A

PROJECT REPORT ON

FACE RECOGNISATION ACROSS NON- UNIFORM MOTION BLUR, ILLUMINATION AND POSE

Submitted to JNT University for the partial fulfillment of the requirements for the award of the degree of

Bachelor of Technology

In

ELECTRONICS AND COMMUNICATION ENGINEERING

By

K. VYANJAN KUMAR	**22L25A0410**
G. VENKATA TARUN KUMAR	**22L25A0408**
C. VENKATESWARLU	**21L21A0450**
M. DHANESWAR REDDY	**21L21A0465**
M. SURYA NARAYANA	**21L21A0467**
Y. SRIKANTH	**22L25A0419**

Under the guidance of

Mr.D.VENKATA RAMI REDDY, M.Tech

Asst. Professor,

Department of ECE

Department of Electronics and Communication Engineering

VAAGDEVI INSTITUTE OF TECHNOLOGY & SCIENCE

(Approved by A.I.C.T.E., New Delhi, Affiliated to JNTUA, Anantapuramu)

(Accredited by NAAC with B^+ Grade)

Peddasettipalli, Proddatur

Kadapa, A.P.-516360

2021-2025

Sri Sri Sri Mookambika Educational Society's
VAAGDEVI INSTITUTE OF TECHNOLOGY & SCIENCE
(Approved by A.I.C.T.E, New Delhi, Affiliated to JNTUA, Anantapuramu)
(Accrediated by NAAC with Grade B+) Peddasettipalle, Proddatur - 516360

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CERTIFICATE

This is to certify that the project work entitled **"FACE RECOGNISATION ACROSS NON- UNIFORM MOTION BLUR, ILLUMINATION AND POSE"** is a bonafide record submitted by **K.VYANJAN KUMAR (22L25A0410), G.VENKATA TARUN KUMAR (22L25A0408), C.VENKATESWARLU (21L21A0450), M.DHANESWAR REDDY (21L21A0465), M.SURYA NARAYANA (21L21A0467), Y.SRIKANTH (22L25A0419)** in partial fulfillment for the award of the degree of Bachelor of Technology in **"Electronics and Communication Engineering"** for the year **2021-2025**, the work reported here in does not form a part of any other thesis on which a degree has been awarded earlier.

This is to further certify that they have worked for a period of one semester preparing their work under our supervision and guidance.

PROJECT GUIDE	**HEAD OF THE DEPARTMENT**
Mr.D.VENKATA RAMI REDDY, M.Tech.	**Mr.S.M.K.SUKUMAR REDDY, M.Tech.(Ph.D).**
Assistant Professor,	Associate Professor,
Department of ECE.	Department of ECE.

External project viva-voice held on: ____________________

INTERNAL EXAMINER	**EXTERNAL EXAMINER**

# Acknowledgement

An endeavor of a long period can be successful only with the advice of many well-wishers. We take this opportunity to express our deep gratitude and appreciation to all those who encouraged us for the successful completion of the main project work.

We are thankful to our guide, ***Mr.D.VENKATA RAMI REDDY,*** ***M.Tech,*** Assistant Professor in the Department of E.C.E., Vaagdevi Institute of Technology and Science, Proddatur, for his valuable guidance and suggestions in analyzing and testing until the end of the main project work completion.

Our special thanks to ***Mr.S.M.K.SUKUMAR REDDY,*** ***M.Tech,(Ph.D),*** Head of Electronics & Communication Engineering Department, Vaagdevi Institute of Technology & Science, Proddatur. During the progress of work, for his timely suggestions and help despite his busy schedule.

We wish to convey our gratitude and express sincere thanks to all P.R.C (Project Review Committee) members for their support and cooperation rendered for the successful submission of our project work.

We wish to express our gratitude to ***Dr.B.SIDDESWARA RAO,*** ***Ph.D,*** Principal of Vaagdevi Institute of Technology and Science, Proddatur, for his consistent help and encouragement in completing the project work.

We are very much thankful to ***Sri G.HUSSAIN REDDY,*** Honorable Secretary of Mookambika Educational Trust, for his consistent help in providing good facilities in our college.

Whatever we are at present is due to the blessings of our Great Father and mother. Last not least, we express our thanks to all faculties of VITS for assistance during the facilities in the laboratory, our well-wisher friends, and lab technicians who encouraged us during this project work successfully.

PROJECT ASSOCIATES

K. VYANJAN KUMAR	22L25A0410
G. VENKATA TARUN KUMAR	22L25A0408
C. VENKATESWARLU	21L21A0450
M. DHANESWAR REDDY	21L21A0465
M. SURYA NARAYANA	21L21A0467
Y. SRIKANTH	22L25A0419

ABSTRACT

This project mainly addresses the building of face recognition system across Non-Uniform Motion Blur, Illumination and Pose by using Principal Component Analysis (PCA) or Hotelling Transform (HT). PCA is a statistical approach used for reducing the number of variables in face recognition. In PCA, every image in the training set is represented as a linear combination of weighted eigenvectors called eigenfaces. These eigenvectors are obtained from covariance matrix of a training image set. The weights are found out after selecting a set of most relevant Eigenfaces. Recognition is performed by projecting a test image onto the subspace spanned by the eigenfaces and then classification is done by measuring minimum Euclidean distance. A number of experiments were done to evaluate the performance of the face recognition system.

Non-uniform motion blur, caused by irregular camera movements or object motion, is modeled using transformation spread functions (TSF) that adaptively correct distortions in facial features. Illumination variations are mitigated through preprocessing techniques that normalize lighting conditions or adapt recognition algorithms to dynamic environments. Pose differences are addressed by incorporating PCA's adaptability to extract invariant features from various facial orientations. Additionally, the integration of PCA with machine learning algorithms, such as Support Vector Machines (SVM), enhances classification accuracy.

This robust framework leverages MATLAB's Image Processing and Computer Vision Toolboxes, enabling real-time implementation of face recognition systems in dynamic environments. Future advancements in hybrid algorithms combining PCA with deep learning promise further improvements in recognition reliability across diverse datasets. Applications span multiple domains, including security systems, mobile authentication, and personalized user experiences.

In this project, we used a training database image of students and faculty of Electronics and Communication Engineering department, Batch 2021-2025, Vaagdevi Institute of Technology and Science, Proddatur and some other person database for testing of the accuracy of the project.

CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABBREVIATION	EXPANSION
MATLAB	Matrix Laboratory
TIFF	Tagged Image File Format
BMP	Bitmap Image
TSF	Transformation Spread Function
PCA	Principal Component Analysis
LDA	Linear Discriminant Analysis
ICA	Independent Component Analysis
SVM	Support Vector Machines
LBP	Local Binary Pattern
ASM	Active Shape Models
HT	Hotelling Transform
KLT	Karhunen Loeve Transform
EVD	Eigen Values Decomposition
SVD	Singular Values Decomposition
2D	2 Dimensional
1D	1 Dimensional
JPEG/JPG	Joint Photographic Experts Group
CNN	Convolutional Neural Network

LIST OF TABLES

CHAPTER-1

INTRODUCTION

1.1 INTRODUCTION TO MATLAB

1.1.1 Matrix Laboratory (MATLAB)

MATLAB, short for Matrix Laboratory, is a high-performance programming environment developed by MathWorks. It is designed primarily for technical computing, offering tools for numerical computation, visualization, and programming. MATLAB is widely used in fields such as engineering, science, mathematics, and data analysis due to its intuitive environment and robust function.

As of 2025, MATLAB has more than four million users worldwide. MATLAB continues to be widely used in colleges and universities globally as of 2025. It is employed in over **6500 universities worldwide** supporting teaching and research across various technical disciplines. MATLAB's popularity stems from its versatility in fields like engineering, data science, and computational biology. Many institutions offer campus-wide licenses, enabling students and faculty to access MATLAB for academic purposes.

1.1.2 History and Background

MATLAB was invented by mathematician and computer programmer Cleve Moler. The idea for MATLAB was based on his 1960s PhD thesis. Moler became a math professor at the University of New Mexico and started developing MATLAB for his students as a hobby.

He developed MATLAB's initial linear algebra programming in 1967 with his one-time thesis advisor, George Forsythe. This was followed by Fortran code for linear equations in 1971.

Before version 1.0, MATLAB "was not a programming language; it was a simple interactive matrix calculator. There were no programs, no toolboxes, no graphics. And no ODEs or FFTs."

The first early version of MATLAB was completed in the late 1970s. The software was disclosed to the public for the first time in February 1979 at the Naval Postgraduate School in California. Early versions of MATLAB were simple matrix calculators with 71 pre-built functions. At the time, MATLAB was distributed for free to universities. Moler would leave copies at universities he visited, and the software developed a strong following in the math departments of university campuses.

In the 1980s, Cleve Moler met John N. Little. They decided to reprogram MATLAB in C and market it for the IBM desktops that were replacing mainframe computers at the time.

MATLAB was first released as a commercial product in 1984 at the Automatic Control Conference in Las Vegas. MathWorks, Inc. was founded to develop the software, and the MATLAB programming language was released. The first MATLAB sale was the following year, when Nick Trefethen from the Massachusetts Institute of Technology bought ten copies.

By the end of the 1980s, several hundred copies of MATLAB had been sold to universities for student use.] The software was popularized largely thanks to toolboxes created by experts in various fields for performing specialized mathematical tasks. Many of the toolboxes were developed because of Stanford students that used MATLAB in academia, then brought the software with them to the private sector.

Over time, MATLAB was re-written for early operating systems created by Digital Equipment Corporation, VAX, Sun Microsystems, and for Unix PCs. Version 3 was released in 1987. The first MATLAB compiler was developed by Stephen C. Johnson in the 1990s.

In 2000, MathWorks added a Fortran-based library for linear algebra in MATLAB 6, replacing the software's original LINPACK and EISPACK subroutines that were in C. MATLAB's Parallel Computing Toolbox was released at the 2004 Supercomputing Conference and support for graphics processing units (GPUs) was added to it in 2010.

Some especially large changes to the software were made with version 8 in 2012. The user interface was reworked, and Simulink's functionality was expanded. By 2016, MATLAB had introduced several technical and user interface improvements, including the MATLAB Live Editor notebook, and other features.

1.1.3 Software Requirement

MATLAB Software is a powerful software environment used for numerical computing, programming, and visualization. It provides an extensive library of built-in functions and toolboxes for various applications, including signal processing, image analysis, and machine learning. With its user-friendly interface and versatile scripting language, MATLAB is widely used in academia, engineering, and research. One of its core strengths is its ability to handle matrix and vector calculations seamlessly. Users can create stunning plots and graphs for data visualization, making it easier to interpret complex datasets. Additionally, MATLAB supports integration with hardware for real-time simulations and control applications. Its adaptability and vast functionality make it an indispensable tool in the scientific and technical community.

