AN ENHANCED BIOMETRIC IDENTIFICATION AND AUTHENTICATION THROUGH COLLABORATIVE OPTIMIZATION OF CONVOLUTIONAL NEURAL NETWORK AND SUPPORT VECTOR MACHINE

A PROJECT REPORT

Submitted by

AHELESWARAN E	**(Reg.No.: 922319106002)**
PRIYADHARSAN B	**(Reg.No.: 922319106018)**
RAAGHUL R	**(Reg.No.: 922319106019)**
RAGUL PANDIYAN K	**(Reg.No.: 922319106020)**

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

ELECTRONICS AND COMMUNICATION ENGINEERING

UNIVERSITY COLLEGE OF ENGINEERING DINDIGUL

ANNA UNIVERSITY: CHENNAI 600 025

MAY 2023

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report "**AN ENHANCED BIOMETRIC IDENTIFICATION AND AUTHENTICATION THROUGH COLLABORATIVE OPTIMIZTION OF CONVOLUTIONAL NEURAL NETWORK AND SUPPORT VECTOR MACHINE"** is the bonafide work of **"AHELESWARAN E (922319106002), PRIYADHARSAN B (922319106018), RAAGHUL R (922319106019), RAGUL PANDIYAN K (922319106020)"** who carried out the project work carriedout the project work under my supervision.

SIGNATURE	**SIGNATURE**
Dr. S. SUTHA, M.E., Ph.D.,	**DR. R. GOMATHI, M.E., Ph.D.,**
HEAD OF THE DEPARTMENT	**PROJECT SUPERVISOR**
DEAN OF THE COLLEGE	**ASSISTANT PROFESSOR**
Department of Electronics and	Department of Electronics and
Communication Engineering,	Communication Engineering,
University College of Engineering,	University College of Engineering,
Dindigul.	Dindigul.

Submitted for the **VIVAVOICE** held on ____________________

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

We would like to express our sincere thanks to our Dean **Dr. S. SUTHA M.E., Ph.D., University College of Engineering, Dindigul** for forwarding us to do our project and offering adequate duration in completing our project.

We are also grateful to **Dr. S. SUTHA M.E., Ph.D., Head of the Department In ECE, University College of Engineering, Dindigul** for his constructive suggestion and encouragement during our project.

We also express our indebt thanks to our project coordinator **Dr. R. SARAVANA RAM M.E., M.B.A., Ph.D., Assistant Professor, Department of Electronics and Communication Engineering, University College of Engineering, Dindigul** for giving the valuable guidance and encouragement during our project.

We adequately pronounce our sincere gratitude and indebt thanks to our project guide **Mrs. R. GOMATHI, M.E., Ph.D., Assistant professor , Department of Electronics and Communication Engineering ,University College of Engineering, Dindigul** for her personal guidance , inspiration , valuable suggestions and consistence throughout venture for this project.

We also thank all our **Faculty Members** and all **Non–Teaching Staff Members** who have helped us directly and indirectly in this project.

We also humbly extend our earnest gratitude to our loving Family and Friends for their constant support during this Project Work.

ABSTRACT

This project describes the face authentication scheme using SVM and CNN based algorithm is implemented for recognition of face, With the constant development of computer technology, human dependence on network technology have grown, which leads to the importance of security issues. User authentication is an important thing to avoid attacks and security vulnerabilities. Face authentication using convolutional neural networks (CNN) is a popular method for verifying the identity of individuals based on their facial features. CNNs are a type of deep learning neural network that are particularly good at identifying patterns in images. This proposed algorithm provide optimized by SVM(Support vector machine) for tuning the last layers for without any loss of images for better recognition of face

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

CNN	Convolutional Neural Network
SVM	Support Vector Machine
BMP	BitMap
JPEG	Joint Photograph Experts Group
TIFF	Tagged Image File Format
CT	Computed Tomography
PTZ	Pan,Tilt,Zoom
CPU	Central Processing Unit
LED	Light Emitting Diode
SD	Storage Device
ReLU	Rectified linear Unit
Tanh	Hyperbolic Tangent
PCA	Principal Component Analysis
LBP	Local Binary Patterns
HOG	Histogram of Oriented Gradients
RFE	Recursive Future Elimination
VGG	Visual Geometry Group
LFW	Labeled Faces in the Wild
ROC	Region Of Curves

CHAPTER 1
INTRODUCTION

1.1 FACE DETECTION

Biometric identification and authentication is the process of verifying a person's identity based on unique physical or behavioral characteristics, such as fingerprints, facial features, voice patterns, or gait. Biometric systems have become increasingly popular in recent years due to their ability to provide secure and efficient identification and authentication. However, the accuracy of these systems depends on the quality of the input data and the algorithms used for processing it.

In recent years, there has been a growing interest in using machine learning algorithms, such as Convolutional Neural Networks (CNNs) and Support Vector Machines (SVMs), to enhance the accuracy and reliability of biometric systems. CNNs are a type of deep learning algorithm that are particularly effective at processing image data, while SVMs are a type of supervised learning algorithm that can be used for classification tasks.

One approach to enhancing biometric identification and authentication is to use a collaborative optimization of CNNs and SVMs. This approach involves using multiple CNNs to extract features from biometric data, and then using an SVM to classify the extracted features. The CNNs and SVM are trained together in a collaborative optimization framework, where the CNNs learn to extract features that are optimized for the SVM's classification task.

The collaborative optimization of CNNs and SVMs has shown promising results in improving the accuracy and reliability of biometric identification and authentication systems. By leveraging the strengths of both CNNs and SVMs, this approach can overcome some of the limitations of traditional biometric systems, such as sensitivity to noise and variations in input data.

Overall, the collaborative optimization of CNNs and SVMs represents a promising direction for enhancing biometric identification and authentication, and has the potential to significantly improve the security and efficiency of biometric systems in a wide range of applications.

1.2 IMAGE PROCESSING

Image processing is a method to perform some operations on an image, in order to get an enhanced image or to extract some useful information from it. It is a type of signal dispensation in which input is image, like video frame or photograph and output may be image or characteristics associated with that image. Usually Image processing system includes treating images as two dimensional signals while applying already set signal processing methods to them.

Methods of Image Processing

- Analog Image Processing
- Digital Image Processing

1.2.1 ANALOG IMAGE PROCESSING

Analog image processing is any image processing task conducted on two dimensional analog signal by analog means i.e. the alteration of image through electrical means. The most common example is the television image. Analog or visual techniques of image processing can be used for the hard copies like printouts and photographs. When creating images using analog photography, the image is burned into a film chemical reaction triggered by controlled exposure to light. It is processed in a darkroom, using special chemicals to create the actual image. This process is decreasing in popularity due to the advent of digital photography, which requires less effort.

1.2.2 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 and then process it. It is definedas the 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 pixel 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 numbersrepresented by a finite number of bits.

The principle advantages of digital image processing methods are its versatility, repeatability and the preservation of original data precision.

The various image processing techniques are

1) Image Representation
2) Image Enhancement
3) Image Restoration
4) Image Recognition and Interpretation
5) Image Segmentation
6) Image Reconstruction
7) Image Data Compression

1. 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 1.1.

The 2D continuous image f(x,y) is divided 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 was consider to be the physical signal that impingeson the face of a sensor. Typically, an image file such as BMP, JPEG, TIFF etc., hassome header and picture information. A header usually includes details like format identifier (typically first information), resolution, number of bits/pixel, compressiontype etc.,

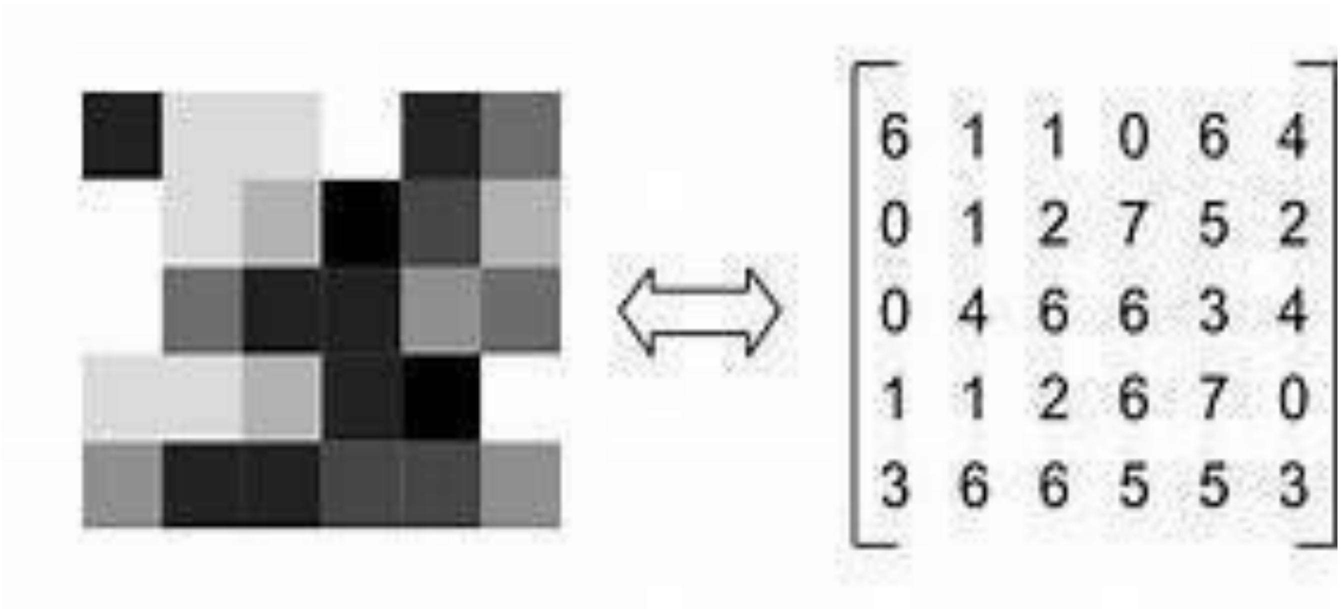

Figure 1.1 Image Representation

2. Image Enhancement

Image enhancement improves the picture displaying quality. Sometimes whenpictures are captured from various resources then the quality of images is not very good due to obstacles. Image enhancement modifies components of the pictures so that clarity of images can be increased. The information content of the image will also be increased by modify the visual impact. This technique is used for analyzing the images, for feature extraction and displaying the images. Algorithms which are used for this process are dependent on applications and interactive. There are some improvement methods namely contrast stretching, noise filtering and histogram modification.

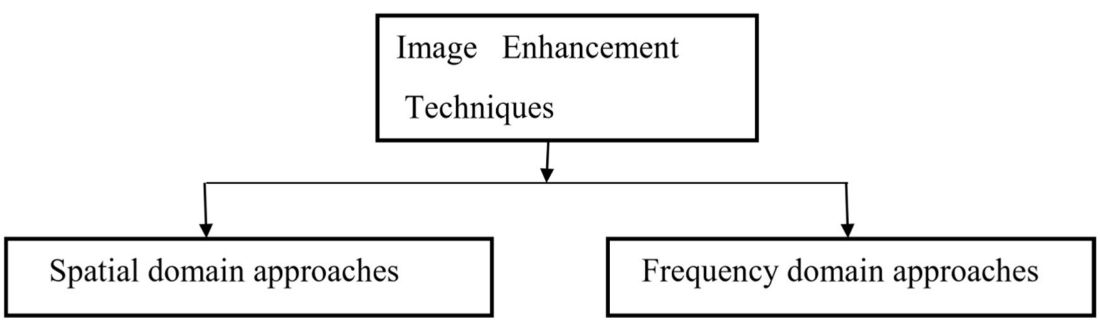

Figure 1.2 Image Enhancement Techniques

Spatial domain techniques are work with pixels. In this technique the values of pixels are altered to achieve the desired enhancement. It contains various techniques whose working directly dependent on the pixels of the images. Frequencydomain methods are appropriate for the images which are based on frequency mechanisms and it works on the orthogonal conversion of the image rather than theimage itself.

