholding is that changes in illumination across the scene may cause some parts to be brighter (in the light) and some parts darker (in shadow) in ways that have nothing to do with the objects in the image. We can deal, at least in part, with such uneven illumination by determining thresholds locally. That is, instead of having a single global threshold, we allow the threshold itself to smoothly vary across the image.

## 5.3 Thresholding algorithm

In image processing, the balanced histogram thresholding method (BHT), is a very simple method used for automatic image thresholding. Like Otsu's Method and the Iterative Selection Thresholding Method, this is a histogram based thresholding method. This approach assumes that the image is divided in two main classes: The background and the foreground. The BHT method tries to find the optimum threshold level that divides the histogram in two classes.

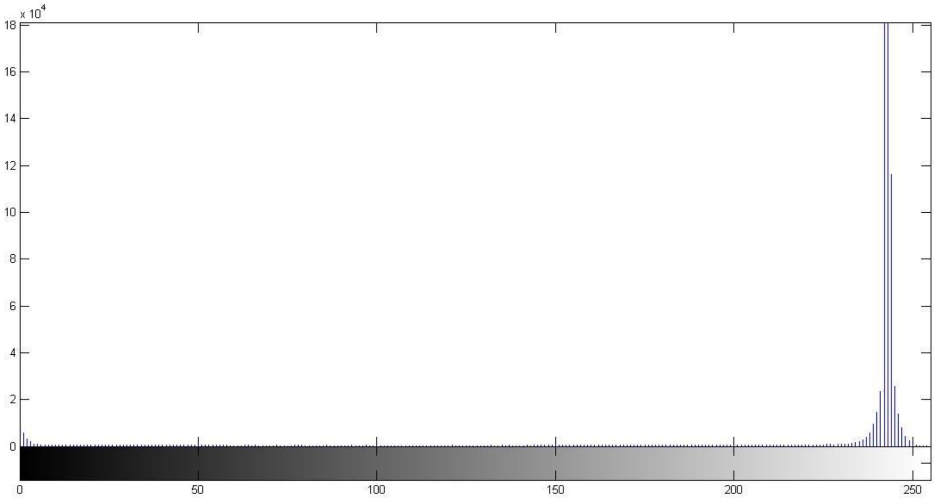


**FIGURE 5.3:** Histogram to calculate the threshold

# WORK AND DESCRIPTION

We have used the simplest thresholding method which replace each pixel in an image with a black pixel if the image intensity $I_{i,j}$ is less than some fixed constant T (that is, $I_{i,j} < T$), or a white pixel if the image intensity is greater than that constant.

## 6.1 Application Flow

1. Image of the reference answer sheet with correct answers, and store answers for each questions.

2. Image of a student’s answer sheet.

3. Automatically compute the number of correct answers and give statistics for the student.

## 6.2 Algorithm

We have 50 questions and 5 options for each.

So total number of bubbles= 250.

1. Run two loops one from i=1: diffx other j=1: diffy.
2. Diffx and diffy are approximate size of the rectangular choice boxes.
3. Then check if the Intensity Iij is less than the set threshold than increase a counter n.
4. If total grey pixels in the rectangle are greater than a certain percentage then mark the corresponding choice.