Key Features of MATLAB

MATLAB stands out from other programming environments due to its unique combination of simplicity and power. Here are its core features.

a. Matrix and Array-Centric Computing

MATLAB is built around matrices and arrays, making it ideal for linear algebra, numerical analysis, and other computations involving matrix operations. For example:

```
%Matrix Example
A = [1 2; 3 4]; % Create a matrix
B = A * A;      % Matrix multiplication
```

b. Comprehensive Toolboxes

MATLAB offers specialized toolboxes collections of functions designed for specific domains such as:

- Signal Processing Toolbox
- Image Processing Toolbox
- Machine Learning Toolbox
- Control Systems Toolbox

c. Visualization Tools

MATLAB excels in data visualization, enabling users to plot 2D and 3D graphs, build interactive dashboards, and create animations. Visualizing data is as simple as:

```
% Visualization Example
x = 0:0.01:10;
y = sin(x);
plot(x, y);
```

d. Programming and Scripting

MATLAB supports procedural, object-oriented, and functional programming paradigms. Users can write scripts, develop custom functions, and design algorithms efficiently:

```
% Script Example
x = [1, 2, 3, 4, 5];
y = x.^2; % Square each element of x
plot(x, y); % Plot the results
```

e. App Design

MATLAB includes App Designer, which allows users to create professional applications using a drag-and-drop interface.

f. Integration and Interoperability

MATLAB can interface with languages such as Python, C++, and Java, making it easy to incorporate into diverse workflows.

g. Toolboxes

MATLAB offers specialized toolboxes tailored for specific applications like image processing, machine learning, deep learning, optimization, signal processing, and more.

1.1.4 MATLAB Environment

MATLAB's environment is user-friendly and interactive, consisting of the following components:

a. Command Window

The Command Window lets users execute commands immediately. It is ideal for testing snippets of code and running simple calculations as shown in figure 1.1.1.

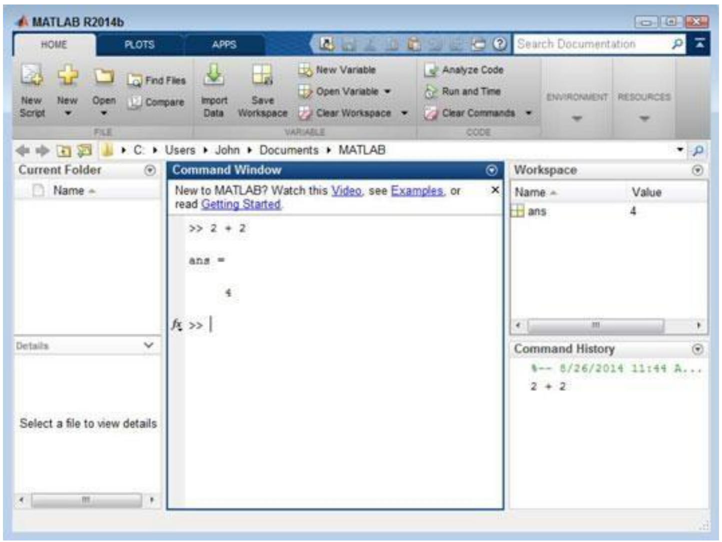


Figure 1.1.1 Command Window

b. Editor

The MATLAB Editor is an integrated tool within MATLAB's development environment that is used for writing, editing, and debugging scripts, functions, and other files. It provides an intuitive interface with a variety of features to make coding and debugging easier for users

The Editor is used for writing, editing, and saving scripts. Scripts are reusable pieces of code that simplify repetitive tasks as shown in figure 1.1.2.

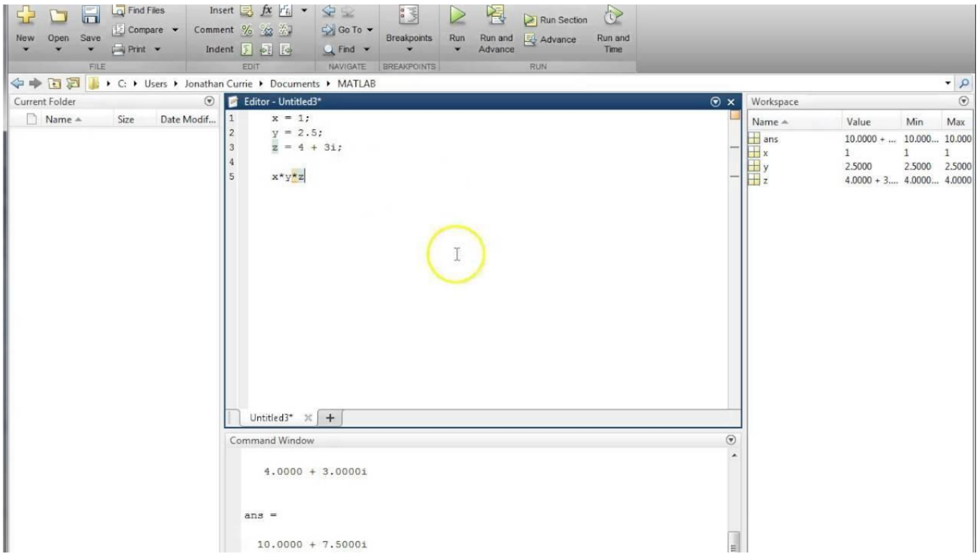
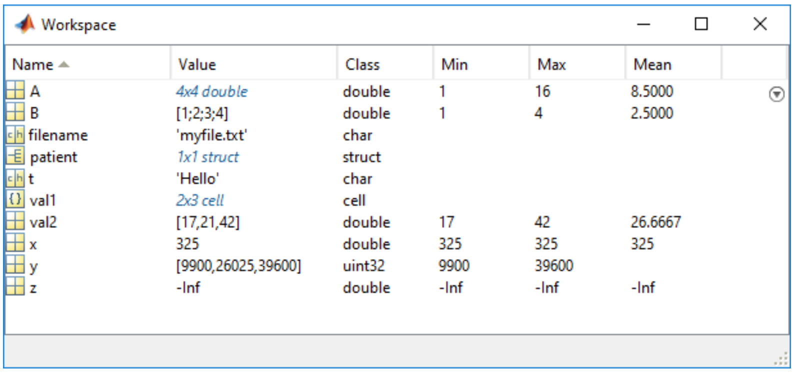


Figure 1.1.2 Editor Window

c. Workspace

The Workspace shows active variables, allowing users to monitor their data during Execution as shown in figure 1.1.3.



Name	Value	Class	Min	Max	Mean
A	4x4 double	double	1	16	8.5000
B	[1;2;3;4]	double	1	4	2.5000
filename	'myfile.txt'	char			
patient	1x1 struct	struct			
t	'Hello'	char			
val1	2x3 cell	cell			
val2	[17,21,42]	double	17	42	26.6667
x	325	double	325	325	325
y	[9900,26025,39600]	uint32	9900	39600	
z	-Inf	double	-Inf	-Inf	-Inf

Figure 1.1.3 Workspace Window

d. Figure Window

The Figure Window displays visualizations, such as graphs, plots, and images shown in figure 1.1.4.

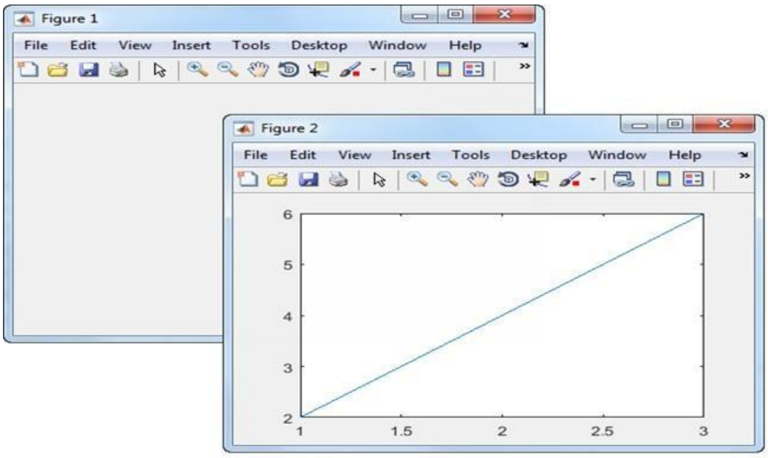


Figure 1.1.4 Figure Window

e. Toolboxes

Toolboxes extend MATLAB's functionality into specific areas of expertise as shown in figure 1.1.5

Figure 1.1.5 Toolbox Window

1.1.5 Getting Started with MATLAB

For beginners, getting started involves

a. Installing MATLAB: Obtain a license from MathWorks or your institution.

b. Learning the Basics: Start with commands like:

```
% Example
disp('Hello, World!'); % Display text
x = [1, 2, 3];         % Create an array
```

c. Exploring Documentation: MATLAB's documentation is comprehensive and beginner friendly.

1.1.6 Applications

MATLAB is utilized in numerous fields, showcasing its versatility:

a. Engineering:

Engineers use MATLAB to simulate systems, analyze designs, and optimize performance. Examples include:

- Designing control systems for autonomous vehicles.
- Simulating electrical circuits.

b. Data Science:

MATLAB provides tools for big data analysis, including data cleaning, visualization, and predictive modelling.

c. Scientific Research:

MATLAB helps scientists simulate experiments, solve equations, and analyse results.

d. Machine Learning and AI:

MATLAB supports deep learning, reinforcement learning, and model training, enabling researchers to work on cutting-edge AI projects.

e. Finance:

Financial analysts use MATLAB for portfolio optimization, risk analysis, and forecasting.

1.2 INTRODUCTION TO IMAGE PROCESSING

1.2.1 Image Processing

Image Processing is a technique to enhance raw images received from cameras/sensors placed on satellites, space probes and aircraft or pictures taken in normal day-to- day life for various applications.

Various techniques have been developed in Image Processing during the last four to five decades. Most of the techniques are developed for enhancing images obtained from unmanned spacecrafts, space probes and military reconnaissance flights. Image Processing systems are becoming popular due to easy availability of powerful personnel computers, large size memory devices, graphics software's etc.

Image Processing is used in various applications such as:

- Remote Sensing
- Medical Imaging
- Non-destructive Evaluation
- Forensic Studies
- Textiles
- Material Science.
- Military
- Film industry
- Document processing
- Graphic arts
- Printing Industry

The common steps in image processing are image scanning, storing, enhancing and interpretation. The schematic diagram of image scanner-digitizer diagram is shown in figure 1.2.1.

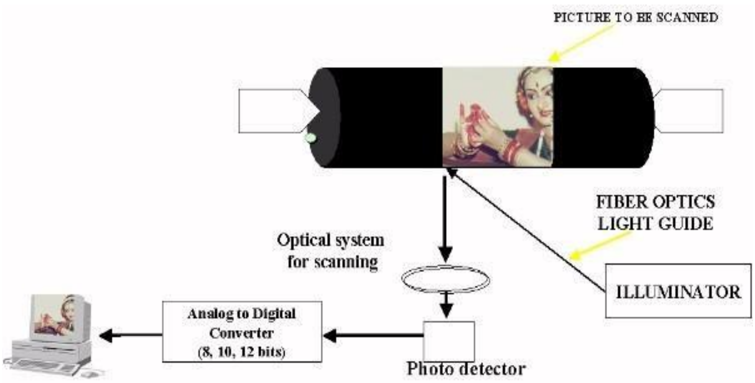


Figure 1.2.1 Image Processing

1.2.2 Methods of Image Processing

There are two methods available in Image Processing.

a. Analog Image Processing

Analog Image Processing refers to the alteration of image through electrical means. The most common example is the television image.

The television signal is a voltage level which varies in amplitude to represent brightness through the image. By electrically varying the signal, the displayed image appearance is altered. The brightness and contrast controls on a TV set serve to adjust the amplitude and reference of the video signal, resulting in the brightening, darkening and alteration of the brightness range of the displayed image.

b. Digital Image Processing

In this case, digital computers are used to process the image. The image will be converted to digital form using a scanner – digitizer (as shown in Figure 1) and then process it. It is defined as subjecting numerical representations of objects to a series of operations in order to obtain a desired result. It starts with one image and produces a modified version of the same. It is therefore a process that takes an image into another.