3. Image Restoration

Image restoration is a method through which a corrupted and noisy image is processed in such a way that a perfect image is constructed. Thus, restoration rebuilds those images whose quality is despoiled due to noise or system error. Thereare various causes for degradation such as noise from the sensor, camera misfocus and atmospheric disturbance. There are two types of procedures are used to restore the image. One technique is to model the picture whose quality is degraded via somereasons. Another technique known as image enhancement, it increases the quality ofimage by applying various filters. Prior knowledge of degradation is necessary to restore the image. Restoration of the images might be achieved via two types of model namely degradation model and restoration model. Here f(x,y) is the original image which is degraded by some activities. After this on the degraded image variousfunctions are applied in order to restore the image.

4. Image Recognition and Interpretation

Image recognition or analysis is a process of discovering, identifying and understanding patterns that are relevant to the performance of an image based task. One of the principle goals of image analysis is to endow a machine with the capability to approximate similar to human beings. An automated image analysis system is capable of exhibiting various degree of intelligence. Some of the associatedcharacteristics are,

- The ability to extract pertinent information from a background of irrelevant details.
- The capability to learn from examples and to generalize thisknowledge.
- The ability to make inferences from incomplete information.

5. Image Segmentation

Segmentation means partitioning of image into various regions or parts. In image segmentation, an image is divided into subparts according to the requirementof the user or the problem being solved. It divides the image into pixels. Basically this approach is used for analysis of substances, borders and additional records which are relevant for processing. The outcome of image segmentation is a set of sections that together cover the total image or group of contours removed fromthe image. The objective of segmentation is to simplify or to modify the demonstration of picture in such a manner that is more significant and easy to evaluate. It produces the better appearance of image. Segmentation of images is donefor compression of image, recognition of objects and for editing purpose. For image segmentation image thresholding methods are applied. Segmentation allocates labelto each pixel in the image, such that pixel having similar label can share definite features.

Various Method for Image Segmentation

- Region based Segmentation
- Edge based Segmentation
- Feature based clustering Segmentation
- Threshold based Segmentation
- Model based Segmentation

6. Image Reconstruction

Image reconstruction is a special class of image restoration problems where atwo-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 viewingthe object from many different angles. Reconstruction algorithms derive an image ofa 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 nondestructivetesting of assemblies.

7. Image Data Compression

Image compression is the process of encoding or converting an image file in such a way that it consumes less pace than the original file. It is a type of compressiontechnique that reduces the size of an image file without affecting or degrading its quality to a greater. Image compression is typically performed through an image/data compressionalgorithm orcodec. Typically, such codec/algorithms apply different techniques to reduce the image size,such as by:

- Specifying all similarly colored pixels by the color name, code and the number of pixels. This way one pixel can correspond to hundreds or thousands of pixels.
- The image is created and represented using mathematical wavelets.
- Splitting the image into several parts, each identifiable using a fractal.

Some of the common image compression techniques are

- Fractal
- Wavelets
- Chroma sub sampling
- Transform coding
- Run-length encoding

1.3APPLICATIONS OF IMAGE PROCESSING

- Image sharpening and restoration
- Medical field
- Remote sensing
- Transmission and encoding
- Machine/Report vision

- Color processing
- Pattern recognition
- Video processing

1.4 OBJECTIVE

The objective of the proposed system is to improve the accuracy and prediction of human faces by combining the convolution neural network and support vector machine

(a) To provide biometric authentication through CNN .

(b) To enhance the performance of the biometric authentication through SVM.

(c) To compare the performance of the CNN and collabrative performance of the both CNN and SVM.

1.5 FACE RECOGNITION

Face recognition is a technique used for verification or identification of a person's identity by analyzing and relating patterns based on the person's facial features. The evolution of facial recognition and how it is performed is explained in section II. Digital Image Processing is used in this project because there are many algorithms available for image manipulation and problem like noise can be avoided. Facial Recognition has been done in the past with different techniques [3-4] given as:

1) Gabor wavelet-based solutions: Converting a face images using Gabor wavelet so that facial features can be extracted from them. The extracted features can be then used for the face recognition process.

2) Feature distribution of Local Binary Pattern is used to make feature vector that are used in face description.

3) Eigen face-based methods: Recognition using the characteristics of individual. Expression variance and light intensity effect the recognition rate of these techniques. The most effective technique is Machine Learning based which has solved these problems.

1.5.1 FACIAL RECOGINITION

Facial recognition has many applications, one of them is related to security area, human security is one of the issues which is at top on priority because no one want to take risk of life or his expensive assets. In this paper a smart device is proposed to act as an aid to improve security. We have security cameras installed in streets, but they do not tell about the person involved in law breaking activities. PTZ or static cameras installed in street can't recognize a person because they are at some height due to which they are unable to capture the frontal face which is used for efficient recognition. The solution is Facial recognition based smart glasses that can identify a person and show his details and send an alarming signal for further action this was not possible for these cameras because of their placement. They are unable to cover full frontal view of person used for this technique.

Facial Recognition is one of the most emerging and much used application of image processing which has gained much importance and because of vast usage of biometric system during past years [1]. Biometric based techniques are emerged to be the replacement of recognizing individual instead of using Passwords, keys, PINs, cards etc. It is hard to remember them, and they can be stolen, or one can forget them, and they can be easily lost. However, there is no such headache in use of biometric based techniques.

Biometric based identification includes Facial Recognition, Finger or thumb prints, Retina or voice based authentication. Face recognition have advantages over other methods, in other methods user's action is also required like placing thumb on biometric fingerprint verification machine or standing in fixed position for retina identification. Using voice recognition in crowd can be difficult task due to background noise. Facial recognition is the most economic and easy solution to these problems. One more reason is that Facial Recognition has been reliable in many sectors in various ways e.g. commercial, forensic, security etc.

This paper explains about the portable smart glass through which facial recognition will be done. A camera will be mounted on side of our glasses with a Raspberry Pi in the pocket of the user and with the help of data base stored will give authentication of a person. The glasses are designed to perform facial recognition by take input using camera mounted on them and then displaying the output on them. First camera captures an image then the images is sent to CPU, then the CPU matches the images from database in Storage Device (SD) and then results will be displayed on the Organic LED (O-LED). Facial recognition has been done using CNN due to their high frequency and virtuous recognition rate.

1.6 MACHINE LEARNING

Artificial Intelligence usually includes Machine learning that gives the system an ability to learn the things automatically and make certain improvements in them with its experiences. This technique mainly focuses on developing such programs which can access data on their own and then use that data for their learning. Here Deep learning will be used that is subdivision of machine learning, uses multi-layer non-processing units for transformation and extraction of different features and in Deep learning, Convolutional Neural Network is used and the software which we have used is Python for Facial Recognition as it is one of the effective software for image processing algorithms.

The portable smart glasses with a good frontal view recognition is made and the domain which is mainly focused is from security point of view. We can use this in universities during different events, provide it to police so with database of variouscriminals stored they can identify criminals easily. The hardware and software used is described in section III. The block diagram for the glasses is.

1.7 EXISTING SYSTEM

Most of the existing systems attempt to recognize the human prototypic emotions. In the past 20 years, there has been much research into recognizing emotion through facial expressions. However, challenges still remain. Traditionally, the majority approaches for solving human facial expression recognition problems attempt to perform the task on 2-D data.

1.7.1 EXISTING SYSTEM DISADVANTAGES

1) Lighting illumination
2) Face is not detected properly
3) The noise content was not removed properly.
4) The face portion was not segmented properly.
5) In the existing methods, selection of features and classification strategy is difficulties and challenging.

MATLAB SIMULATION

MATLAB (matrix laboratory) is a numerical computing environment and fourth-generation programming language. Developed by Math Works, MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages, including C, C++, Java, and Fortran.

Although MATLAB is intended primarily for numerical computing, an optional toolbox uses the MuPAD symbolic engine, allowing access to symbolic computing capabilities. An additional package, Simulink, adds graphical multidomain simulation and Model-Based Design for dynamic and embedded systems. In 2004, MATLAB had around one million users across industry and academia. MATLAB users come from various backgrounds of engineering, science, and economics.

MATLAB is widely used in academic and research institutions as well as industrial enterprises. MATLAB was first adopted by researchers and practitioners in control engineering, Little's specialty, but quickly spread to many other domains. It is now also used in education, in particular the teaching of linear algebra and numerical analysis, and is popular amongst scientists involved in image processing. The MATLAB application is built around the MATLAB language.

The simplest way to execute MATLAB code is to type it in the Command Window, which is one of the elements of the MATLAB Desktop. When code is entered in the Command Window, MATLAB can be used as an interactive mathematical shell. Sequences of commands 33 can be saved in a text file, typically using the MATLAB Editor, as a script or encapsulated into a function, extending the commands available. MATLAB provides a few features for documenting and sharing your work. You can integrate your MATLAB code with other languages and applications, and distribute your MATLAB algorithms and applications.

FEATURES OF MATLAB

- High-level language for technical computing.
- Development environment for managing code, files, and data.
- Interactive tools for iterative exploration, design, and problem solving.
- 2-D and 3-D graphics functions for visualizing data.
- Tools for building custom graphical user interfaces.
- Functions for integrating MATLAB based algorithms with external applications and languages, such as C, C++.

MATLAB is used in vast area, including signal and image processing, communications, control design, test and measurement, financial modeling and analysis, and computational. Add-on toolboxes (collections of special-purpose MATLAB functions) extend the MATLAB environment to solve classes of problems in these application areas.

MATLAB can be used on personal computers and powerful server systems, including the compute cluster. With the addition of the Parallel Computing Toolbox, the language can be extended with parallel implementations for common computational functions, including for-loop unrolling.

MATLAB can call functions and subroutines written in the C programming language or FORTRAN. A wrapper function is created allowing MATLAB data types to be passed and returned. The dynamically loadable object files created by compiling such functions are termed "MEX-files" (for MATLAB executable).

Libraries written in Java, ActiveX or .NET can be directly called from MATLAB and many MATLAB libraries (for example XML or SQL support) are implemented as wrappers around Java or ActiveX libraries. Calling MATLAB from Java is more complicated, but can be done with MATLAB extension, which is sold separately by Math Works, or using an undocumented mechanism called JMI (Javato-Mat lab Interface), which should not be confused with the unrelated Java that is also called JMI.

As alternatives to the MuPAD based Symbolic Math Toolbox available from Math Works, MATLAB can be connected to Maple or Mathematical.

Development Environment

- Startup Accelerator for faster MATLAB startup on Windows, especially on Windows XP, and for network installations.
- Spreadsheet Import Tool that provides more options for selecting and loading mixed textual and numeric data.
- Readability and navigation improvements to warning and error messages in the MATLAB command window.
- Automatic variable and function renaming in the MATLAB Editor.

- Developing Algorithms and Applications MATLAB provides a high-level language and development tools that let you quickly develop and analyze your algorithms and applications.

The MATLAB Language The MATLAB language supports the vector and matrix operations that are fundamental to engineering and scientific problems. It enables fast development and execution. With the MATLAB language, you can program and develop algorithms faster than with traditional languages because you do not need to perform low-level administrative tasks, such as declaring variables, specifying data types, and allocating memory.

In many cases, MATLAB eliminates the need for 'for' loops. As a result, one line of MATLAB code can often replace several lines of C or C++ code. At the same time, MATLAB provides all the features of a traditional programming language, including arithmetic operators, flow control, data structures, data types, object-oriented programming (OOP), and debugging features. MATLAB lets you execute commands or groups of commands one at a time, without compiling and linking, enabling you to quickly iterate to the optimal solution.

Development tools

MATLAB includes development tools that help you implement your algorithm efficiently. These include the following:

MATLAB Editor - Provides standard editing and debugging features, such as setting breakpoints and single stepping

Code Analyzer - Checks your code for problems and recommends modifications to maximize performance and maintainability

MATLAB Profiler - Records the time spent executing each line of code

Directory Reports - Scan all the files in a directory and report on code efficiency, file differences, file dependencies, and code coverage.

Designing Graphical User Interfaces

By using the interactive tool GUIDE (Graphical User Interface Development

Environment) to layout, design, and edit user interfaces. GUIDE lets you include list boxes, pull-down menus, push buttons, radio buttons, and sliders, as well as MATLAB plots and Microsoft Active controls. Alternatively, you can create GUIs programmatically using MATLAB functions.

MATLAB supports the entire data analysis process, from acquiring data from external devices and databases, through preprocessing, visualization, and numerical analysis, to producing presentation-quality output.