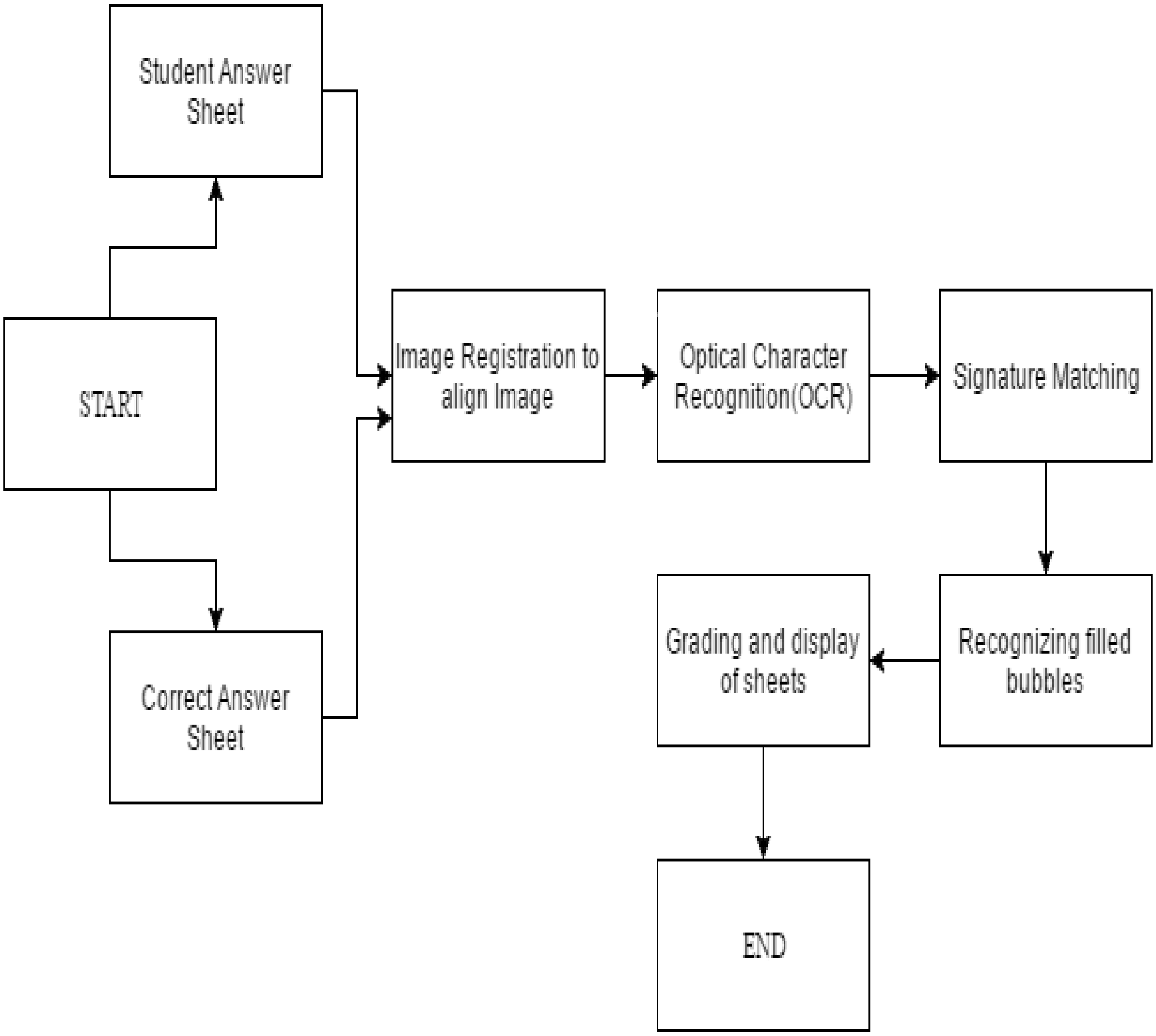

Student Answer Sheet
START
Correct Answer Sheet
Image Registration to align Image
Optical Character Recognition(OCR)
Signature Matching
Recognizing filled bubbles
Grading and display of sheets
END