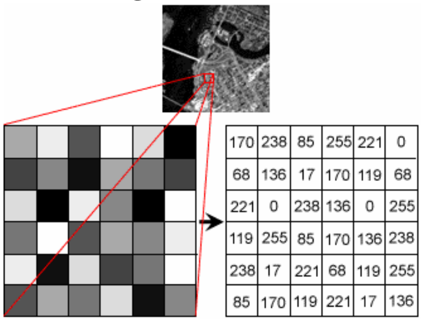
The term *digital image processing* generally refers to processing of a two-dimensional picture by a digital computer. In a broader context, it implies digital processing of any two-dimensional data. A digital image is an array of real numbers represented by a finite number of bits.

The various Image Processing techniques are:

a. Image representation
b. Image preprocessing
c. Image enhancement
d. Image analysis
e. Image segmentation
f. Image classification
g. Image restoration
h. Image reconstruction
i. Image data compression
j. Image conversion

a. Image Representation

An image defined in the "real world" is considered to be a function of two real variables, for example, $f(x,y)$ with f as the amplitude (e.g. brightness) of the image at the *real* coordinate position (x,y). The effect of digitization is shown in figure 1.2.2**.**



170	238	85	255	221	0
68	136	17	170	119	68
221	0	238	136	0	255
119	255	85	170	136	238
238	17	221	68	119	255
85	170	119	221	17	136

Figure 1.2.2 Image Representation

The 2D continuous image $f(x,y)$ is divide into N *rows* and M *columns*. The intersection of a row and a column is called as *pixel*. The value assigned to the integer coordinates $[m,n]$ with $\{m=0,1, 2,...,M-1\}$ and $\{n=0,1,2,...,N-1\}$ is $f[m,n]$. In fact, in most cases $f(x,y)$--which we might consider to be the physical signal that impinges on the face of a sensor. Typically, an image file such as BMP, JPEG, TIFF etc., has some header and picture information.

b. Image Preprocessing

Scaling

The theme of the technique of *magnification* is to have a closer view by magnifying or zooming the interested part in the imagery. By reduction, we can bring the unmanageable size of data to a manageable limit. For resampling an image Nearest Neighborhood, Linear, or cubic convolution techniques are used.

i. Magnification

This is usually done to improve the scale of display for visual interpretation or sometimes to match the scale of one image to another. To *magnify* an image by a factor of 2, each pixel of the original image is replaced by a block of 2x2 pixels, all with the same brightness value as the original pixel image as shown in figure 1.2.3.

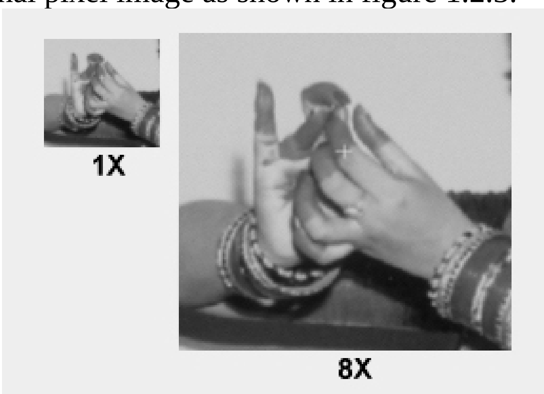


Figure 1.2.3 Image Magnification

ii. Reduction

To *reduce* a digital image to the original data, every mth row and mth column of the original imagery is selected and displayed. Another way of accomplishing the same is by taking the average in 'm x m' block and displaying this average after proper rounding of the resultant value.

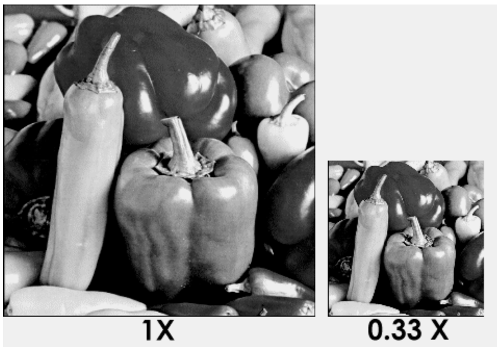


Figure 1.2.4 Image Reduction

iii. Rotation

Rotation is used in image mosaic, image registration etc. One of the techniques of rotation is 3-pass shear rotation, where rotation matrix can be decomposed into three separable matrices image as shown in figure 1.2.5: <u>3-pass shear rotation</u>

$$R = \begin{vmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{vmatrix} = \begin{vmatrix} 1 & -\tan\alpha/2 \\ 0 & 1 \end{vmatrix} \begin{vmatrix} 1 & 0 \\ \sin\alpha & 1 \end{vmatrix} \begin{vmatrix} 1 & -\tan\alpha/2 \\ 0 & 1 \end{vmatrix}$$

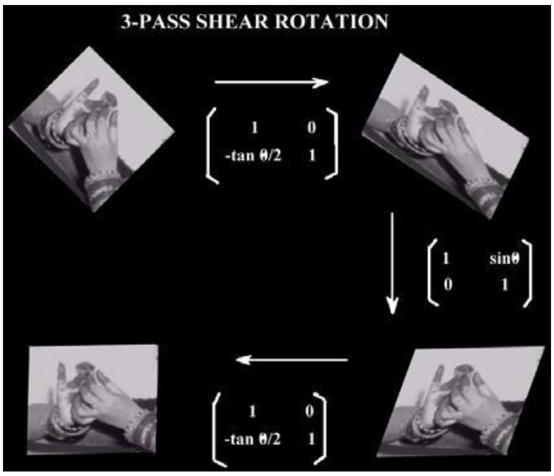


Figure 1.2.5 3-Pass Shear Rotation

Advantages

- No scaling – no associated resampling degradations.
- Shear can be implemented very efficiently

iv. Mosaic

Mosaic is a process of combining two or more images to form a single large image without radiometric imbalance. Mosaic is required to get the synoptic view of the entire area, otherwise capture as small images image as shown in figure 1.2.6

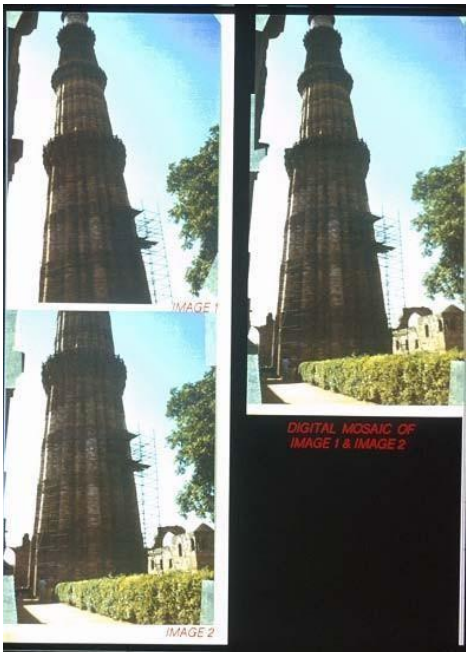


Figure 1.2.6 Image Mosaicking

c. Image Enhancement Techniques

Sometimes images obtained from satellites and conventional and digital cameras lack in contrast and brightness because of the limitations of imaging sub systems and illumination conditions while capturing image. Images may have different types of noise. In *image enhancement*, the goal is to accentuate certain image features for subsequent analysis or for image display [1,2]. Examples include contrast and edge enhancement, pseudo-colouring, noise filtering, sharpening, and magnifying. Image enhancement is useful in feature extraction, image analysis and an image display. The enhancement process itself does not increase the inherent information content in the data. It simply emphasizes certain specified image characteristics. Enhancement algorithms are generally interactive and application- dependent.

Some of the *enhancement techniques* are:

i. Contrast Stretching
ii. Noise Filtering
iii. Histogram modification

i. Contrast Stretching:

Some images (eg. over water bodies, deserts, dense forests, snow, clouds and under hazy conditions over heterogeneous regions) are homogeneous i.e., they do not have much change in their levels. In terms of histogram representation, they are characterized as the occurrence of very narrow peaks. The homogeneity can also be due to the incorrect illumination of the scene image as shown in figure 1.2.7.

Ultimately the images hence obtained are not easily interpretable due to poor human perceptibility. This is because there exists only a narrow range of gray-levels in the image having provision for wider range of gray-levels. The *contrast stretching* methods are designed exclusively for frequently encountered situations. Different stretching techniques have been developed to stretch the narrow range to the whole of the available dynamic range.

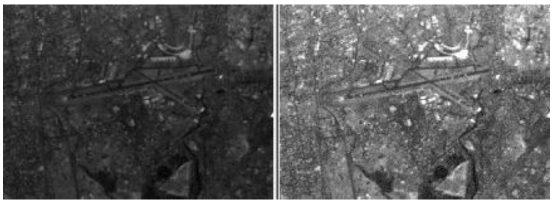


Figure 1.2.7 contrast stretching

ii. Noise Filtering

Noise filtering is used to filter the unnecessary information from an image. It is also used to remove various types of noises from the images. Mostly this feature is interactive. Various filters like low pass, high pass, mean, median etc., are available image as shown in figure 1.2.8 and figure 1.2.9.

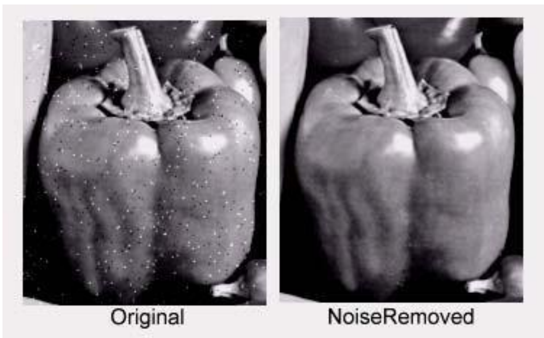


Figure 1.2.8 Noise Removal

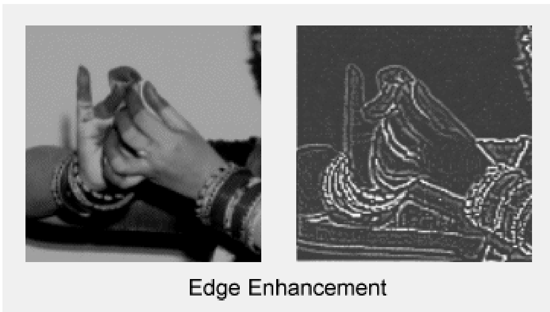


Figure 1.2.9 Edge Enhancement

iii. Histogram Modification

Histogram has a lot of importance in image enhancement. It reflects the characteristics of image. By modifying the histogram, image characteristics can be modified. One such example is Histogram Equalization image as shown in figure 1.2.10.

Histogram equalization is a nonlinear stretch that redistributes pixel values so that there is approximately the same number of pixels with each value within a range. The result approximates a flat histogram. Therefore, contrast is increased at the peaks and lessened at the tails.