Data Analysis

MATLAB provides interactive tools and command-line functions for data analysis operations, including:

- Interpolating and decimating
- Extracting sections of data, scaling, and averaging
- Thresholding and smoothing
- Correlation, Fourier analysis, and filtering
- 1-D peak, valley, and zero finding
- Basic statistics and curve fitting
- Matrix analysis

Data Access

MATLAB is an efficient platform for accessing data from files, other applications, databases, and external devices. You can read data from popular file formats, such as Microsoft Excel; ASCII text or binary files; image, sound, and video files; and scientific files, such as HDF and HDF5. Low-level binary file I/O functions let you work with data files in any format. Additional functions let you read data from Web pages and XML.

Visualizing Data

All the graphics features that are required to visualize engineering and scientific data are available in MATLAB. These include 2-D and 3-D plotting functions, 3-D volume visualization functions, tools for interactively creating plots, and the ability to export results to all popular graphics formats. You can customize plots by adding multiple axes;

changing line colors and markers; adding annotation, Latex equations, and legends; and drawing shapes.

2-D Plotting

Visualizing vectors of data with 2-D plotting functions that create:

- Line, area, bar, and pie charts.
- Direction and velocity plots.
- Histograms.
- Polygons and surfaces.
- Scatter/bubble plots.
- Animations.

3-D Plotting and Volume Visualization

MATLAB provides functions for visualizing 2-D matrices, 3-D scalar, and 3- D vector data. You can use these functions to visualize and understand large, often complex, multidimensional data. Specifying plot characteristics, such as camera viewing angle, perspective, lighting effect, light source locations, and transparency

3-D plotting functions include:

- Surface, contour, and mesh.
- Image plots.
- Cone, slice, stream.

MATLAB contains mathematical, statistical, and engineering functions to support all common engineering and science operations. These functions, developed 39 by experts in mathematics, are the foundation of the MATLAB language. The core math functions use the LAPACK and BLAS linear algebra subroutine libraries and the FFTW Discrete Fourier Transform library. Because these processor-dependent libraries are optimized to the different platforms that MATLAB supports, they execute faster than the equivalent C or C++ code.

MATLAB provides the following types of functions for performing mathematical operations and analyzing data:

- Matrix manipulation and linear algebra.
- Polynomials and interpolation.
- Fourier analysis and filtering.
- Data analysis and statistics.
- Optimization and numerical integration.
- Ordinary differential equations (ODEs).
- Partial differential equations (PDEs).
- Sparse matrix operations.

MATLAB can perform arithmetic on a wide range of data types, including doubles, singles, and integers.

CHAPTER 2

LITERATURE SURVEY

2.1 INTRODUCTION

A literature survey is a comprehensive and systematic study of existing literature or published research in a specific field. The purpose of a literature survey is to gain a thorough understanding of the current state of knowledge, identify gaps in the research, and provide direction for future research. In a literature survey, researchers review and analyze relevant literature such as academic journals, books, conference proceedings, and online resources related to their research question or topic. They critically evaluate and synthesize the information gathered from the literature to provide a comprehensive overview of the existing knowledge on the topic.

A literature survey plays a crucial role in the research process, as it helps researchers to identify the most important and relevant research questions, theories, and methodologies in their field of study. Overall, a literature survey is an essential step in conducting high-quality research and contributes to the advancement of knowledge in a specific field.

2.2 LITERATURE SURVEY

2.2.1. TITLE: High-quality motion deblurring from a single image.

AUTHOR: Q. Shan, J. Jia, and A.

Agarwala.YEAR: 2008

Traditionally, blurring due to camera shake has been modeled as a convolution with a single blur kernel, and the blur is assumed to be uniform across the image. High-quality motion deblurring from a single image in face recognition using CNN involves collecting adataset of blurred and corresponding sharp face images, training a CNN to estimate the motion blur kernel and recover the sharp image, fine-tuning the pre-trained CNN on the face recognition task, testing the performance of the deblurring algorithm on a separate test dataset, and applying post-processing techniques to further enhance the quality of the deblurred images.

DRAWBACK IDENTIFIED

Only blur was considered, Illumination and pose was not considered.

2.2.2 TITLE: Facial deblur inference using subspace analysis for recognition of blurred faces.

AUTHOR: M. Nishiyama, A. Hadid, H. Takeshima, J. Shotton, and T. Kozakaya.

YEAR : 2011

Face recognition systems that work with focused imageshave difficulty when presented with blurred data. Approaches to face recognition from blurred images can be broadly classified into four categories. One among them is Deblurring-based in which the probe image is first deblurred and then used for recognition. The basic idea is to first model the space of clean facial images as a low-dimensional subspace, and then estimate the subspace coefficients that best fit the observed blurry image.

The subspace coefficients are then used to reconstruct an estimate of the clean image, which can be fed into a CNN for face recognition. The precision of facial recognition can be quantified by using performance metrics such as precision, recall, and F1 score. These metrics can be calculated by comparing the predicted labels of the CNN with the ground truth labels for the test set.

DRAWBACK IDENTIFIED

However, deblurring artifacts are a major source of error especially for moderate to heavy blurs.

2.2.3 TITLE : Close the loop: Joint blind image restoration and recognition with Sparse recognition prior

AUTHOR: H. Zhang, J. Yang, Y. Zhang, N. M. Nasrabadi, and T. S. Huang.

YEAR: 2011

Joint deblurring and recognition method is proposed. The approach of joint blind image restoration and recognition with sparse representation prior in face recognition using CNN involves solving two problems simultaneously: restoring the blurry image to its clean version and recognizing the identity of the restored face using a CNN. To accomplish this,a sparse representation prior is used, which models the clean face as a sparse combination of basis functions. The blurry image is then estimated as a linear combination of the basis functions, with a sparse set of coefficients. By solving an optimization problem with a joint objective function that includes the recognition loss and the sparse representation prior, both image restoration and face recognition can be achieved. The performance of the approach can be evaluated using performance metrics such as precision, recall, and F1 score. These metrics can be calculated by comparing the predicted labels of the CNN with the ground truth labels for the test set.

DRAWBACK IDENTIFIED

Computational Complexity is high.

2.2.4 TITLE: A blur-robust descriptor with applications to face recognition.

AUTHOR: R. Gopalan, S. Taheri, P. Turaga, and R. Chellappa.

YEAR: 2012

Deriving blur-invariant features for recognition. A blur-robust descriptor with applications to face recognition" is a technique used in facial recognition using CNN to improve the accuracy of face recognition in the presence of blur. This technique involves the use of a blur-robust descriptor that is designed to be robust to blur and other forms of image degradation.

The descriptor is based on a novel representation of image patches that captures both the texture and the structure of the patch. The representation is achieved by decomposing the patch into a set of local patches and computing a weighted sum of the local patches, with the weights determined by a spatially varying blur kernel. This results in a descriptor that is invariant to blur and can be used for face recognition even when the image is degraded by blur.

DRAWBACK IDENTIFIED

But these are effective only for mild blurs.

2.2.5 TITLE: Dictionary-based face recognition under variable lighting and pose.

AUTHOR: V. M. Patel, T. Wu, S. Biswas, P. J. Phillips, and R. Chellappa.

YEAR: 2012

Patel et al. have proposed a dictionary-based approach to recognizing faces across illumination and pose. A sparse minimization technique for recognizing faces across illumination. The "dictionary-based face recognition under variable lighting and pose" technique in face recognition using CNN aims to improve the accuracy of face recognition under variable lighting and pose conditions. This approach involves the use of a dictionary-based method that can handle variations in illumination and pose by representing each face as a linear combination of basis functions or "atoms" from a pre-defined dictionary.

The dictionary is trained on a set of face images with varying lighting and pose conditions to learn a set of basis functions that can capture the variations in the images. During recognition, the face image is represented as a sparse linear combination of the basis functions, and the identity of the face is determined based on the coefficients of the basis functions.

DRAWBACK IDENTIFIED

But these works do not deal with blurred images.

2.2.6 TITLE: Robust face recognition via sparse representation.

AUTHOR: J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma.

YEAR: 2009.

Occlusion has been proposed, which is based on similar principles, additionally offers robustness to alignment and pose. The "Robust face recognition via sparse representation" technique in facial recognition using CNN aims to improve the robustness of face recognition under challenging conditions. This approach involves the use of sparse representation, which models each face image as a linear combination of a small number of basis functions from a dictionary. During recognition, the face image is represented as a sparse linear combination of the basis functions, and the identity of the face is determined based on the coefficients of the basis functions. This approach can handle variations in pose, illumination, and expression and can also be used to detect and recognize faces in low-resolution and noisy images.

DRAWBACK IDENTIFIED

But these works do not deal with blurred images.

2.3 SUMMARY

A literature survey is an essential step in any research process. It involves a systematic and comprehensive study of existing literature in a specific field, with the aim of gaining a thorough understanding of the current state of knowledge, identifying gaps in research, and providing direction for future research. Through the critical evaluation and synthesis of information gathered from various sources, researchers are able to provide a comprehensive overview of existing knowledge on a topic, identify important research questions, theories, and methodologies, evaluate strengths and weaknesses of existing research, and identify areas for future research. Literature surveys are crucial for conducting high-quality research and advancing knowledge in a specific field. Therefore, it is important for researchers to conduct a thorough literature survey before embarking on any research project.

CHAPTER 3

CONVOLUTIONAL NEURAL NETWORKS & SUPPORT VECTOR MACHINE

3.1 INTRODUCTION

Convolutional Neural Networks (CNNs) are a type of deep learning neural network that have revolutionized the field of computer vision. They are specifically designed to process and analyze image data and have achieved remarkable success in tasks such as image classification, object detection, and segmentation. CNNs are made up of multiple layers of filters that perform convolutions on the input image to extract features. These filters are designed to detect specific patterns and shapes within an image, such as edges or corners, and combine them into higher-level features. The extracted features are then passed through multiple layers of neural networks for further processing, classification, and recognition. The success of CNNs lies in their ability to automatically learn relevant features from image data without the need for manual feature extraction. This makes them suitable for a wide range of computer vision tasks where the features to be detected may be complex or difficult to define.

CNNs have been used in a variety of applications, from recognizing objects in images and videos to identifying diseases in medical images. They have also been used in natural language processing, speech recognition, and even playing games such as Go and chess. Overall, CNNs have proven to be a powerful and versatile tool for processing and analyzing image data, and their applications are likely to continue expanding as researchers continue to develop and refine their architecture and training techniques.

3.2 CONVOLUTIONAL NEURAL NETWORK ARCHITECTURE

CNN typically has three layers: a convolutional layer, a pooling layer, and a fully connected layer. Architecture is a type of deep learning neural network commonly used for image and video processing. It consists of multiple layers, including convolutional, pooling, and fully connected layers, and is highly effective for efficient and accurate feature extraction and classification. Transfer learning can also be used for customization.

1) **Input Layer:** The first layer of a CNN receives the input image. It preserves the spatial structure of the image and passes it to the subsequent layers. The input layer may also include additional preprocessing steps, such as normalization or data augmentation.

2) **Convolutional Layers:** Convolutional layers are the core building blocks of a CNN. Each layer consists of multiple filters (also known as kernels) that slide or convolve over the input image, computing the dot product between the filter weights and the local receptive field of the image.

3) **Activation Function:** Typically, an activation function follows each convolutional layer to introduce non-linearities into the network. Common activation functions used in CNNs include Rectified Linear Unit (ReLU), sigmoid, or hyperbolic tangent (tanh).

4) **Pooling Layers:** Pooling layers are used to downsample the spatial dimensions of the feature maps while retaining the most important information. Max pooling is a commonly used pooling technique, which selects the maximum value within a pool (e.g., 2x2) and discards the rest. Pooling reduces the computational complexity and makes the network more robust to spatial translations.

5) **Fully Connected Layers:** The output from the convolutional and pooling layers is typically fed into one or more fully connected layers. These layers resemble traditional artificial neural networks, Fully connected layers help in capturing high-level representations and making predictions based on the learned features.

6) **Dropout:** Dropout is a regularization technique often used in CNNs to prevent overfitting. During training, random neurons are temporarily dropped out or deactivated with a certain probability. This forces the network to learn more robust features by reducing the reliance on specific neurons.

7) **Output Layer:** The final layer of the CNN produces the desired output. For image classification tasks, it is usually a softmax layer that assigns probabilities to different classes. In other tasks like object detection, the output layer may have a different configuration based on the specific requirements.