**FIGURE 6.1:** Flow chart of application

# CHAPTER 7

## RESULT

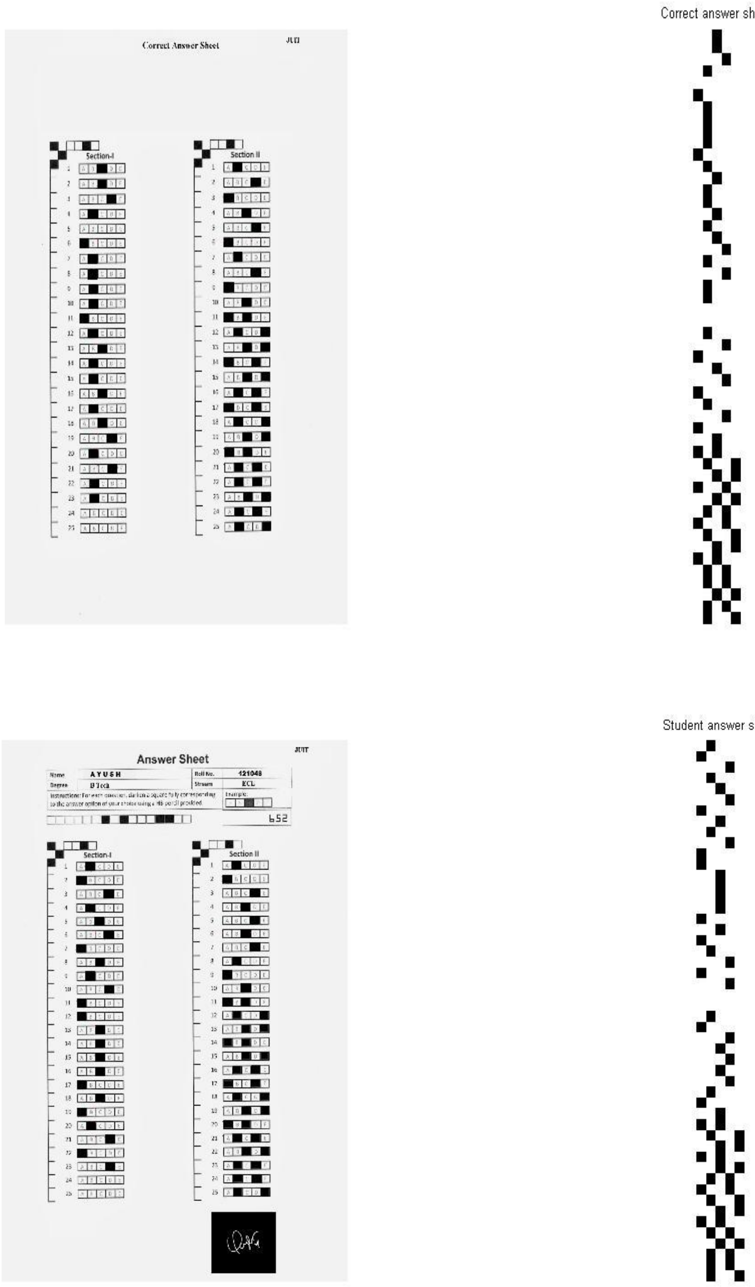
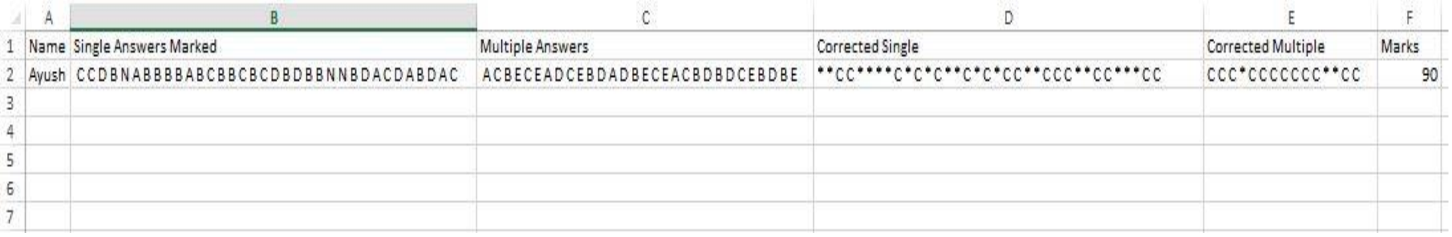


**FIGURE 7.1:** Final Detected Answer Sheets

```
white_points1 =   672
white_points2 =   697
total_matched_percentage =   96.4132
OUTPUT_MESSAGE =Sign Matched

str1 = C C D B N A B B B B A B C B B C B C D B D B B N N B D A C D A B D A C
str2 = B A D B C D A C B D A A C C C C A C A B D A D N N B A D C D C D B A C
strc1 = A C B E C E A D C E B D A D B E C E A C B D B D C E B D B E
strc2 = A C B E C E A C C E B D A D B E C E A C B D C E B D B D B E
correct = * * C C * * * * C * C * C * * C * C * C C * * C C C * * C C * * * C C
correct1 = C C C * C C C C C C C * * C C
marks =   90
```



|  | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | Name | Single Answers Marked | Multiple Answers | Corrected Single | Corrected Multiple | Marks |
| 2 | Ayush | CCDBNABBBBABCBBCBCDBDBBNNBDACDABDAC | ACBECEADCEBDADBECEACBDBDCEBDBE | **CC****C*C*C**C*C*CC**CCC**CC***CC | CCC*CCCCCCC**CC | 90 |
| 3 |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |

**Figure 7.2:** Excel sheet of student answer sheet

# CONCLUSION

The application is based on camera images whose quality can be affected by capturing conditions. Images taken by camera can be tilted, rotated, or blurred by poor illumination condition. In order to test the robustness of our algorithm on these variations, we took pictures from different perspectives and illumination conditions, and applied our extraction algorithm on these images.