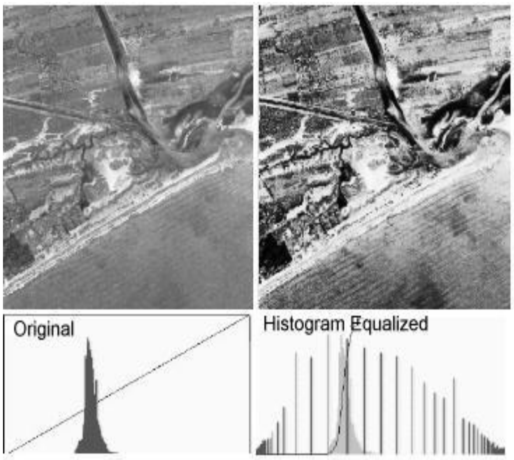


Figure 1.2.10 Histogram equalized output

d. Image Analysis

Image analysis is concerned with making quantitative measurements from an image to produce a description of it. In the simplest form, this task could be reading a label on a grocery item, sorting different parts on an assembly line, or measuring the size and orientation of blood cells in a medical image. More advanced image analysis systems measure quantitative information and use it to make a sophisticated decision, such as controlling the arm of a robot to move an object after identifying it or navigating an aircraft with the aid of images acquired along its trajectory.

Image analysis techniques require extraction of certain features that aid in the identification of the object. Segmentation techniques are used to isolate the desired object from the scene so that measurements can be made on it subsequently. Quantitative measurements of object features allow classification and description of the image.

e. Image Segmentation

Image segmentation is the process that subdivides an image into its constituent parts or objects. The level to which this subdivision is carried out depends on the problem being solved, i.e., the segmentation should stop when the objects of interest in an application have been isolated e.g., in autonomous air-to- ground target acquisition, suppose our interest lies in identifying vehicles on a road, the first step is to segment the road from the image and then to segment the contents of the road down to potential vehicles. Image thresholding techniques are used for image segmentation.

f. Image Classification

Classification is the labeling of a pixel, or a group of pixels based on its grey value. Classification is one of the most often used methods of information extraction. In Classification, usually multiple features are used for a set of pixels i.e., many images of a particular object are needed. In Remote Sensing area, this procedure assumes that the imagery of a specific geographic area is collected in multiple regions of the electromagnetic spectrum and that the images are in good registration. Most of the information extraction techniques rely on analysis of the spectral reflectance properties of such imagery and employ special algorithms designed to perform various types of 'spectral analysis'. The process of multispectral classification can be performed using either of the two methods: Supervised or Unsupervised.

In *Supervised classification*, the identity and location of some of the land cover types such as urban, wetland, forest etc., are known as priori through a combination of field works and toposheets. The analyst attempts to locate specific sites in the remotely sensed data that represents homogeneous examples of these land cover types. These areas are commonly referred as TRAINING SITES because the spectral characteristics of these known areas are used to 'train' the classification algorithm for eventual land cover mapping of reminder of the image. Multivariate statistical parameters are calculated for each training site. Every pixel both within and outside these training sites is then evaluated and assigned to a class of which it has the highest likelihood of being a member image as shown in figure 1.2.11.

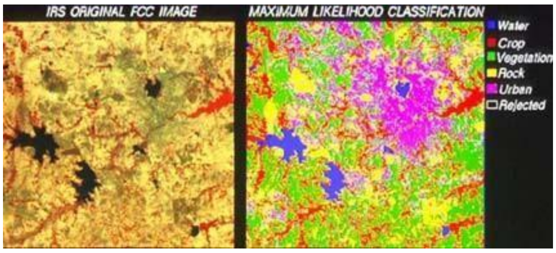


Figure 1.2.11. Image Classification

In an *Unsupervised classification*, the identities of land cover types has to be specified as classes within a scene are not generally known as priori because ground truth is lacking or surface features within the scene are not well defined. The computer is required to group pixel data into different spectral classes according to some statistically determined criteria.

The comparison in medical area is the labeling of cells based on their shape, size, color and texture, which act as features. This method is also useful for MRI images.

g. Image Restoration

Image restoration refers to removal or minimization of degradations in an image. This includes de-blurring of images degraded by the limitations of a sensor or its environment, noise filtering, and correction of geometric distortion or non-linearity due to sensors.

Image is restored to its original quality by inverting the physical degradation phenomenon such as defocus, linear motion, atmospheric degradation and additive noise image as shown in figure 1.2.12

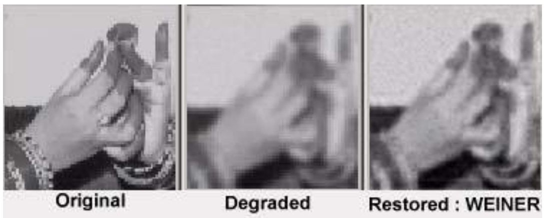


Figure 1.2.12 Weiner – Image Restoration

h. Image Reconstruction from Projections

Image reconstruction from projections is a special class of image restoration problems where a two- (or higher) dimensional object is reconstructed from several one-dimensional projections. Each projection is obtained by projecting a parallel X-ray (or other penetrating radiation) beam through the object. Planar projections are thus obtained by viewing the object from many different angles. Reconstruction algorithms derive an image of a thin axial slice of the object, giving an inside view otherwise unobtainable without performing extensive surgery. Such techniques are important in medical imaging (CT scanners), astronomy, radar imaging, geological exploration, and non-destructive testing of assemblies image as shown in figure 1.2.13.

Figure 1.2.13 MRI Slices

i. Image Compression

Compression is a very essential tool for archiving image data, image data transfer on the network etc. They are various techniques available for lossy and lossless compressions. One of most popular compression techniques. JPEG (Joint Photographic Experts Group) uses Discrete Cosine Transformation (DCT) based compression technique. Currently wavelet-based compression techniques are used for higher compression ratios with minimal loss of data image as shown in figure 1.2.14.

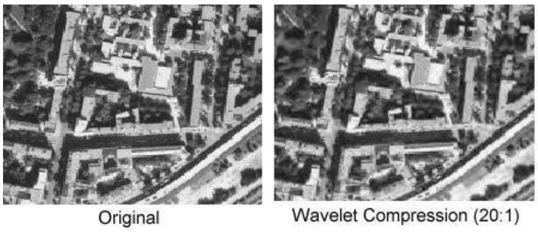


Figure 1.2.14. Wavelet Image Compression

j. Image Conversion

Image conversion in MATLAB involves processing and transforming images, typically to change their format, colour space, or resolution, among other possibilities image as shown in figure 1.2.15

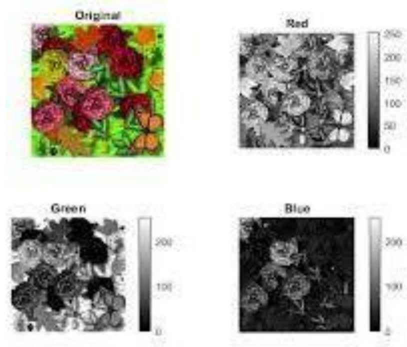


Figure 1.2.15. Image Conversion

1.3 INTRODUCTION TO FACE RECOGNISATION ACROSS NON-UNIFORM MOTION BLUR, ILLUMINATION AND POSE

1.3.1 Objective

The objective of "Introduction to Face Recognition Across Non-Uniform Motion Blur, Illumination, and Pose" is to develop robust algorithms for face recognition in challenging conditions. These conditions include non-uniform motion blur caused by camera shake, varying illumination levels, and changes in pose. The research aims to address the limitations of existing methods, which struggle with these real-world scenarios. By modelling blurred faces as geometrically transformed versions of sharp images and incorporating techniques to handle illumination and pose variations, the study seeks to improve the accuracy and reliability of face recognition systems**.**

1.3.2 Challenges in Face Recognition

a. Motion Blur:

Non-uniform motion blur occurs when the camera or subject moves unpredictably, leading to distortions that vary across the image. This type of blur is particularly challenging because traditional algorithms often assume uniform blur and struggle to handle space-varying distortions**.**

b. Illumination Variations:

Changes in lighting can significantly affect the appearance of facial features, making it difficult for recognition systems to match faces accurately. Shadows, highlights, and varying light sources can alter the perceived structure of a face.

c. Pose Variations:

Differences in facial orientation, such as tilts or rotations, can lead to mismatches between the stored facial data and the captured image. Pose variations are common in unconstrained environments, such as surveillance footage or casual photography.

1.3.3 Solutions and Methodologies

Researchers have developed advanced algorithms to tackle these challenges. For instance:

a. Transformation Spread Function (TSF):

This technique models non-uniform motion blur by representing blurred faces as a combination of geometrically transformed versions of the original sharp image.

b. Sparse Camera Trajectory:

By assuming a sparse trajectory in camera motion space, algorithms can estimate and correct for motion blur.

c. Bi-Convex Sets:

The set of images obtained from a face by non-uniform blurring and illumination changes forms a bi-convex set, which can be exploited for robust recognition.

d. Multi-Scale Implementation:

Efficient computation and memory usage are achieved by implementing recognition algorithms at multiple scales.

CHAPTER-2

EXISTING METHOD

2.1 METHODOLOGY

Different types of method are implemented for Face recognition across non-uniform motion, illumination and pose. They are:

- Linear Discriminant Analysis (LDA)
- Independent Component Analysis (ICA)
- Support Vector Machines (SVMs)
- Local Binary Pattern (LBP) Features
- Active Shape Models (ASM)

2.2 TYPES OF METHOD

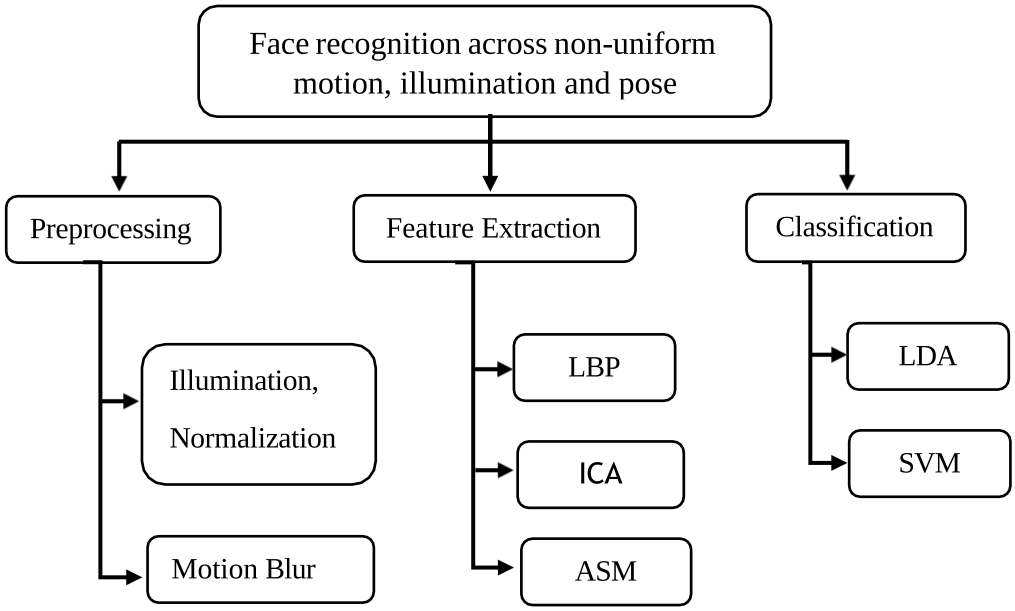


Figure 2.1 Types of Method

Face recognition across non-uniform motion, illumination and pose are used different methods based on the characteristics. They are:

- Preprocessing
- Feature Extraction
- Classification

2.2.1 Preprocessing:

This is the initial stage in face recognition, aimed at preparing images by reducing noise and addressing challenges like motion blur, illumination variations, and pose differences.

Steps Involved:

i. Motion Blur Handling: Using filters (e.g., Gaussian or Wiener) or deblurring algorithms to correct blurred images.

ii. Illumination Normalization: Applying histogram equalization, gamma correction, or Retinex theory to standardize lighting.

iii. Pose Alignment: Using techniques like geometric transformations or 3D modelling to align faces to a canonical pose.