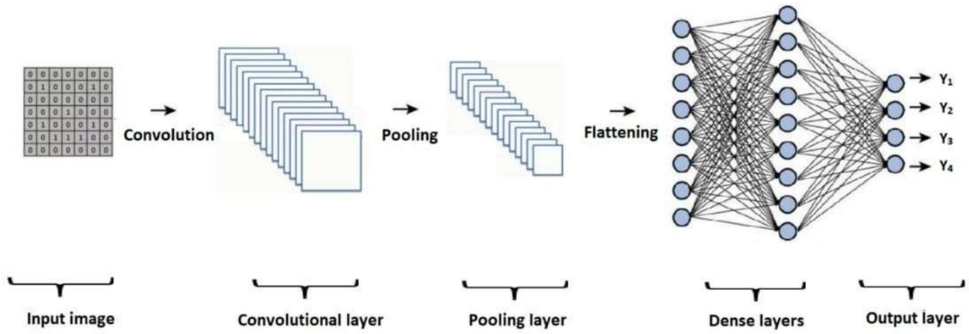

Figure 3.1 Schematic Diagram of CNN

3.2.1 CONVOLUTIONAL LAYER

A Convolutional Layer is a key component of a Convolutional Neural Network (CNN) used in image processing tasks. It applies a set of learnable filters to the input image to extract important features and create a feature map. The filters are trained during the training process, allowing the CNN to automatically learn relevant features from image data. One of the key advantages of convolutional layers is their ability to learn features that are invariant to translation, allowing CNNs to identify objects in an image regardless of their position, rotation, or scale.

Convolution layer is the first layer to extract features from an input image. By learning image features using a small square of input data, the convolutional layer preserves the relationship between pixels. It is a mathematical operation which takes two inputs such as image matrix and a kernel or filter.

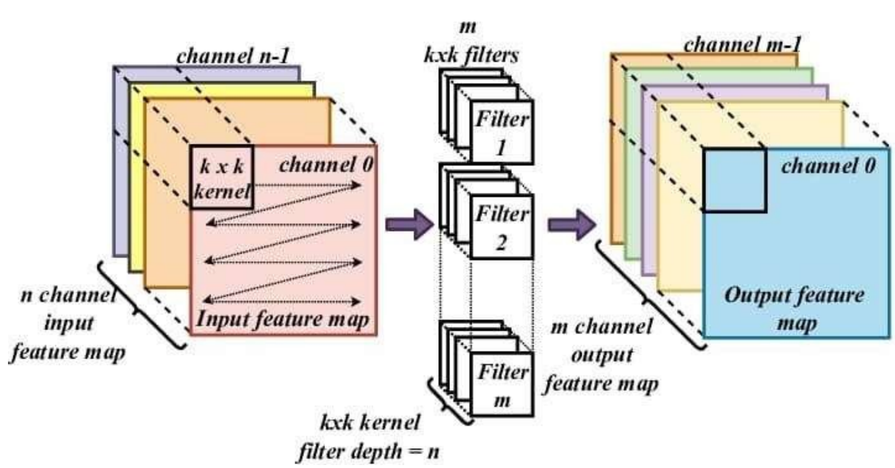

Figure 3.2 Convolution Layer Operation

1) **Convolution Operation:** The convolutional layer applies a set of learnable filters (also called kernels) to the input image or feature map. Each filter is a small matrix that slides across the input spatially, computing dot products between its values and the corresponding input values. This process captures local patterns or features.

2) **Feature Maps:** Each filter generates a feature map as it convolves across the input. A feature map represents the response of a specific filter to different local patterns present in the input. Multiple filters are used in a convolutional layer to capture different types of features.

3) **Stride:** The stride determines the step size at which the filter moves across the input. A stride greater than 1 reduces the spatial dimensionality of the output feature map, resulting in spatial downsampling.

4) **Padding:** Padding is often applied to the input image to preserve spatial dimensions during convolution. It adds additional border pixels around the input, allowing the filters to operate on pixels near the edges and maintain the output size.

5) **Activation Function**: After the convolution operation, an activation function (such as ReLU) is typically applied element-wise to introduce nonlinearity. This helps the network learn complex and abstract features.

6) **Pooling:** Convolutional layers are often followed by pooling layers. Pooling reduces the spatial dimensions of the feature maps, typically by selecting the maximum (MaxPooling) or average (AveragePooling) value within a local region. Pooling helps to downsample the feature maps while preserving important features.

7) **Depth:** Convolutional layers have a depth dimension, which represents the number of filters or channels in the layer. Each filter generates a separate feature map, and the depth dimension represents the number of feature maps produced by the layer.

8) **Parameters:** The parameters of a convolutional layer include the filter weights (learned during training) and biases (optional). These parameters are updated through backpropagation to optimize the network during training.

Example:

The dimension of the image matrix is h×w×d.

The dimension of the filter is fh×fw×d.

The dimension of the output is (h-fh+1)×(w-fw+1)×1.

Convolutional Neural Network

Let's start with consideration a 5*5 image whose pixel values are 0, 1, and filter matrix 3*3 as:

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

5 × 5 – Image Matrix 3 × 3 – Filter Matrix

Figure 3.3 Filtering of image

The convolution of 5*5 image matrix multiplies with 3*3 filter matrix is called "Features Map" and show as an output.

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 4 & 3 & 4 \\ 2 & 4 & 3 \\ 2 & 3 & 4 \end{bmatrix}$$

Convolved Feature

Figure 3.4 Convolution of image

Convolution of an image with different filters can perform an operation such as blur, sharpen, and edge detection by applying filters.

3.2.2 POOLING LAYER

A Pooling Layer in a Convolutional Neural Network (CNN) reduces the spatial dimensions of the feature maps generated by the convolutional layers by dividing them into non-overlapping regions and computing a summary statistic for each region, such as the maximum or average value. This down samples the feature maps, reduces the computational complexity of the network, and introduces a degree of invariance to small variations in the position of features within the image, making the network more robust to input variations.

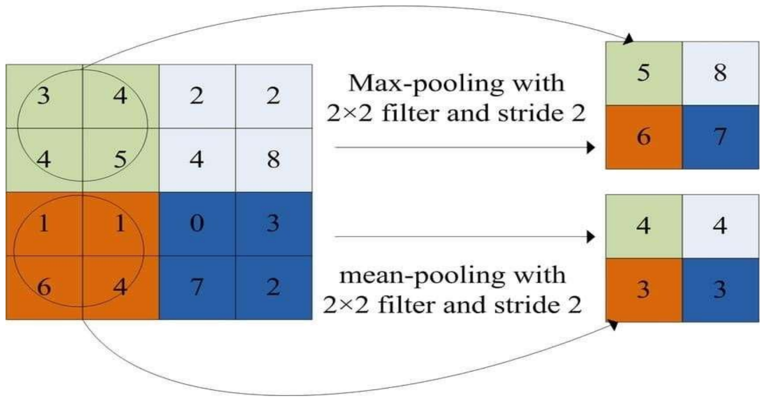

Figure 3.5 Max Pooling Layer Operation

3.2.3 FULLY CONNECTED LAYER

A Fully Connected Layer in a Convolutional Neural Network (CNN) is a layer where each neuron is connected to every neuron in the previous layer, used to map high-level features to a set of output classes. The output of the Fully Connected Layer is a set of class scores, which represent the probability of the input image belonging to each class. During training, the weights of the Fully Connected Layer are learned using backpropagation and gradient descent to minimize the error between the predicted output and the actual output.

3.2.4 NON-LINEARITY LAYER

Non-linearity Layers, also known as Activation Function Layers, are a crucial component of neural networks, including Convolutional Neural Networks (CNNs). These layers apply a non-linear function to the output of the previous layer, allowing the network to learn more complex representations and make accurate predictions.

Common activation functions used in Non-linearity Layers include Rectified Linear Units (ReLU), Sigmoid, and Tanh. ReLU is the most popular activation function used in CNNs due to its computational efficiency and ability to handle sparse input data, while Sigmoid and Tanh are used less frequently due to the vanishing gradient problem.

Non-linearity Layers are typically inserted after convolutional and fully connected layers in CNNs, enabling the network to learn more complex features and perform better on tasks such as image classification or natural language processing. Overall, Non-linearity Layers play a crucial role in the success of neural networks by introducing non-linearity and enabling the network to model complex relationships between inputs and outputs.

3.2.5 APPLICATION OF CNN

Convolutional Neural Networks (CNNs) have numerous applications in various fields. Some of the most common applications of CNNs include:

1) **Image Recognition:** CNNs have been used extensively for image recognition tasks such as object detection, face recognition, and scene classification. CNNs can identify patterns in images and accurately classify them into different categories.
2) **Natural Language Processing:** CNNs have also been used for natural language processing tasks such as sentiment analysis and text classification. CNNs can learn to recognize patterns in text data and classify text into different categories based on those patterns.
3) **Autonomous Vehicles:** CNNs are used in autonomous vehicles to identify and track objects such as pedestrians, vehicles, and traffic signs.
4) **Medical Image Analysis:** CNNs have been used for medical image analysis tasks such as identifying tumors, detecting abnormalities in medical scans, and predicting patient outcomes.
5) **Video Analysis:** CNNs are used in video analysis applications such as activity recognition, object tracking, and video classification.

3.3 SUPPORT VECTOR MACHINE

3.3.1 INTRODUCTION

Support Vector Machines (SVMs) are a class of machine learning algorithms used for classification and regression analysis. SVMs are popular due to their ability to handle high-dimensional data and provide accurate predictions even with limited training data.

The main idea behind SVMs is to find the best hyperplane that separates data into different classes. A hyperplane is a line or plane that divides a dataset into two classes such that the distance between the closest data points of each class is maximized.

The SVM algorithm finds the hyperplane that maximizes the margin, which is the distance between the hyperplane and the closest data points of each class. SVMs are capable of handling both linear and non-linearly separable data by transforming the input data into a higher dimensional space using a technique called kernel trick.

This transformation allows the data to be separated by a hyperplane in the higher dimensional space, even if it is not separable in the original feature space. SVMs have a wide range of applications, including image classification, text classification, and bioinformatics. SVMs have been shown to provide high accuracy and are widely used in machine learning research and real-world applications.

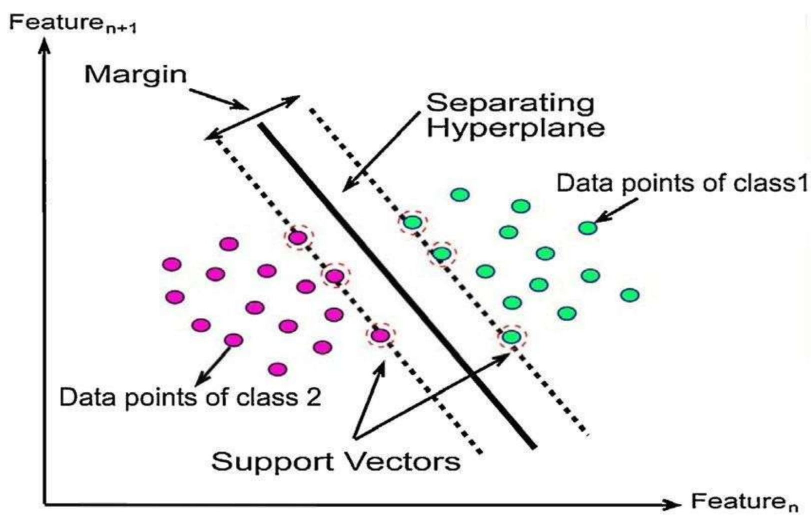

Figure 3.6 Structure Of SVM

3.3.2 SVM optimization

Support vector machine (SVM) optimization is a common technique used in face recognition systems to improve the accuracy of the classification model. Here are some of the steps you can take to optimize SVM for face recognition: Feature extraction: The first step in optimizing an SVM-based face recognition system is to extract the most discriminative features from the face images. There are many feature extraction techniques available, including Principal Component Analysis (PCA), Local Binary Patterns (LBP), and Histogram of Oriented Gradients (HOG).

Feature selection: After feature extraction, it is often useful to reduce the dimensionality of the feature space by selecting the most informative features. This can be done using techniques such as Recursive Feature Elimination (RFE) or Select Best. Parameter tuning: The performance of an SVM classifier is highly dependent on the choice of hyperparameters such as the kernel type, regularization parameter (C), and gamma. These hyperparameters can be optimized using techniques such as grid search or randomized search.