For well-aligned images with uniform brightness, our algorithm works well. Our iterative experiments show almost 100% accuracy in this case.

For small tilted images with no rotation, our algorithm still works fine, but less accuracy than the first case.

For images with large tilt and rotation, the accuracy is hampered with up to some percent result being inaccurate.

# REFERENCES

1. M. Kamel and A. Zhao, “Extraction of binary character/graphics images from grayscale document images,” Graph. Models Image Process, 203–217, 1993.
2. T. Abak, U. Baris¸, and B. Sankur, “The performance of thresholding algorithms for optical character recognition,” Intl. Conf. Document Anal. Recog. ICDAR’97, pp. 697–700, 1997.
3. O. D. Trier and A. K. Jain, ‘‘Goal-directed evaluation of binarization methods,’’ IEEE Trans. Pattern Anal. Mach. Intell. PAMI-17, 1191– 1201, 1995
4. B. Bhanu, ‘‘Automatic target recognition: state of the art survey,’’ IEEE Trans. Aerosp. Electron. Syst. AES-22, 364–379, 1986.
5. Sachin A. Murab, Vaishali. M.Deshmukh “An Empirical Study of Signature Recognition & Verification System Using Various Approaches” International Journal of Engineering and Advanced Technology (IJEAT)ISSN: 2249–8958, Volume-2.
6. Debasish Jena, Banshidhar Majhi and Sanjay Kumar Jena “Improved Offline Signature Verification Scheme Using Feature Point Extraction Method” Journal of Computer Science 4 (2): 111-116, 2008.
7. M. Sezgin and R. Tasaltin, “A new dichotomization technique to multilevel thresholding devoted to inspection applications,” Pattern Recogn. Lett. 21, 151–161, 2000.
8. Barbara, Zitova, J., Flusser, “Image Registration methods: a survey”, Image and Vision Computing, Vol. 21, No. 11, pp. 977-1000, 2003.
9. J. C. Russ, “Automatic discrimination of features in gray-scale images,” J. Microsc, 263–277, 1987.
10. Canny, J., A Computational Approach To Edge Detection, IEEE Trans. Pattern Analysis and Machine Intelligence, 8(6):679–698, 1986.

11. D. Ziou and S. Tabbone (1998) "Edge detection techniques: An overview", International Journal of Pattern Recognition and Image Analysis, 8(4):537–559, 1998.

12. "D. Boulfelfel, R.M. Rangayyan, L.J. Hahn, and R. Kloiber, 1994, "Three-dimensional restoration of single photon emission computed tomography images", IEEE Transactions on Nuclear Science, 41(5): 1746-1754, October 1994."

13. P.B.; Ramakrishnan, A.G. (1987-05-29). Word Level Multi-script Identification.Pattern Recognition Letters, Vol. 29, pp. 1218 - 1229, 2008

14. Li, Yuhai, L. Jian, T. Jinwen, X. Honbo. “A fast rotated template matching based on point feature.” Proceedings of the SPIE 6043 (2005): 453-459.

15. R. C. Gonzalez, R. E. Woods, “Digital Image Processing” Second Edition, Pearson Education.

16. B. Sirmacek, C. Unsalan. “Urban Area and Building Detection Using SIFT Keypoints and Graph Theory”, IEEE Transactions on Geoscience and Remote Sensing, Vol.47 (4), pp. 1156-1167, April 2009.

17. R. C. Gonzalez, R. E. Woods, S. L. Eddins “Digital Image Processing Using MATLAB”, Pearson Education, 2004.

18. K. Jain “Fundamentals of Digital Image Processing” PHI.

19. Ashish Arora, GUI Based OMR System for Recognition of filled Bubbles in Scanned OMR sheets in Absence of OMR Machine, Course Project at IIT, 30-38, 2011.

20. Richard Szeliski, Image Alignment and Stitching: A Tutorial, December 10, 2006.