Merits:

- Enhances image quality for accurate recognition.
- Helps reduce variations caused by external factors (blur, light, pose).
- Creates a uniform input format for feature extraction.

Demerits:

- Computationally expensive for large datasets.
- Some preprocessing techniques may alter useful information in the image.
- Complex challenges like extreme blur or lighting cannot always be fully resolved.

2.2.2 Feature Extraction:

This stage identifies and captures the most relevant characteristics of a face, like texture, shape, or patterns, to represent it numerically.

Common Techniques:

i. Local Binary Patterns (LBP):

Local Binary Patterns (LBP) are a powerful and simple texture operator that has found widespread use in various image processing and computer vision applications.

Merits:

- Robustness to Illumination Changes
- Computational Simplicity
- Discriminative Power
- Effective Texture Description

Demerits:

- Sensitivity to Noise
- Limited Global Information
- Rotation Sensitivity
- Grayscale Limitations

ii. Independent Component Analysis (ICA):

Independent Component Analysis (ICA) is a statistical and computational technique used to separate multivariate signals into additive subcomponents, assuming that these subcomponents are statistically independent.

Merits:

- Blind Source Separation
- Identification of Independent Source

- Applications in Various Fields
- Non-Gaussian Utilization

Demerits:

- Assumption of Independence
- Non-Gaussian Requirement
- Computational Complexity
- Sensitivity to Noise
- Linearity Assumption
- Order and Scaling Ambiguity

iii. Active Shape Models (ASM):

Active Shape Models (ASMs) are a powerful technique in computer vision for object localization and segmentation. They use statistical models of object shapes to find and outline objects in images.

Merits:

- Shape Constraints
- Statistical Modelling
- Robustness
- Effective Object Localization

Demerits:

- Dependency on Training Data
- Landmark Placement
- Initialization Sensitivity
- Computational Cost
- Handling of large shape variations
- Topology Limitations

2.2.3 Classification

This is the final stage where the extracted features are fed into machine learning models to classify and identify faces**.**

Common Techniques:

i. Linear Discriminant Analysis (LDA):

Linear Discriminant Analysis (LDA) is a powerful and widely used technique, particularly in the field of supervised machine learning for classification and dimensionality reduction.

Merits:

- Dimensionality Reduction

- Classification
- Handling Multicollinearity
- Interpretability
- Computational Efficiency

Demerits:

- Assumptions
- Linearity
- Sensitivity to Outliers
- Class Imbalance
- Singularity Problems
- Non-normal distribution

ii. Support Vector Machines (SVMs):

Support Vector Machines (SVMs) are a powerful and versatile set of supervised machine learning algorithms used for classification, regression, and outlier detection.

Merits:

- Effective in High-Dimensional Spaces
- Kernel Trick
- Robust to Overfitting
- Effective with Small Datasets
- Versatility

Demerits:

- Computationally Expensive
- Parameter Tuning
- Not Suitable for Large Datasets
- Sensitivity to Noise
- Lack of Probabilistic Outputs

CHAPTER-3

PROPOSED SYSTEM

3.1 PRINCIPAL COMPONENT ANALYSIS (PCA) :

3.1.1 Introduction to PCA

Principal Component Analysis (PCA) is also known as, Hotelling Transform (HT) or Karhunen-Loève Transform (KLT). Principal Component Analysis (PCA) is used for dimensionality reduction image as shown in figure 3.1.1. It is found very effective in the areas of Image and Signal Processing. In area of Image and Signal Processing, PCA is mainly used for reducing size of feature vectors that is used for object recognition and object classification problems.

PCA can be implemented using:

a. EVD (Eigen Values Decomposition)
b. SVD (Singular Values Decomposition)

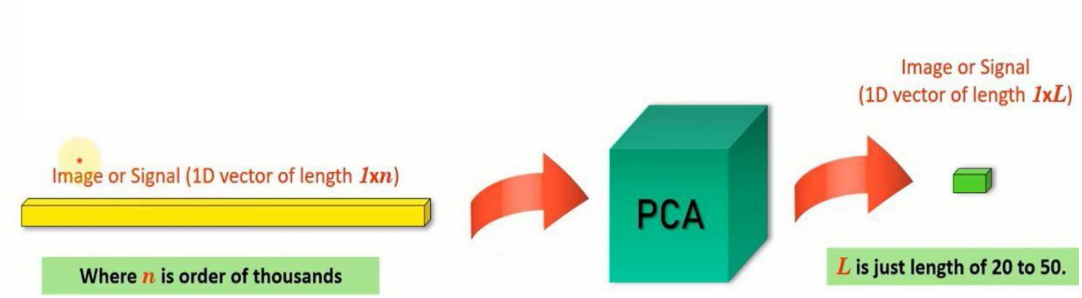


Figure 3.1.1 PCA Reduction

a. EVD (Eigen Values Decomposition)

EVD is a matrix factorization technique that decomposes a square matrix into a set of eigenvectors and eigenvalues.

b. SVD (Singular Values Decomposition)

SVD is a matrix factorization technique that decomposes a matrix into three other matrices: U, Σ, and V^T.

Given a matrix A, SVD decomposes it as: $A = U\Sigma V^T$, where:

- U is a unitary matrix.
- Σ is a rectangular diagonal matrix of singular values.
- V^T is the transpose of a unitary matrix.

3.1.2 Dimensionality Reduction

Consider a data set, which is measured in x-y coordinate system as shown in figure by yellow points. In x-y coordinate system, the data representation needs larger values of x and y. We can observe that data is spread along direction u and hence correlated. It means covariance between x and y values is more. v is another axis that is orthogonal to u. If we choose new axis system u-v for this data instead of x-y, it gives more compact representation of data with small variation along v axis. The origin point of new axis u-v are placed at mean m of the data. Axis u is known as Principal Axis 1 and v is known as Principal Axis 2. In axis system u-v, the data is de-correlated. It means that co-variance between u and v values become very small or zero. For a given set of data, Principal Component Analysis finds the axis system defined by the principal directions of variance (i.e. the u-v axis system as shown in figure-3.1.2). If we can reduce the effect of small variations along v axis, then all data points can be assumed on u axis only, therefore, dimensionality is reduced from 2D to 1D in this case.

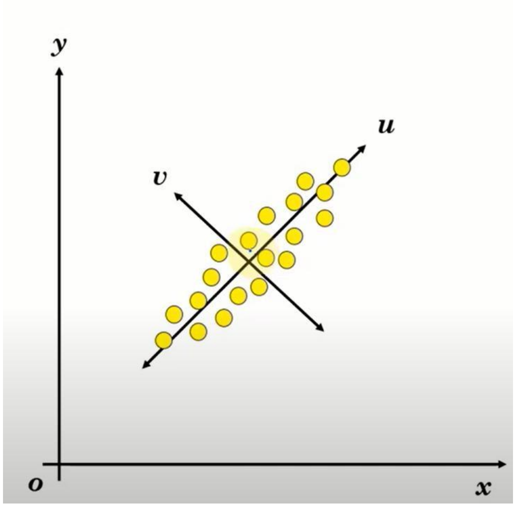


Figure 3.1.2 PCA Component

Covariance Matrix = [var(x) cov(x,y)]

[cov(x,y) var(x)]

In case of x-y axis system, where data is correlated, non diagonal Values (covariance) are strong in covariance matrix and similar to diagonal values (variance) i.e. diagonal is not dominant. In case of u-v axis system, where data is de-correlated Covariance matrix has larger diagonal values (variance) and non diagonal values (covariance) are very small. Therefore, our objective here is to diagonalize the covariance matrix of data set so that it can represent de-correlated data set. And new axis system i.e. orthogonal principal axis can be found which are represented by eigen vectors of covariance matrix.

3.1.3 Implementation of PCA

Let us implement the PCA for a set of images.

- Suppose we have n number of images (x1, x2, x3,........ Xn) of size M x N.
- First, we reshape 2D images to convert them in 1D (Signals are already 1D). Therefore, each image can be represented as 1D vector of size 1 x MN.
- Now we arrange all the images to form a data matrix X such that,

$$X = \begin{matrix} [x1] \\ [x2] \\ [\ldots] \\ [xn] \end{matrix} = \begin{matrix} [x11\ x12\ x13\ldots\ x1MN] \\ [x21\ x22\ x23\ldots\ x2MN] \\ [\ldots\ldots\ldots\ldots] \\ [xn1\ xn2\ Xn3\ldots xnMN] \end{matrix}$$

- The size of this matrix [X] is n x MN.
- Now find mean m (Column wise) of the data matrix X. (m = X': m is known as Mean Image)
- Subtract mean m from each row of X. (Shifting origin to mean of the data for new u - v axis)

Xm = X -m (mean cantered data of size n x MN)

- Find Covariance matrix of Xm,

$$\mathbf{Q = (X^T m . Xm)/(n-1)}$$

- The size of this matrix [Q] is MN x MN, which is very very large. (Main drawback)

This covariant matrix Q is for data set for old axis system where data is correlated. Therefore, we can see that non diagonal values are also strong (cov). To achieve new axis system or to decorrelate the data, we must diagonalize the matrix Q.

In Linear Control System theory, we know that we can diagonalize a matrix with help of a transformation matrix P, such that,

$$\mathbf{(P^{-1}QP=A)}$$

For an orthogonal matrix P, we can write,

$$\mathbf{(P^{T}QP = A)}$$

Where, A is a diagonal matrix of size MN x MN whose diagonal carries all the eigen values (λ) of the matrix Q

$$A = \begin{bmatrix} \lambda 1 & & & & 0 \\ & \lambda 2 & & & \\ & & \lambda 3 & & \\ & & & . & \\ 0 & & & & \lambda MN \end{bmatrix}$$

Matrix P can be found by eigenvectors of Q. As size of Q is MN x MN, therefore, there will be MN eigenvectors of size MN x 1 which are orthogonal to each other.

$$P1 = \begin{matrix} [P11] \\ [P12] \\ [\ldots] \\ [P1MN] \end{matrix}$$

P2 = [P21]

[P22]

[...]

[P2MN]

P3 = [P31]

[P32]

[...]

[P3MN]

...

PMN = [PMN1]

[PMN2]

[...]

[PMNMN]

If we arrange all these eigenvectors side by side, then we can form transformation matrix P as follows:

P = [P1 P2 P3 ... PMN] = [P11 P21 P31 ... PMN1]

[P12 P22 P32 ... PMN2]

[....................................]

[P1MN P2MN P3MN.....PMNMN]

The size of P is MN x MN. This matrix P's also known as Model matrix.

Now this matrix P can be used to decorrelate the data set X to project it to new axis system. This transformation is given as,

Here is the text format of the information presented in the image:

Now this matrix P can be used to decorrelate the data set X to project it to new axis system. This transformation is given as,

$$\mathbf{T_{nxMN} = [Xm]_{n\,x\,MN}.[P]_{MNXMN} = [X\text{-}m]_{nxMN}\,.[P]_{MNXMN}}$$

Here we can see that transformed data set T is of size n x MN. It means no reduction is obtained because we have taken full matrix P for transformation. Where all the eigenvectors are representing all the Principal Axis, therefore, dimension reduction is not achieved.

To achieve dimensionality reduction, we consider only few columns (eigen vectors) of matrix P.

We select only *L* numbers of eigenvectors (columns of *P*) corresponding to *L* numbers of largest eigenvalues (λ).