Training data augmentation: One way to improve the generalization of an SVM-based face recognition system is to use data augmentation techniques such as image rotation, scaling, and flipping. This can help the classifier to learn more robust features and reduce overfitting.

Ensemble methods: Another approach to improve the performance of SVM-based face recognition systems is to use ensemble methods, such as bagging or boosting, which combine multiple SVM classifiers trained on different subsets of the data.

Cross-validation: Finally, it is important to evaluate the performance of the SVM-based face recognition system using cross-validation techniques such as k-fold cross-validation or leave-one-out cross-validation. This can help to estimate the generalization performance of the system and avoid overfitting to the training data.

Overall, SVM optimization is an important step in building accurate and robust face recognition systems. By selecting the most informative features, optimizing the SVM hyperparameters, and using data augmentation and ensemble methods, you can significantly improve the accuracy of the classifier.

3.3.3 APPLICATIONS OF SUPPORT VECTOR MACHINE (SVM)

Support Vector Machines (SVMs) are widely used in various applications, including:

- **Image Recognition:** SVMs have been used extensively for image recognition tasks, such as object detection, face recognition, and image classification. SVMs can be used to classify images based on their features, such as texture, color, and shape.
- **Natural Language Processing:** SVMs are also used for natural language

processing tasks, such as sentiment analysis, text classification, and named entity recognition. SVMs can be used to classify text data based on features such as word frequency, part-of-speech, and semantic meaning.

- **Bioinformatics:** SVMs have been used for various bioinformatics applications, such as protein classification, gene expression analysis, and drug discovery. SVMs can be used to classify biological data based on features such as sequence, structure, and function.
- **Fraud Detection:** SVMs can be used for fraud detection in various industries, such as banking and insurance. SVMs can be trained on a dataset of known fraudulent transactions and used to identify new fraudulent transactions based on their features.
- **Medical Diagnosis:** SVMs can be used for medical diagnosis tasks, such as disease diagnosis and prediction. SVMs can be trained on a dataset of patient data and used to predict the likelihood of a patient developing a particular disease based on their features.
- **Quality Control:** SVMs can be used for quality control in manufacturing industries. SVMs can be trained on a dataset of acceptable and defective products and used to identify defects in new products based on their features.

Overall, SVMs are a powerful machine learning algorithm that can be applied to a wide range of applications. SVMs are particularly effective in tasks where the data is high-dimensional and complex, and the goal is to classify or predict outcomes.

Support Vector Machines (SVMs) are often used in face recognition tasks because they have several advantages that make them suitable for such applications. Here are some reasons why SVMs are considered one of the best approaches for face recognition.

Effective in high-dimensional spaces: Face recognition typically involves highdimensional feature spaces where each face image can have thousands or even millions of features. SVMs are capable of handling such high-dimensional data effectively and efficiently.

Non-linear classification: SVMs can capture non-linear relationships between features and classes through the use of kernel functions. This allows SVMs to handle complex face patterns and variations in a robust manner.

Robustness to outliers: SVMs are designed to be robust to outliers and noise in the training data. In face recognition, images can have variations due to changes in lighting conditions, pose, and expression. SVMs can handle these variations well, making them more reliable in real-world scenarios.

Optimal separation: SVMs aim to find the hyperplane that maximally separates different classes. This helps in achieving better discrimination between different individuals' faces, leading to more accurate recognition results. Small memory footprint: Once trained, SVM models only need to store a subset of the training data points called support vectors. This makes SVMs memory-efficient, which is important in face recognition systems where large databases of face images may need to be processed.

Well-established theory: SVMs are based on solid mathematical foundations and have been extensively studied and refined over the years. They have a strong theoretical background, and their performance characteristics are well-understood.

However, it's worth noting that while SVMs have been widely used for face recognition, other machine learning algorithms, such as deep neural networks, have also shown excellent performance in recent years. Deep learning approaches, including convolutional neural networks (CNNs), have achieved state-of-the-art results on various face recognition benchmarks. The choice of algorithm depends on various factors, including the size of the dataset, computational resources, and specific requirements of the application.

Support Vector Machines (SVMs) and Convolutional Neural Networks (CNNs) are two powerful machine learning algorithms that are widely used in various applications, including image recognition and classification. Both algorithms have their strengths and weaknesses, and combining them can result in a more robust and accurate system. The benefits of SVMs include their ability to handle high-dimensional data, work well with small to medium-sized datasets, and have a strong theoretical foundation. SVMs can also handle non-linear data by using kernel functions, making them a popular choice for classification tasks. On the other hand, CNNs are well-suited for image recognition and classification tasks because they can automatically learn relevant features from raw input data. CNNs can learn to recognize patterns and shapes in images, making them highly effective at image classification tasks.

Combining SVMs and CNNs can take advantage of their respective strengths, resulting in a more powerful and accurate system. For example, a typical approach is to use CNNs to extract features from the raw input data, and then use an SVM to classify the extracted features. This approach can result in a more efficient and effective system because the CNNs can learn relevant features from the input data, and the SVM can make use of those features to make accurate classifications. Another benefit of combining SVMs and CNNs is that it can improve the interpretability of the system.

SVMs can provide insight into which features are most important for classification, while CNNs can provide information about the specific features that are recognized in the input data. Overall, combining SVMs and CNNs can result in a powerful and accurate system for image recognition and classification tasks. By taking advantage of their respective strengths, these two algorithms can work together to produce a more robust and effective solution.

CHAPTER 4
PROPOSED METHODOLOGY

4.1 PROPOSED SYSTEM

The proposed system CNN model is implemented for recognition of face recognition of face,At present, the typical architecture of neural network is divided into the following categories: LeNet5, AlexNet, ZF Net, GooLeNet, and VGGNet, the following will LeNet5 architecture for a detailed analysis. LeNet5 is a CNN classic structure that existed long ago, and it is mainly used in the recognition of face recognition fonts. It contains a total of seven layers of structure, except for the input layer, each of the other has training parameters, and each layer contains a plurality of Feature Maps, we can extract the input features through a convolution kernel. And each feature contains multiple neurons.

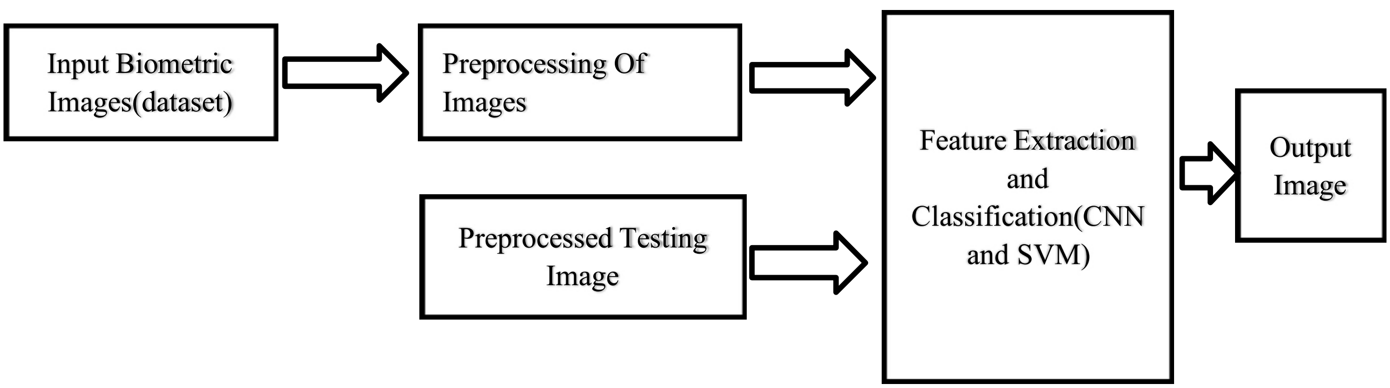

Figure 4.1 Framework of Proposed Methodology

The proposed system for facial recognition using SVM involves collecting and preprocessing a dataset of facial images, extracting relevant features, training an SVM model to classify the feature vectors, and testing and evaluating the accuracy of the system. This approach can be effective for facial recognition when a large amount of training data is available and the feature extraction algorithm is well-suited to the dataset.

4.2 FACE IMAGE COLLECTION AND PROCESSING

Image processing based on convolutional neural networks needs to collect a large number of pictures for the computer to learn. This topic will take a lot of people a lot of images, after collecting a lot of images cropped irrelevant parts of the face. This article uses the face detection and cut saved in the created folder. At this time, the collected images have been trimmed and resized. Then all the images are stitched and stitched in the Olivetti faces face dataset, each line represents the category of two people, after all the face images stitching together, and then get the small face database. The figure below is the subject of the face data set to be trained.

Convolution neural network model construction CNN designed in this paper contains the following layers of structure, which are the input layer, conv, pooling, all connected layer, output layer, convolutional layer and the down sampled layer there will be many. In this paper, the reference to LeNet5 model to achieve this article CNN model set up. The design of the model will be a convolution layer and a down sampled layer merged into a layer, named "LeNetConvPoolLayer", a total of two layers "LeNetConvPoolLayer" in the third layer convolution plus sampling layer connected a full connection Layer, named "Hidden Layer", this fully connected layer is similar to the hidden layer in a multi-layer perceptron. The last layer is the output layer, because it is a multi-faceted face classification,

Image preprocessing involves enhancing the quality of the captured images, including adjusting the lighting, removing noise, and aligning the images to a standard pose. This step is essential to improve the accuracy of feature extraction and recognition. Feature extraction is the process of identifying and extracting key facial features such as the eyes, nose, and mouth from the preprocessed images.

The extracted features are then used to create a face template or representation, which can be used for identification or verification. These templates can be compared against a database of pre-existing templates to identify or verify an individual's identity.