Therefore, new size of reduced matrix P_{PCA} is *MN x L*. The transformation becomes

$$T_{n\times L} = [X - m]_{n\times MN} \cdot [P_{PCA}]_{MN\times L} \quad ---- (1)$$

Here L is order of 20 - 50, and MN is order of thousands, therefore, we get huge reduction in dimension of data set X {$[X_{nxMN}] > [T_{nxL}]$}.

As we know that each row of matrix X_{nXMN} is one image of size 1 x MN. Now after transformation, our data matrix X_{nXMN} reduces to T .... ,Where each row of matrix T......., represents one image of size 1 x L.

Now we can transform a single image (I) of size M x N to PCA space as,

$$I_{PCA(1\times L)} = [I - m]_{(1\times MN)} \cdot [P_{PCA}]_{(MN\times L)} \quad ---- (2)$$

Also, inverse transform exist i.e. we can get our original image from PCA space to spatial domain by,

$$I_{(1\times MN)} = I_{PCA(1\times L)} [P_{PCA}]^{T}{}_{(L\times MN)} + m_{(1\times MN)} \quad ---- (3)$$

Equation (2) and (3) represent summary of the PCA.

Example:

For example, if we have 100 images of size 128 x 128, then size of data set matrix will be,

100 x (128 * 128) = 100 x 16384

The size of covariant matrix Q becomes 16384 x 16384 and size of transformation matrix P is also 16384 x 16384.

If we select only 50 columns of matrix P corresponding to 50 largest eigen values (λ) then size of reduced transformation matrix P_{PCA} is only 16384 x 50.

From equation (1), the size of transformed data set T is 100 x 50.

$$T_{n\times L} = [X - m]_{n\times MN} \cdot [P_{PCA}]_{MN\times L}$$
$$T_{100\times 50} = [X - m]_{100\times 16384} \cdot [P_{PCA}]_{16384\times 50}$$

Here we can see a huge dimension reduction of data set from 100 x 16384 to 100 x 50.

3.2 BLOCK DIAGRAM

The block diagram has divided into 4 phases depending on the operation of the algorithm. They are:

- Database Preparation
- Training Phase
- Testing Phase
- Result Phase

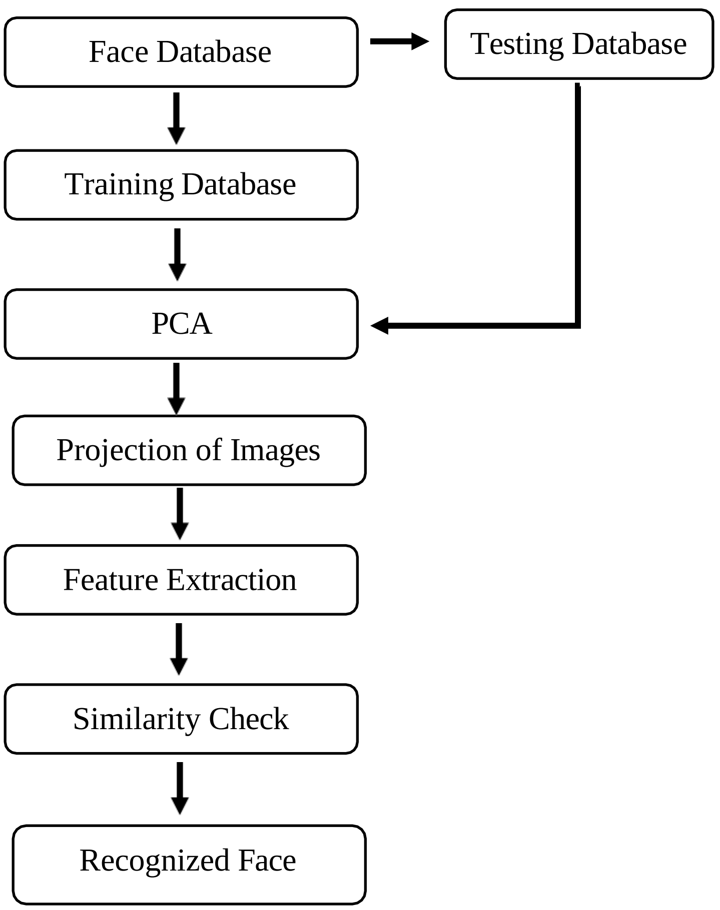


Figure 3.2.1 Block Diagram

3.2.1 Process of PCA

PCA method mathematically follows some steps to perform the recognition task to Figure out the owner of unknown faces by using trained database:

- Obtain a Dataset
- Extract the Mean of Data
- Compute the Covariance Matrix
- Compute the Eigenvectors and Eigenvalues of Covariance Matrix
- Selecting Components and Creating a Feature Vector
- Establishing a New Dataset
- Calling Back the Old Dataset
- Compering the New and Old Datasets with Each Other
- Choosing the Closest Data
- Introducing the Resulted Data as the Recolonized Data

3.2.2 PCA gives the Reduction in Dimensionality

Principal Component Analysis (PCA) gives the reduction in dimensional of the images in 2D to 1D and it is stored as PCA space which is also known as small size PCA Representation. This process taken place on both Training Phase and Testing Phase of our project as shown in figure 3.2.2.

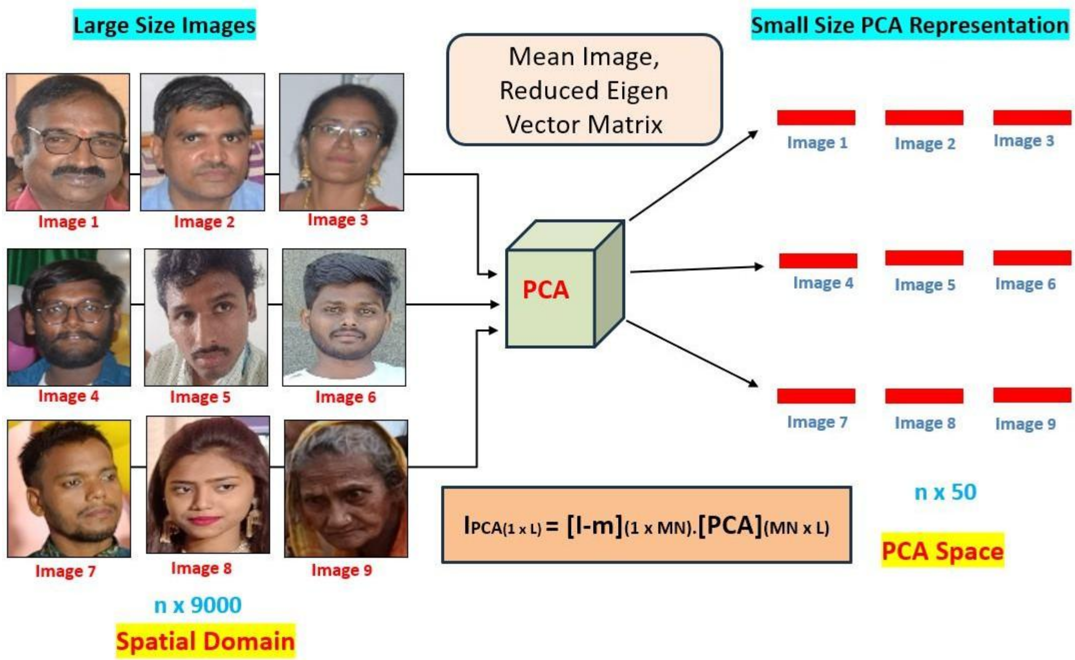


Figure 3.2.2 PCA in Real Time

3.2.3 Database Preparation

For the project we must require for 2 folder which one folder(TrainDatabase) contains the images of straight pose, and another folder(TestDatabase) contains the images like non-uniform motion blur, illumination and different poses as shown in figure 3.2.3

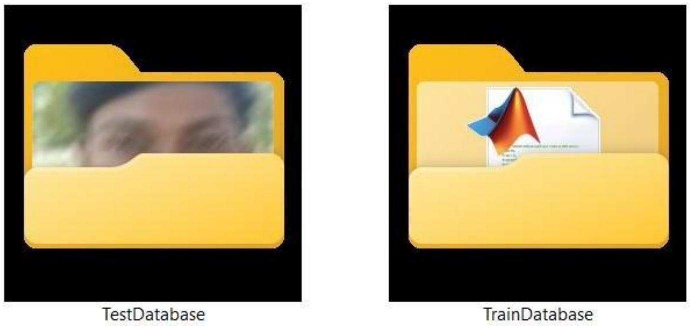


Figure 3.2.3 Database Folder

NOTE:

1. In both, the images should be in the format of JPG and image name should be as 1.jpg, 2.jpg, 3.jpg……….n.jpg and resolution of images should 90 x 100 for both Train Database and Test Database.

2. If we give new images for testing in test database folder it must have related image to that in the Train database also.

3.2.4 Training Phase

In Training Phase, the images data sets of TrainDatabase folder, which are in 2D form images datasets are transformed into 1D dataset and stored in to PCA space. The transformation of image is done by the help of Principle Component Analysis (PCA) as shown in figure 3.2.3. In this phase all the image which are available in the TrainDatabase are converted into dataset at a time and stored into the PCA space which also known as old dataset.

In this phase involves several steps. First, the data is standardized to ensure all variables are on the same scale. Then, a covariance matrix is calculated to understand how the variables relate to each other. Next, eigenvalues and eigenvectors are extracted from the covariance matrix—these represent the magnitude of variance and the directions of maximum variance, respectively. The principal components are chosen based on the eigenvalues, typically focusing on those that explain the most variance. Finally, the data is projected onto these principal components, reducing its dimensions while preserving significant information. This process simplifies the data for further analysis while highlighting its most essential features.

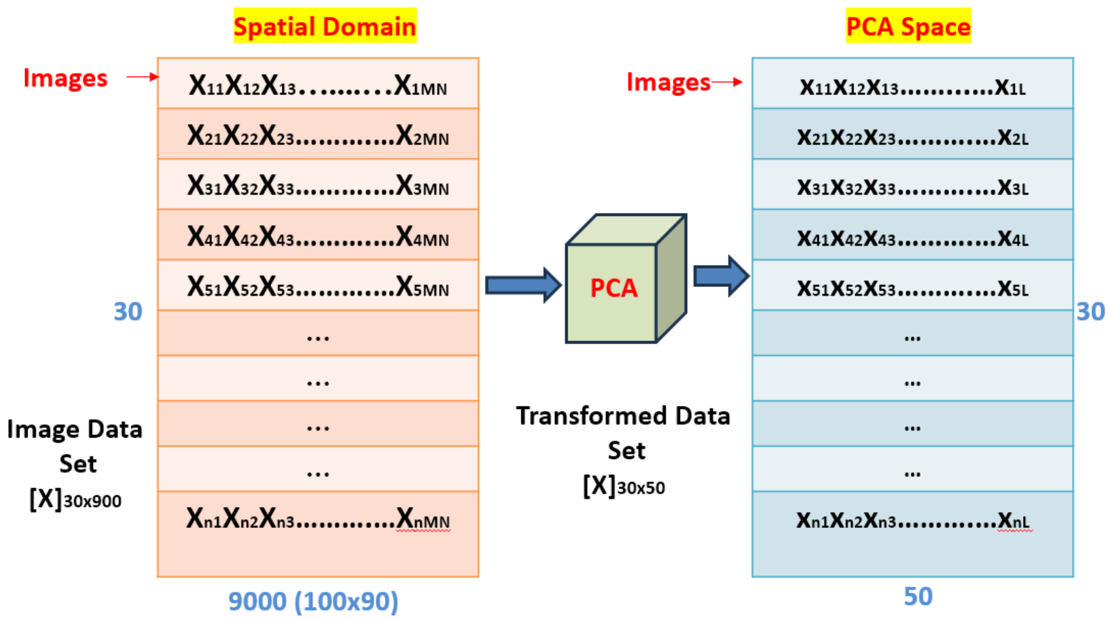


Figure 3.2.4 Training Phase

3.2.5 Testing Phase

In Testing Phase when an input image (Iq) is given for that response then the image is going through the PCA and store image data set in the PCA space. After this process the testing is done at different stages. The stages are:

a. Projection of Images
b. Feature Extraction
c. Similarity Check

a. Projection of Images

From both traindatabase image PCA space image (Ipca) and test database image PCA space image (Iq) are projected in to the feature extraction.

b. Feature Extraction

In this stage both images of train and test PCA stored image features are extracted for the similarity checkup, here different features are extracted such as contrast, saturation, illumination, noise, etc.

c. Similarity Check

In this stage we check the similarity of both train and test PCA space stored image based on the mean variance, least distance, eigenfaces, features vector and image index value.