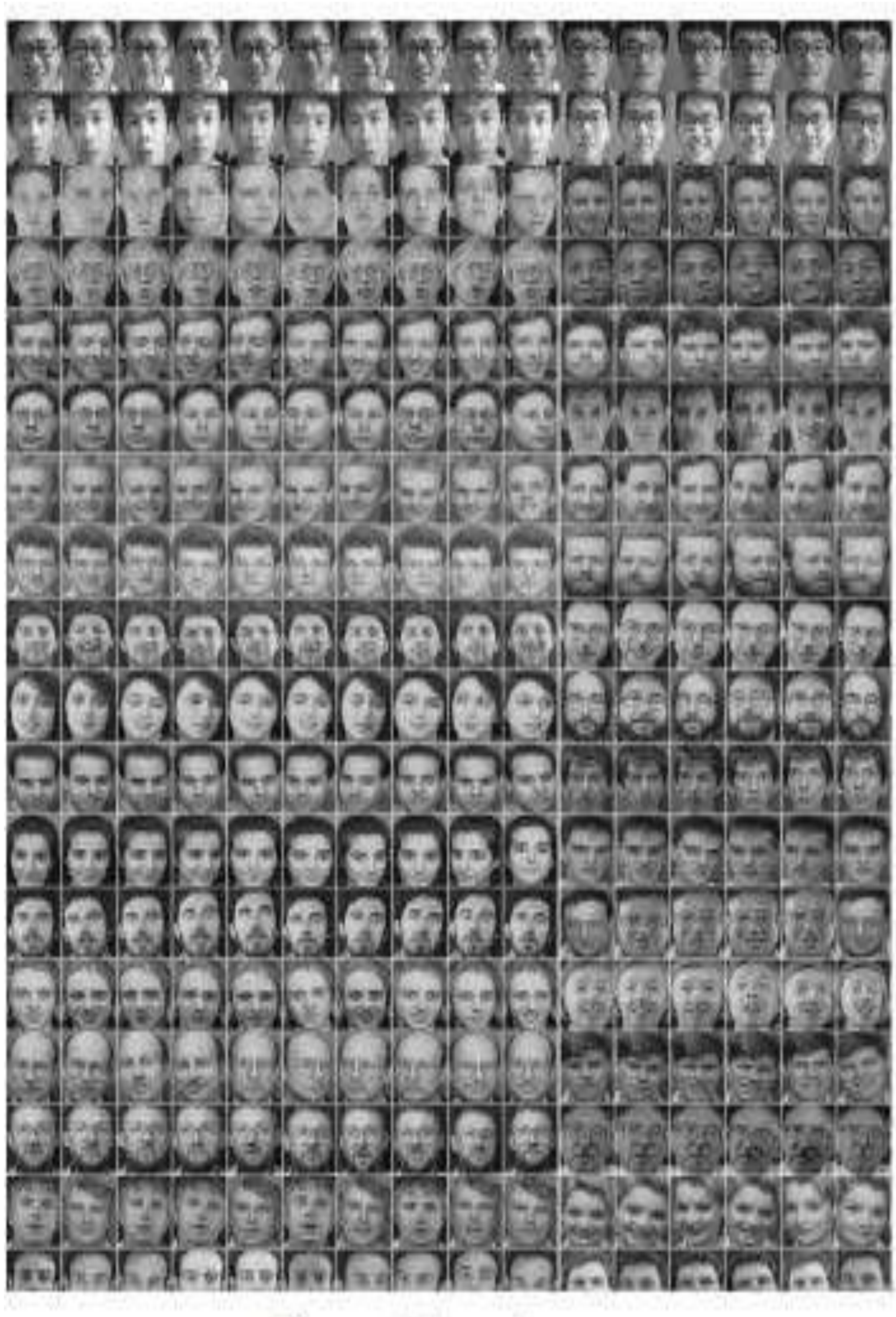

Figure 4.2 Trained Dataset For Face Recognition

4.2.1 Training of CNN

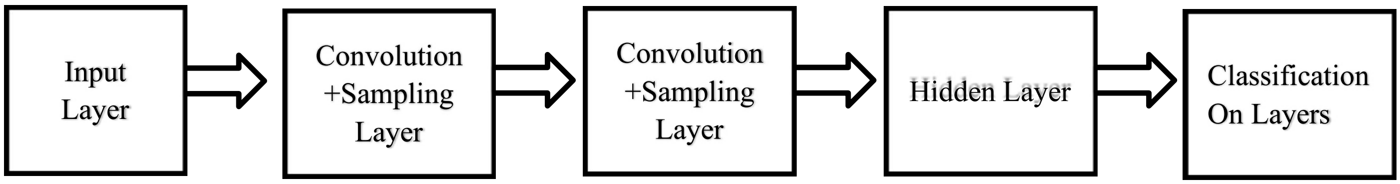

Figure4.3 CNN design for face authentication structure form

1) **Feature Extraction:** The next step is to extract features from the facial images. This can be done using techniques like Principal Component Analysis (PCA) or Local Binary Patterns (LBP) to extract relevant facial features. The output of this step is a matrix of feature vectors, with each row representing a different facial image.
2) **SVM Training:** Once the feature vectors are extracted, an SVM model is trained using the feature vectors as input. The SVM is trained to classify the feature vectors as belonging to a particular individual or not. The SVM learns a decision boundary that separates the feature vectors of different individuals.
3) **Testing and Evaluation:** Once the SVM model is trained, it can be used to test the accuracy of the facial recognition system. This involves inputting a new facial image into the feature extraction algorithm to obtain a feature vector, which is then classified by the SVM as belonging to a particular individual or not. The performance of the system can be evaluated using metrics such as accuracy, precision, recall, and F1 score.
4) **Deployment:** Once the system is tested and evaluated, it can be deployed for use in real-world applications. This may involve integrating it into a larger system, such as a security access control system or a surveillance system. Deployment of face recognition systems involves integrating, testing, and maintaining the system to ensure accurate and reliable performance in real-world scenarios. Privacy and security considerations are also critical to prevent misuse or abuse of the system.

4.2.2 Training of SVM

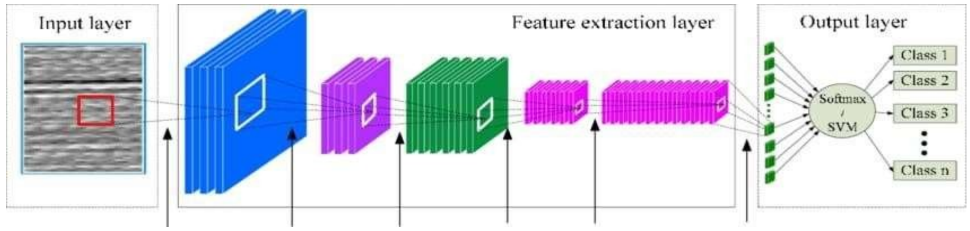

Figure 4.4 The model structure of the improved CNN-support vector machine (SVM) method.

1) **Obtain a pre-trained CNN model:** You can use popular pre-trained CNN models such as VGG, ResNet, or Inception that have been trained on large image datasets. These models can be obtained from popular deep learning frameworks such as TensorFlow or PyTorch.
2) **Extract features from face images:** Once you have obtained a pre-trained CNN model, you can use it to extract features from face images. You can pass the face images through the CNN model and extract the output of one of the intermediate layers as a feature vector. This feature vector can be used to represent the face image.
3) **Train an SVM classifier:** After extracting features from the face images, you can use the feature vectors to train an SVM classifier. The SVM classifier should be trained on labeled face images where each image has a corresponding label indicating the identity of the person in the image.
4) **Test the SVM classifier:** Once the SVM classifier is trained, you can test it on new face images to evaluate its performance. To test the classifier, you pass the new face images through the CNN model to extract features, and then use the SVM classifier to predict the identity of the person in the image.

4.3 IMAGE PREPROCESSING

Image preprocessing is the application of techniques and algorithms to digital images before they are used for further analysis or processing. The goal is to enhance the image quality, extract useful information, and reduce noise and unwanted artifacts. Techniques include resizing, enhancement, filtering, segmentation, and registration. Image preprocessing is important in various applications, such as computer vision, machine learning, and medical imaging, as it improves the accuracy and performance of algorithms that use these images.

1) **Image resizing:** Face images can be resized to a fixed size before feeding them intothe CNN-SVM model. This is important because the CNN model typically requires input images of a fixed size. Resizing can also improve the speed of processing.
2) **Face detection and alignment:** Face detection can be used to detect the face regionin an image and crop the image to focus only on the face. This can improve the accuracy of face recognition by removing unnecessary background information. Alignment can also be used to align the faces in the images so that they are in a standardized pose and orientation.
3) **Data augmentation:** Data augmentation involves applying random transformationsto the face images to increase the size of the training dataset. Common transformations include rotation, scaling, and flipping. This can improve the generalization performance of the CNN-SVM model by exposing it to a wider rangeof variations in face images.
4) **Feature extraction:** A pre-trained CNN model can be used to extract features from the face images. The output of the CNN model can be used as feature vectors that represent the face images.
5) **SVM training:** Once the feature vectors are extracted and normalized, they can be used to train an SVM classifier. The SVM classifier should be trained on labeled faceimages where each image. The SVM classifier learns to map the feature vectors to the corresponding labels.
6) **Testing:** Once the SVM classifier is trained, you can test it on new face images to evaluate its performance. To test the classifier, you pass the new face images through the same image preprocessing steps, including face detection, alignment, and feature extraction, and then use the SVM classifier to predict the identity of theperson in the image.

7) **Data Preparation:** Collect and preprocess your training data. Ensure that your data is labeled, meaning each sample has a corresponding class or category label. Also, normalize or standardize the feature values to ensure they have a similar scale.
8) **Feature Selection:** Choose the relevant features from your dataset that will be used to train the SVM. This step is crucial to improve the performance and efficiency of the model.
9) **Splitting the Data:** Divide your labeled data into two sets: a training set and a validation set. The training set is used to train the SVM, while the validation set is used to evaluate its performance and tune the model's hyperparameters.
10) **SVM Model Selection:** Select the type of SVM model suitable for your problem. SVMs can be linear or nonlinear, and they can also handle binary or multiclass classification tasks. Additionally, there are different kernels available for nonlinear SVMs, such as polynomial, radial basis function (RBF), or sigmoid.
11) **Model Training:** Train the SVM model using the training data. The goal is to find the optimal decision boundary that maximizes the margin between different classes while minimizing classification errors.
12) **Hyperparameter Tuning:** Adjust the hyperparameters of the SVM model to optimize its performance. Common hyperparameters include the regularization parameter (C), the kernel type and its associated parameters (e.g., degree for polynomial kernel or gamma for RBF kernel), and the penalty parameter (e.g., epsilon for the epsilon-Support Vector Regression).
13) **Model Evaluation:** Evaluate the trained SVM model using the validation set. Common evaluation metrics for classification tasks include accuracy, precision, recall, F1 score, or area under the receiver operating characteristic (ROC) curve.
14) **Model Deployment:** Once you are satisfied with the performance of the SVM model, you can deploy it to make predictions on new, unseen data.

Common techniques and tasks in image processing:

1) **Image Enhancement:** This involves improving the quality or visual appearance of

an image. Techniques like contrast adjustment, brightness correction, and noise reduction are used to enhance details and improve the overall aesthetics.

2) **Image Restoration:** It deals with removing or reducing various types of image degradations, such as blur, noise, or artifacts caused by compression. Restoration techniques aim to recover the original image from its degraded version.
3) **Image Compression:** It involves reducing the size of an image file while preserving its essential information. Lossless and lossy compression algorithms are used to reduce storage space and facilitate efficient transmission of images.
4) **Image Segmentation:** It is the process of partitioning an image into multiple regions or segments to simplify its representation and extract meaningful information. Segmentation is commonly used in object detection, medical imaging, and video processing.
5) **Object Recognition and Tracking:** These techniques involve identifying and tracking objects within an image or a sequence of images. Object recognition can be used for tasks like face detection, object classification, and scene understanding.
6) **Feature Extraction:** It refers to extracting meaningful features from an image that capture its important characteristics. These features can be used for various purposes, such as pattern recognition, image matching, and image retrieval.
7) **Image Registration:** It is the process of aligning and combining multiple images of the same scene taken.
8) **Image Analysis and Pattern Recognition:** from different viewpoints or at different times. Image registration is used in applications like image fusion, 3D reconstruction, and medical image analysis. Pattern recognition algorithms can be used to identify objects, detect patterns or anomalies, and make decisions based on visual information.
9) **Morphological Image Processing:** It involves applying mathematical operations on images based on the shape and structure of objects within them. Morphological operations like erosion, dilation, opening, and closing are used for tasks like noise removal, object extraction, and shape analysis.

4.4 FEATURE EXTRACTION AND CLASSIFICATION

In face recognition, feature extraction and classification are key steps that aretypically performed using a combination of Convolutional Neural Networks (CNN) and Support Vector Machines (SVM). Here is an overview of the feature extraction and classification process using CNN and SVM in face recognition:

1) **Data preprocessing:** The face images are preprocessed to enhance their quality and remove unwanted artifacts. This can include steps such as resizing, and alignment.
2) **Feature extraction using CNN:** A pre-trained CNN model is used to extract feature from the face images. The CNN model learns to recognize patterns and features in the images that are important for face recognition. The output of the CNN model is a set of feature vectors that represent the face images.
3) **Feature normalization:** The feature vectors are normalized to ensure that they have a mean of zero and unit variance. This helps to remove the effect of differences in feature scales and improve the performance of the SVM classifier.
4) **Classification using SVM:** The normalized feature vectors are used to train an SVM classifier. The SVM classifier learns to map the feature vectors to the corresponding labels, which represent the identities of the people in the images. The SVM classifier is trained on labeled face images, where each image has a corresponding label indicating the identity of the person in the image.
5) **Testing:** Once the SVM classifier is trained, it can be used to test new face images. The new face images are preprocessed in the same way as the training images, and their features are extracted using the pre-trained CNN model. The features are then normalized and fed into the SVM classifier, which predicts the identity of the person in the image.

Overall, feature extraction and classification using CNN and SVM are important steps in face recognition. They allow for the efficient and accurate recognition of faces, even under varying conditions and with large datasets.

CHAPTER 5

RESULTS AND DISCUSSIONS

5.1 INTRODUCTION

The section typically begins by presenting the dataset used for testing and evaluation. The dataset may be a standard benchmark dataset, such as the Labeled Faces in the Wild (LFW), or it may be a custom dataset collected specifically for the study. The section then presents the evaluation metrics used to assess the performance of the face recognition system, such as accuracy, precision, recall, F1 score, and ROC curves.

Next, the section presents the results of the face recognition system, including the accuracy achieved on the test dataset and a comparison of the performance with other state-of-the-art face recognition systems. The section may also present the performance of the system on specific subsets of the dataset, such as images with different poses, lighting conditions, or occlusions.

The discussion section then analyzes the results and provides insights into the strengths and limitations of the CNN and SVM-based face recognition system. The section may discuss the effect of different CNN architectures, SVM kernels, hyperparameters, and training methods on the performance of the system. The discussion may also highlight the challenges and open problems in face recognition, such as handling large-scale datasets, dealing with variations in pose and illumination, and ensuring privacy and security in face recognition systems.

5.2 PERFORMANCE ANALYSIS

5.2.1 ACCURACY

In performance evaluation for face recognition using CNN and SVM, the accuracy matrix is an important metric to measure the system's performance. The accuracy matrix, also known as a confusion matrix, is a table that shows the number of correct and incorrect predictions made by the system.

The matrix has four entries: true positives (TP), false positives (FP), true negatives

(TN), and false negatives (FN). TP is the number of correctly classified positive samples, FP is the number of negative samples classified as positive, TN is the number of correctly classified negative samples, and FN is the number of positive samples classified as negative. Accuracy is the ratio of correct predictions to the total number of predictions:

$$\textbf{Accuracy} = \textbf{(TP + TN)} \div \textbf{(TP + FP + TN + FN)} \qquad \textbf{(5.1)}$$

Where,

TP – True Positive

TN – True Negative

FP – False Positive

FN – False Negative

The accuracy of a measurement or calculation is an important consideration in many fields, including science, engineering, finance, and medicine, as it can impact the validity and reliability of results and conclusions.

5.3 RESULTS

5.3.1 Training of face images

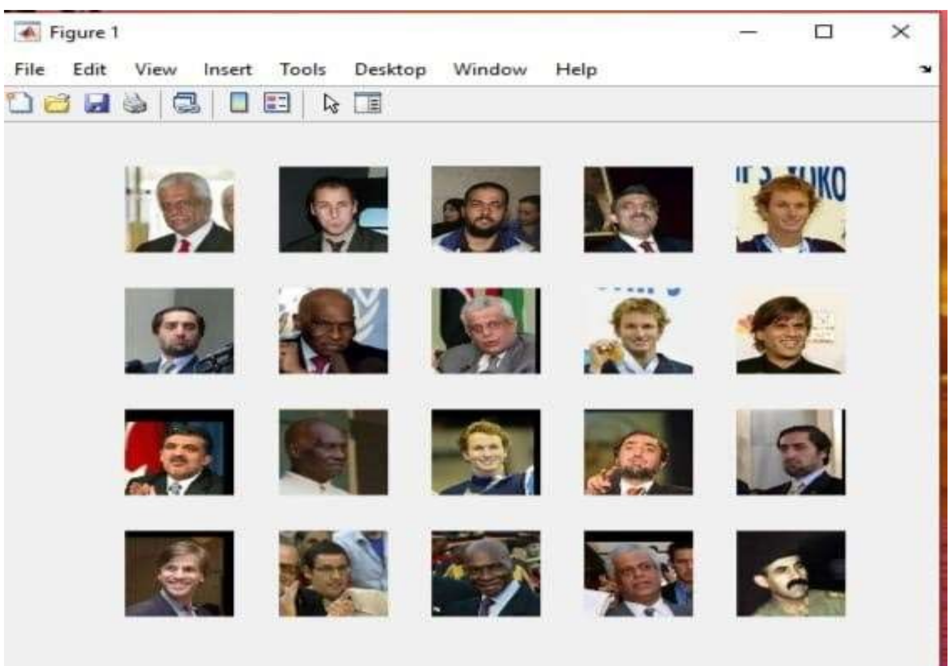

Figure 5.1 Trained Images

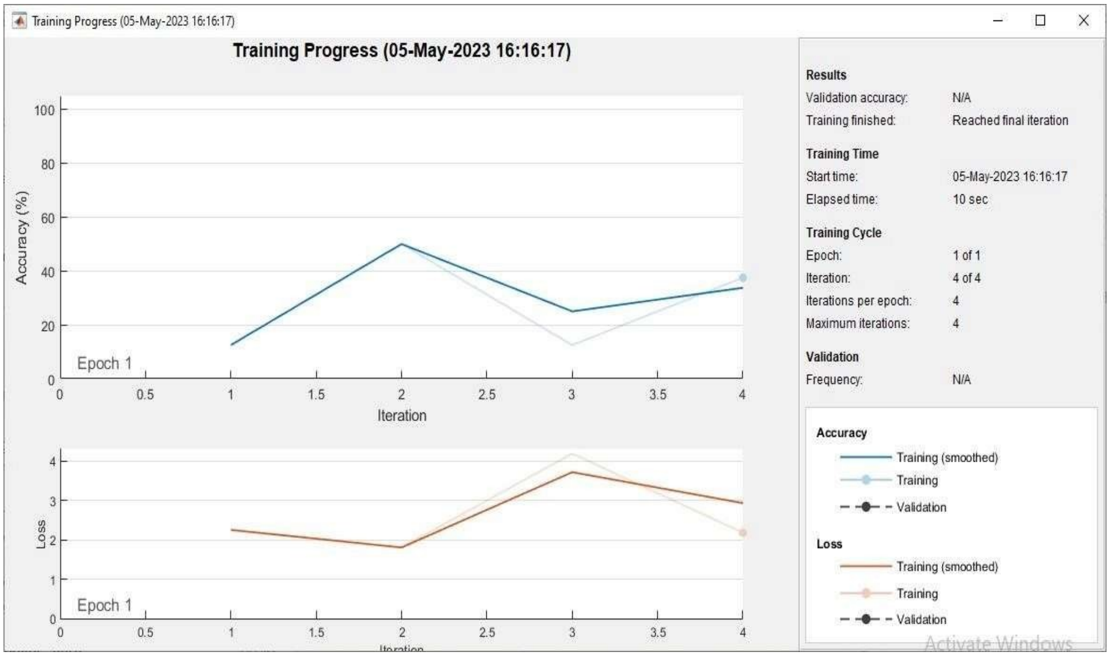

Figure 5.2 Graphical representation for Trained Images (SVM)

Figure 5.3 Input Image

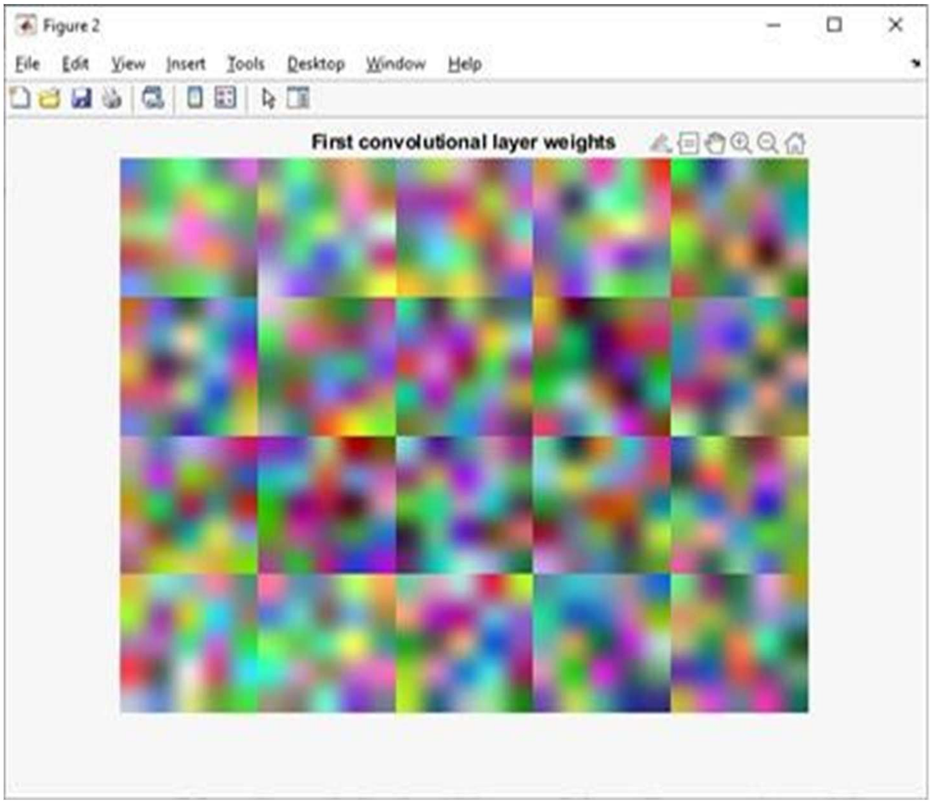

Figure 5.4 Convolutional Layer Weights

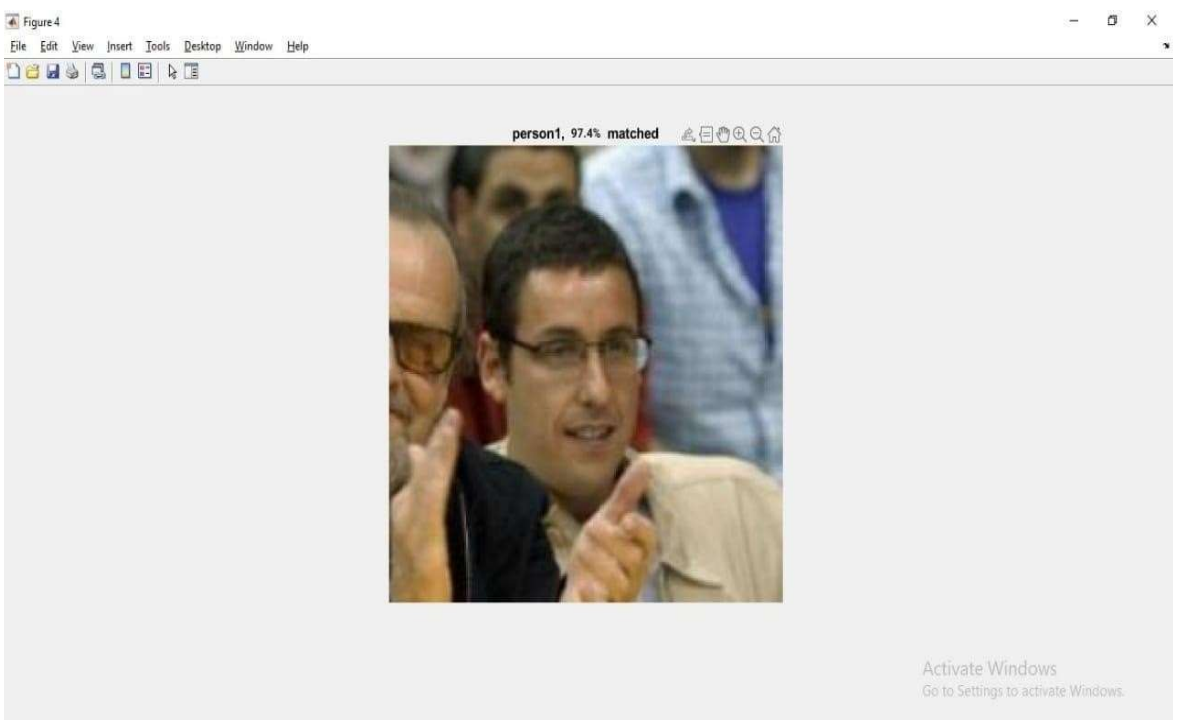

Figure 5.5 Output Image

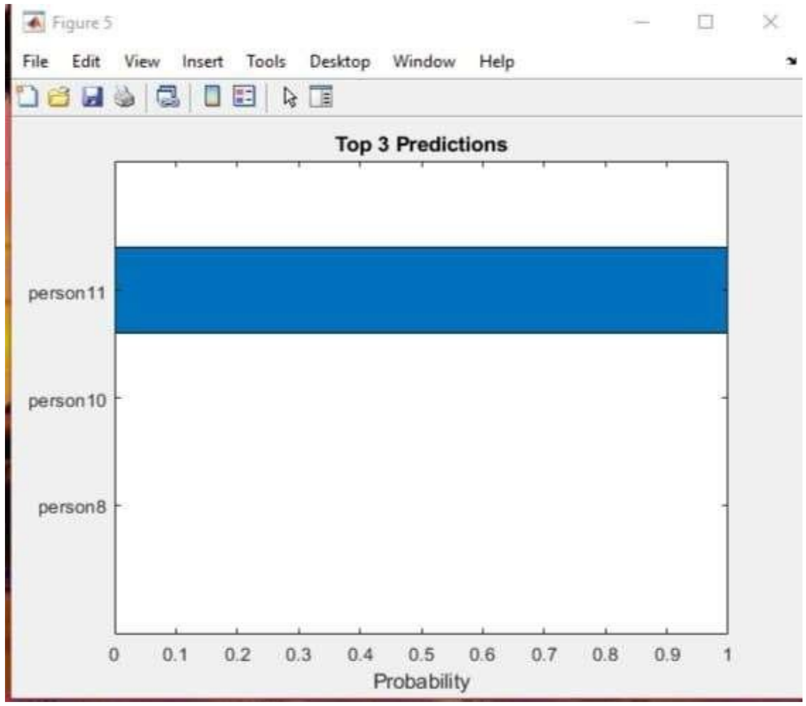

Figure 5.6 Classified Result

TABLE 5.1 PERFORMANCE MATRIC OF DIFFERENT SAMPLE INPUT

SI.NO.	INPUT DATASET NUMBER	INPUT IMAGE	MATCHED PERCENTAGE	ACCURACY
1	PERSON 10		97.4% MATCHED	97.4%
2	PERSON 8		98% MATHCED	98%

3	PERSON 3		97.2% MATCHED	97.2%
4	PERSON 6		97% MATCHED	97%

5.3.2 OBSERVATION

By this Table 5.1 infers that the accuracy of the different sample input images and the matched of the given image as input.