The above 3 stages are important to recognize the exact image of the given input image as shown in figure 3.2.5.

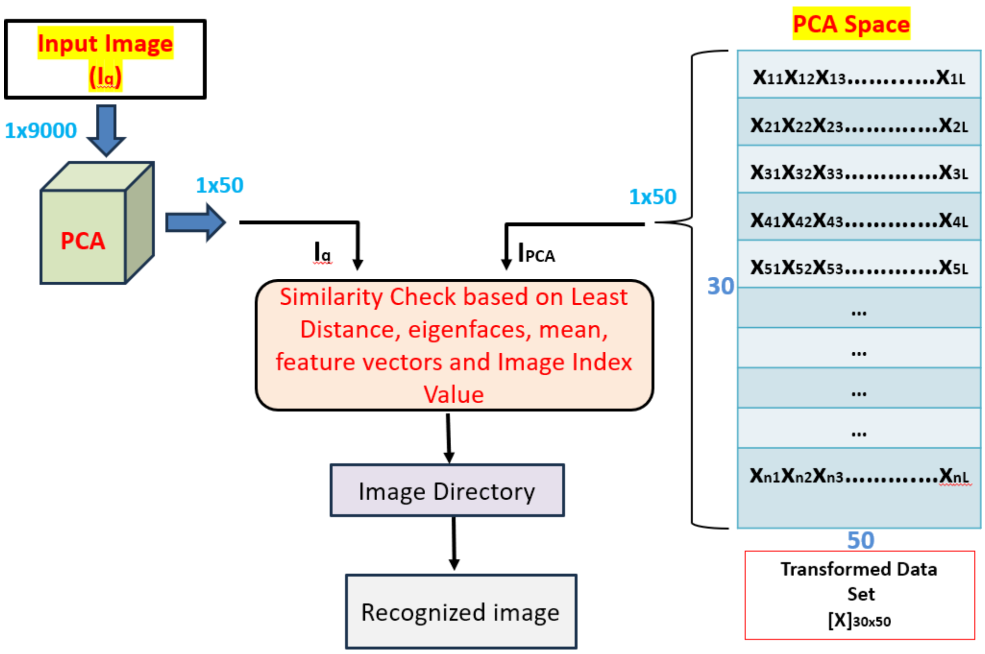


Figure 3.2.5 Similarity Check

3.2.6 Result Phase

This is the final stage where the query image is given as input such as different blur, illumination and pose based on the different features, mean value, eigen value, etc. According to this similarity the image will identify from the Train Database it will project as Recognized image as output.

3.3 SOFTWARE TOOLS

For face recognition across non-uniform motion blur, illumination, and pose using Principal Component Analysis (PCA), MATLAB offers several software tools and functionalities:

3.3.1 Image Processing Toolbox:

MATLAB's Image Processing Toolbox is a powerful suite of tools designed for manipulating, analysing, and enhancing digital images. It allows users to perform a wide range of operations, such as reading and writing images in various formats, adjusting brightness and contrast, applying filters, detecting edges, and segmenting images to identify objects of interest. Additionally, the toolbox supports geometric transformations like resizing and rotation, as well as morphological operations for shape-based analysis. With built-in functions for visualization and detailed analysis, such as measuring object properties or enhancing image quality, the Image Processing Toolbox is highly versatile. It is widely used in fields like medical imaging, computer vision, and industrial automation, providing a robust platform for both simple and complex image-processing workflows.

3.3.2 Computer Vision Toolbox

MATLAB's Computer Vision Toolbox is a specialized toolkit designed to streamline the development and application of computer vision algorithms. It provides functions for tasks such as object detection, feature extraction, image classification, and motion analysis. With built-in support for popular machine learning and deep learning frameworks, users can train and test models for image and video processing. The toolbox also facilitates tasks like camera calibration, stereo vision, and 3D reconstruction. Additionally, it includes tools for analysing and tracking objects in videos, making it valuable for applications like surveillance, autonomous vehicles, and robotics. The Computer Vision Toolbox integrates seamlessly with MATLAB, offering a flexible platform to experiment with innovative vision solutions.

3.3.3 Deep Learning Toolbox

MATLAB's Deep Learning Toolbox provides a comprehensive environment for designing, building, and deploying deep neural networks. It includes pre-built layers, functions, and apps to simplify the creation and training of models for tasks like image recognition, natural language processing, and time-series analysis. This makes the Deep Learning Toolbox a versatile choice for both beginners and experts aiming to solve complex problems through deep learning.

These tools collectively create a robust framework for addressing challenges in face recognition under real-world condition.

CHAPTER-4

EXISTING METHOHD
VS
PROPOSED SYSTEM

4.1 COMPARISION OF EXISTING SYSTEM AND PROPOSED SYSTEM

Here is the comparison of the different existing system and proposed system based on their features:

Features	Existing System				Proposed System
	LDA	ICA	LBP	ASM	PCA
Type	Global features; extracts linear discriminant	Independent features; focuses on statistical independence	Local features; texture-based description	Shape features; statistical modelling of object shapes	Global features; reduces dimensionality by variance analysis
Purpose	Classification dimensional reduction	Signal separation; feature extraction	Texture analysis; pattern recognition	Shape recognition; object alignment	Data compression; feature reduction
Strengths	Maximizes class separability	Handles complex data distributions well	Robust to illumination variations	Effective for shape-based applications	Simplifies complex data
Limitation	Ineffective for non-linear separability	Sensitive to noise; computational overhead	Limited to local and texture-specific tasks	Requires accurate initialization; sensitive to noise	Assumes linearity; can lose critical information
Sensitivity	High	High	High	High	Low
Computation Cost	High	Low	Low	High	Low
Complexity	Low	High	Low	High	Low
Space	Low	High	Low	High	Low

Table 4.1 EXISTING SYSTEM VS PROPOSED SYSTEM

CHAPTER-5

SIMULATION RESULT

5.1 SIMULATION RESULT

Face recognition across non-uniform motion blur, illumination, and pose addresses challenges in real-world scenarios like Non-uniform motion blur, Illumination and Pose as given below.