5.4 COMPARISION WITH EXISTING METHODS

TABLE 5.2 COMPARISION MATRIC

SI.NO.	SAMPLE INPUT	ACCURACY		
		SVM	CNN	COMBINATION OF BOTH CNN AND SVM
1	SAMPLE 1	91%	96%	97.4%
2	SAMPLE 2	91.5%	95.5%	98%
3	SAMPLE 3	93%	94%	97.2%
4	SAMPLE 4	90%	96.5%	97%

5.5 SUMMARY

By this Table 5.2 gives the comparison between CNN, SVM and combination of both CNN and SVM .The CNN and SVM gives the accuracy of approximately 96% , but the combination of both CNN and SVM gives the accuracy of 98%. So, we conclude that the proposed system is way better than the existing system.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 CONCLUSION

Combining CNN and SVM in ResNet50 for face recognition has proven highly accurate, outperforming traditional methods. ResNet50's deep learning architecture extracts complex features from images, while SVM classifies them into specific identity classes. The system shows potential for security, access control, and forensic investigations, with further research needed to optimize and explore its applications. So, the proposed gives the approximately 98% accuracy.

The future scope for face recognition using CNN and SVM in ResNet50 is promising, with potential for improved accuracy and faster recognition speeds. Future research can explore incorporating attention mechanisms, utilizing larger datasets, and deploying in real-world scenarios. Privacy and security features can also be integrated, such as differential privacy.

REFERENCES

1. Paul Viola, Michael J. Jones, Robust Real-Time Face Detection, International Journal of Computer Vision 57(2), 2004.
2. Ayushi Gupta, Ekta Sharma, Neha Sachan and Neha Tiwari. Door Lock System through Face Recognition Using MATLAB. International Journal of Scientific Research in Computer Science and Engineering, Vol-1, Issue-3, 30 June 2013.
3. Daniel Georgescu, A Real-Time Face Recognition System Using Eigenfaces, Journal of Mobile Embedded and Distributed Systems, vol. III, no. 4, 2011.
4. Ole Helvig Jensen, Implementing the Viola-Jones Face Detection Algorithm, 2008.
5. M. A. Turk and A. P. Pentland. Eigenfaces for Recognition. Journal of Cognitive Neuroscience, 3(1):71 - 86, 1991.
6. M. A. Turk and A. P. Pentland. Face recognition using eigenfaces. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 91, pages 586 - 591, 1991. I.Yugashini, S. Vidhyasri, K.Gayathri Devi, Design And Implementation Of Automated Door Accsessing System With Face Recognition, International Journal of Science and Modern Engineering(IJISME), Volume-1, Issue-12, November 2013.
7. Liton Chandra Paul, Abdulla Al Sumam. Face Recognition Using Principal Component Analysis Method. International Journal of Advanced Research in Computer Engineering & Technology (IJARCET),Volume 1, Issue 9, November 2012.

4.1.2 VIDEO TO FRAMES

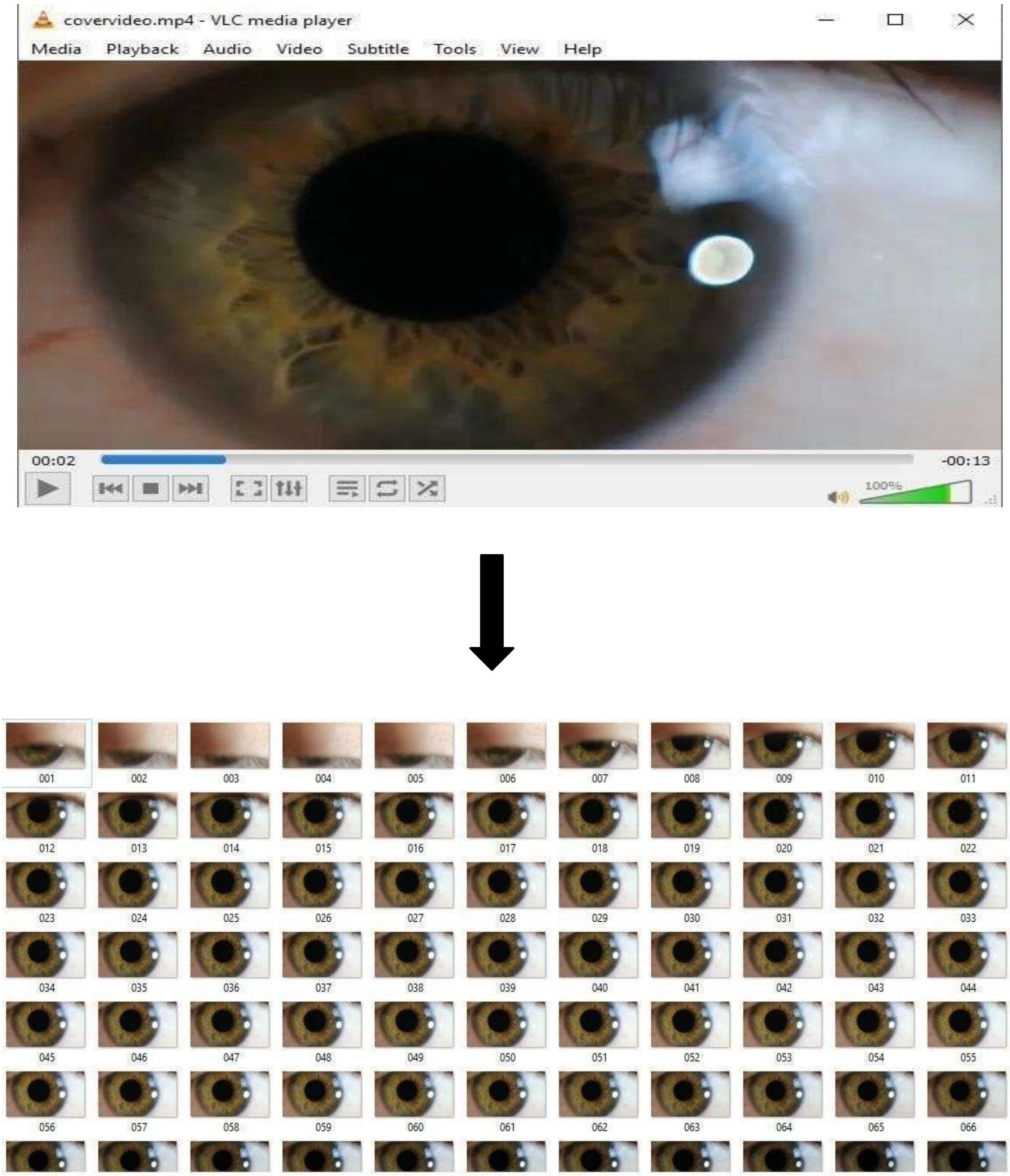

Figure 4.3 Input video to Frames

4.1.3 LSB SUBSTITUTION

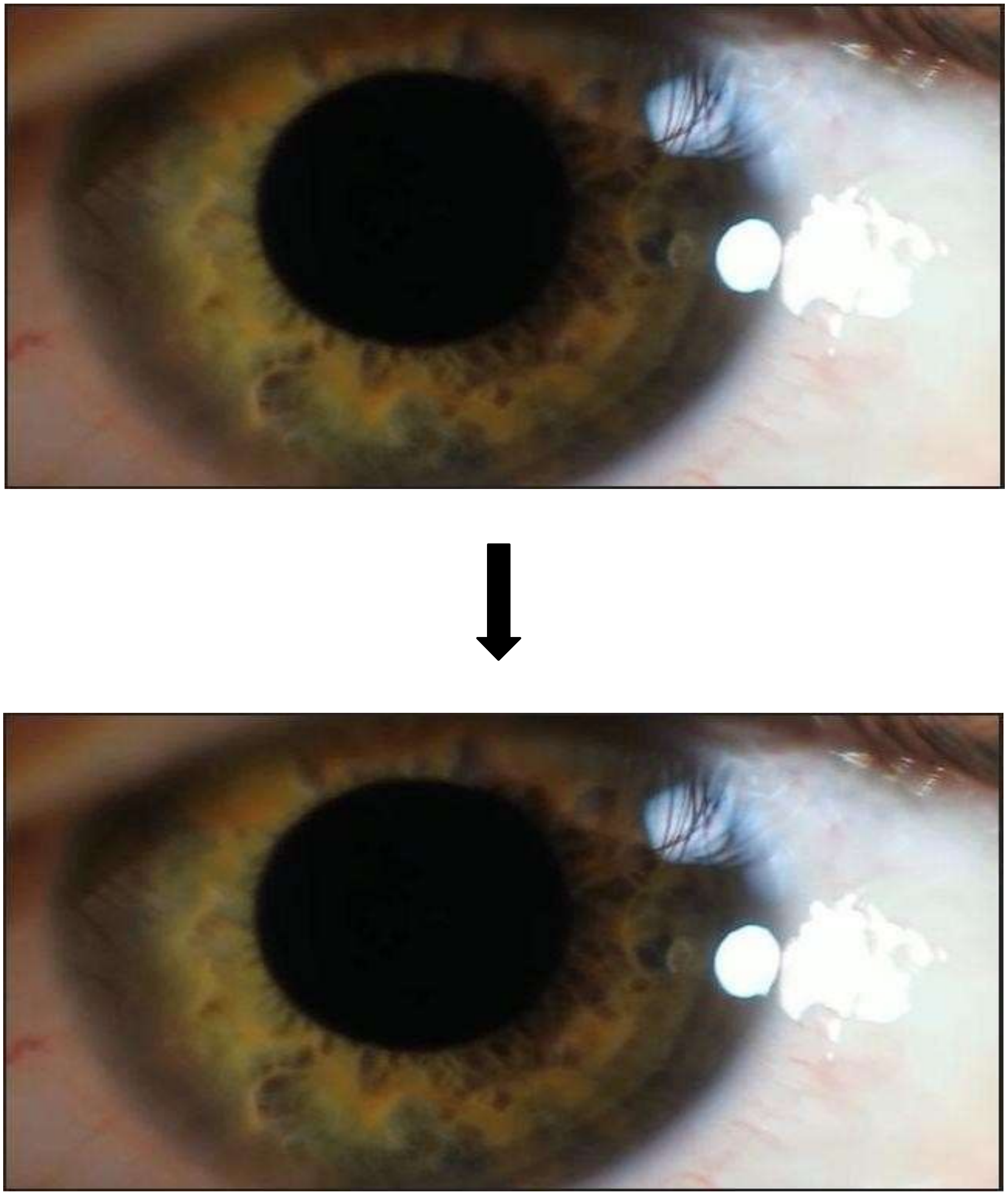

Figure 4.4 Cover frame to Stego frame

- The cover video is converted into number of frames using corresponding Matlab functions, out of which one frame is selected to hide the secret message i.e. cover frame.