5.1.1 Non-Uniform Motion Blur

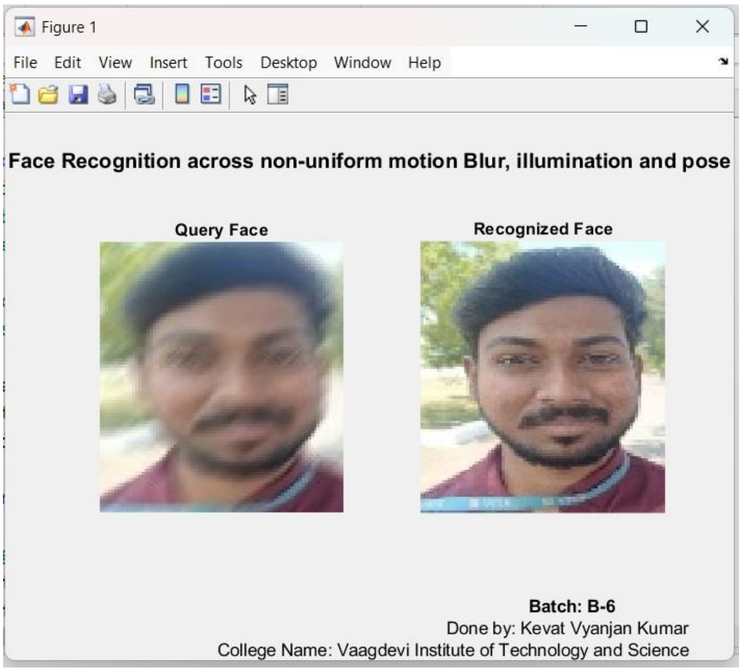


Figure 5.1.1 **Non-Uniform Motion Blur-1**

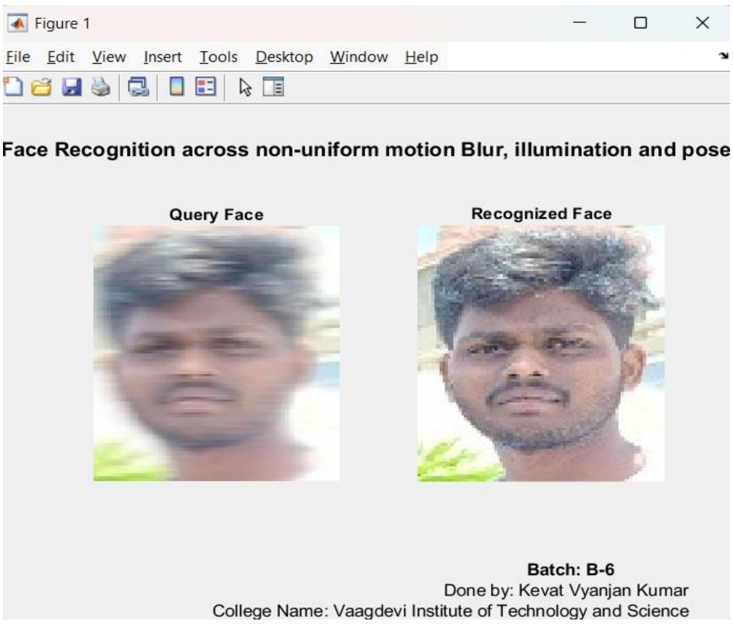


Figure 5.1.2 **Non-Uniform Motion Blur-2**

5.1.2 Illumination

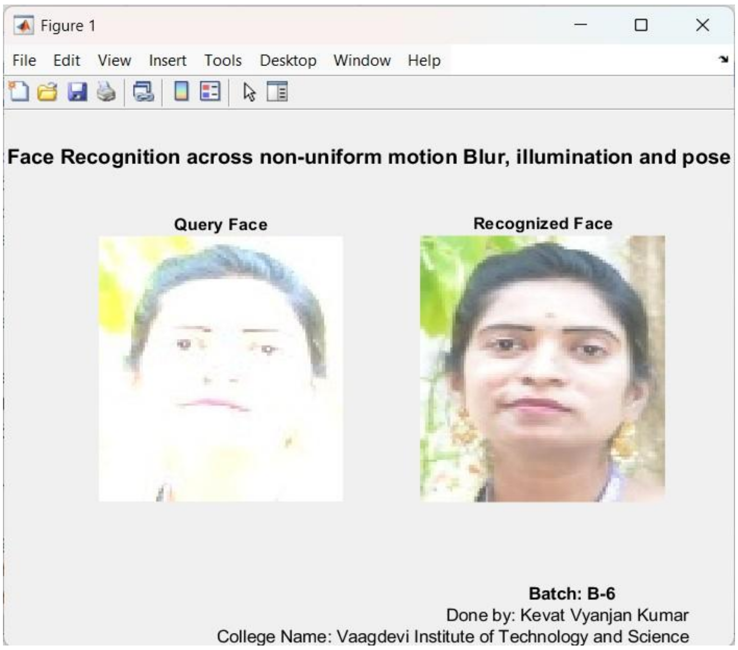


Figure 5.1.3 Illumination-1

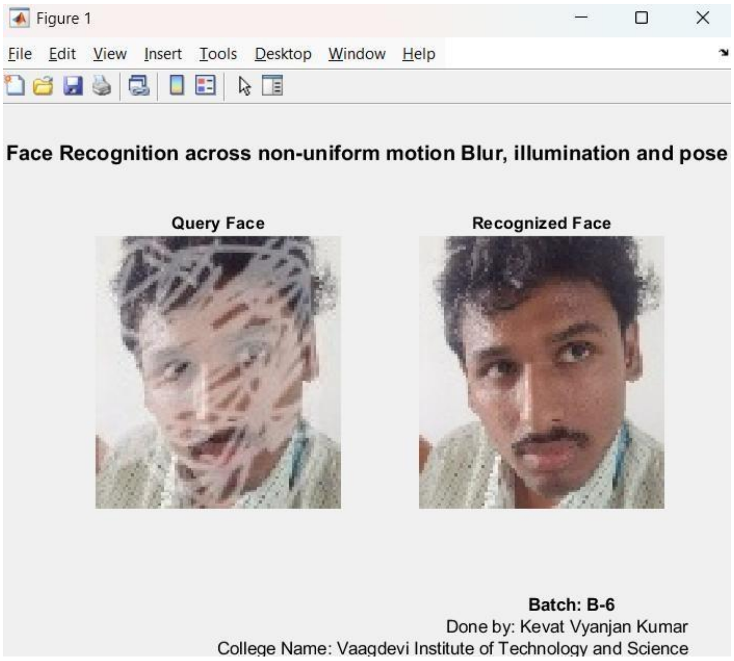


Figure 5.1.4 Illumination-2

5.1.3 Pose

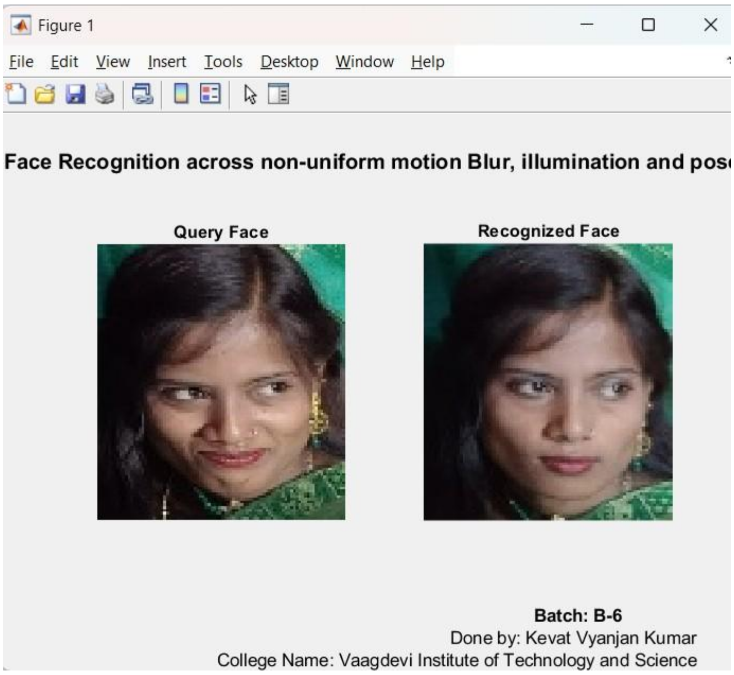


Figure 5.1.5 Pose-1

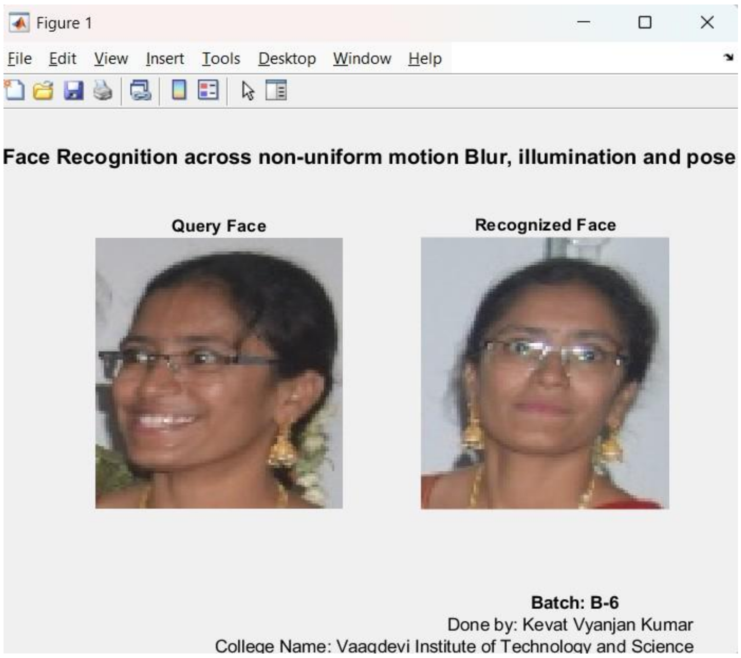


Figure 5.1.6 Pose-2

CHAPTER-6

VALIDATION

6.1 APPLICATIONS

Face recognition across non-uniform motion blur, illumination, and pose has several practical applications, especially in scenarios where traditional recognition systems struggle due to challenging conditions. Here are some key applications with explanations:

a. Surveillance Systems:

In security and monitoring, face recognition systems need to operate effectively even when subjects are moving rapidly, causing motion blur, or under varying lighting conditions. Advanced algorithms can ensure accurate identification in such scenarios

b. Mobile Device Authentication:

Smartphones and tablets often use face recognition for unlocking devices. Robust systems that handle pose variations and illumination changes improve user experience and security.

c. Law Enforcement:

Police and investigative agencies use face recognition to identify suspects or missing people. Systems capable of handling pose and illumination variations are crucial for analyzing footage from diverse sources.

d. Access Control:

In workplaces or restricted areas, face recognition systems can grant or deny access based on facial features. Handling non-uniform motion blur ensures reliability even when individuals are in motion.

e. Healthcare:

In medical settings, face recognition can be used for patient identification, even in challenging conditions like dim lighting or while patients are lying down.

These applications highlight the importance of developing algorithms that can adapt to real-world challenges, ensuring accuracy and reliability in diverse environments.

6.2 ADVANTAGES

Using Principal Component Analysis (PCA) for face recognition across non-uniform motion blur, illumination, and pose variations offers several advantages:

a. Dimensionality Reduction:

PCA reduces the high-dimensional data of facial images to a lower-dimensional representation. This minimizes computational complexity while retaining the most significant features for face recognition.

b. Noise Reduction:

By focusing on the principal components, PCA effectively filters out noise, such as motion blur and minor distortions, improving the accuracy of recognition.

c. Handling Illumination Variations:

PCA emphasizes global facial features, which are less affected by changes in lighting. This makes it robust against varying illumination conditions.

d. Pose Adaptability:

Although PCA is more suitable for frontal face recognition, it can still provide a strong foundation for pose-invariant systems by combining it with other techniques (e.g., pose normalization).

e. Efficient Storage and Processing:

The reduced dimensionality of data saves storage space and enables faster processing, which is crucial for real-time applications like surveillance and mobile device authentication.

While PCA has its limitations, such as assuming linearity, its simplicity and efficiency make it a valuable tool for addressing challenges like motion blur, lighting changes, and pose variations when combined with advanced techniques.

CHAPTER-7

CONCLUSION AND FUTURE SCOPE

7.1 CONCLUSION

The conclusion of using Principal Component Analysis (PCA) for face recognition across non-uniform motion blur, illumination, and pose highlights its effectiveness in dimensionality reduction and feature extraction. PCA helps in identifying key features of faces while discarding irrelevant data, making it robust against variations in lighting, pose, and motion blur. By transforming high-dimensional data into a lower-dimensional space, PCA enhances computational efficiency and recognition accuracy. This approach is particularly valuable in real-world applications where face images are often captured under challenging conditions.

7.2 FUTURE SCOPE

Proposed model gives higher detection accuracy than that of those methods of face recognition across non-uniform motion blur, illumination, and poses using Principal Component Analysis (PCA) is vast and promising. As technology advances, PCA can be further optimized to handle increasingly complex real-world scenarios. One area of growth is the integration of PCA with deep learning techniques, enabling more robust feature extraction and classification under challenging conditions. This hybrid approach could significantly enhance recognition accuracy in dynamic environments.

Another potential lies in real-time applications, such as surveillance systems, where PCA can be adapted for faster processing without compromising precision. Additionally, advancements in hardware, like edge computing devices, can make PCA-based algorithms more efficient and accessible for mobile and embedded systems.

The development of adaptive PCA methods to address diverse datasets and evolving challenges, such as occlusions and expression variations, is another exciting avenue. Furthermore, PCA's role in multimodal biometric systems, combining face recognition with other modalities like voice or iris recognition, could revolutionize security and authentication.

REFERENCES

1. Lin, Kaihan, et al. "Face detection and segmentation based on improved mask R-CNN." Discrete dynamics in nature and society 2020 (2020).

2. Savadatti, Mamta B., et al. "Theoretical Analysis of Viola-Jones Algorithm Based Image and Live-Feed Facial Recognition." 2022 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI). IEEE, 2022.

3. W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, “Face recognition: A literature survey,” ACM Comput. Surv., vol. 35, no. 4, pp. 399–458, Dec. 2003.

4. M. Sorel and F. Sroubek, “Space-variant deblurring using one blurred and one underexposed image,” in Proc. 16th IEEE Int. Conf. Image Process., Nov. 2009, pp. 157–160

5. Y.-W. Tai, P. Tan, and M. S. Brown, “Richardson-Lucy deblurring for scenes under a projective motion path,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no. 8, pp. 1603–1618, Aug. 2011.

6. O.Whyte, J. Sivic, A. Zisserman, and J. Ponce, “Non-uniform deblurring for shaken images,” Int. J. Comput. Vis., vol. 98, no. 2, pp. 168–186, 2012.

7. T. Zhang, Y.-Y. Tang, B. Fang, Z. Shang, and X. Liu, “Face recognition under varying illumination using gradient faces,” Image Processing, IEEE Transactions on, vol. 18, pp. 2599–2606, Nov 2009.

8. Gupta, Ankur, and Purnendu Prabhat. "Towards a resource efficient and privacy-preserving framework for campuswide video analytics-based applications." Complex & Intelligent Systems (2022): 1-16.

9. D. Yi, Z. Lei, and S. Z. Li, “Towards pose robust face recognition,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, June 2013.

10. T. Ahonen, A. Hadid, and M. Pietikainen, “Face description with local binary patterns: Application to face recognition,” Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 28, pp. 2037–2041, Dec 2006.