

SIGNATURE VERIFICATION USING MATLAB - IMAGE PROCESSING

at

**Sathyabama Institute of Science and Technology
(Deemed to be University)**

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Technology Degree in Information Technology

By

M.SURUTHI (REG.NO.37120076)

M.THAMIZHARASI (REG.NO.37120078)



**DEPARTMENT OF INFORMATION TECHNOLOGY
SCHOOL OF COMPUTING
SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

**Accredited with Grade "A" by NAAC
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI – 600119. TAMIL NADU.**

MARCH - 2021



SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with "A" grade by NAAC

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai-600119

www.sathyabamauniversity.ac.in



DEPARTMENT OF INFORMATION TECHNOLOGY

BONAFIED CERTIFICATE

This is to certify that this Professional Training Report is the bonafied work of **M.SURUTHI(37120076)**, **M.THAMIZHARASI(37120078)** who carried out the project entitled "**SIGNATURE VERIFICATION USING MATLAB - IMAGE PROCESSING**" under my supervision from December 2020 to March 2021.

INTERNAL GUIDE

Ms.Vimali.J.S.,M.Tech.,(Ph.D).,

HEAD OF THE DEPARTMENT

DR.R. SUBHASHINI, M.E., Ph.D.,

Submitted for Viva Voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

I am M. Suruthi and M. Thamizharasi hereby declare that the Project Report on entitled **“SIGNATURE VERIFICATION USING MATLAB”** done by me under the guidance of Ms.Vimali.J.S.,M.Tech.,(Ph.D).,at Sathyabama Institute Of Science And Technology is submitted in partial fulfillment of the requirements for the award of Bachelor of technology degree in Information Technology.

DATE:

PLACE: Chennai

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGMENT

We are pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them

We convey my thanks to **Dr.T.Sasikala M.E., Ph.D., Dean**, School of Computing, **Dr.SUBHASHINI M.E.,Ph.D.**, Head of the Department of Information and Technology for providing me necessary support and details at the right time during the progressive reviews.

We would like to express y sincere and deep sense of gratitude to My Project Guide **Ms.Vimali.J.S.,M.Tech.,(Ph.D).**, for his valuable guidance, suggestion and encouragement paved way for the successful completion of my project work

We wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Information Technology** who were helpful in many ways for the completion of the project.

ABSTRACT

An efficient off-line signature verification method based on an interval symbolic representation and a fuzzy similarity measure is proposed. In the feature extraction step, a set of Local Binary Pattern (LBP) based features is computed from both the signature image and its under-sampled bitmap. Interval-valued symbolic data is then created for each feature in every signature class. As a result, a signature model composed of a set of interval values (corresponding to the number of features) is obtained for each individual's handwritten signature class. A novel fuzzy similarity measure is further proposed to compute the similarity between a test sample signature and the corresponding interval-valued symbolic model for the verification of the test sample. A comparison of our results with some recent signature verification methods available in the literature was provided in terms of average error rate and we noted that the proposed method always outperforms when the number of training samples is eight or more.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	V
	LIST OF FIGURES	Viii
	LIST OF ABBREVIATION	lx
1	INTRODUCTION	1
	1.1 INTRODUCTION TO THE PROJECT	1
	1.2 AIM OF THE PROJECT	2
	1.3 SCOPE OF THE PROJECT	2
2	LITERATURE SURVEY	3
	2.1 HANDWRITTEN SIGNATURE VERIFICATION	3
	2.2 OFFLINE SIGNATURE USING DTW	3
	2.3 OFFLINE SIGNATURE USING LOCAL AND INTEREST AND DESCRIPTORS	3
3	IMAGE PROCESSING SYSTEM	4
	3.1 DIGITIZER	4
	3.2 IMAGE PROCESSOR	5
	3.2.1 SEGMENTATION	5
	3.2.2 KNOWLEDGE BASE	6
	3.2.3 DIGITAL COMPUTER	6
	3.2.4 MASS STORAGE	6

3.2.5 HARD COPY DEVICE	6
3.2.6 OPERATOR CONSOLE	7
3.3 IMAGE PROCESSING FUNDAMENTAL	7
3.4 IMAGE PROCESSING TECHNIQUES	7
3.4.1 IMAGE ENHANCEMENTS	8
3.4.2 IMAGE RESTORATION	8
3.4.3 IMAGE ANALYSIS	9
3.4.4 IMAGE COMPRESSION	9
3.4.5 IMAGE SYNTHESIS	9
3.5 APPLICATIONS OF DIGITAL IMAGE PROCESSING	9
3.5.1 MEDICAL APPLICATION	10
3.5.2 SATELLITES IMAGING	10
3.5.3 COMMUNICATION	10
3.5.4 RADAR IMAGE SYSTEM	10
3.5.5 DIFFENCES AND INTELLIGENCE	11
3.6 FUNDAMENTAL OPERATION	11
3.6.1 DILATION AND EROSION	11
3.6.2 OPENING AND CLOSING	11
3.6.3 HIT AND MISS OPERATION	12
3.6.4 SUMMARY OF THE BASIC OPERATION	12

	3.6.5 CLASSIFICATION	13
4	METHODOLOGY	14
	4.1 SIGNATURE DETECTING	14
	4.2 EXISTING SYSTEM	14
	4.2.1 ADVANTAGE EXISTING SYSTEM	14
	4.3 PROPOSED SYETM	14
	4.3.1 ADVANTAGE PROPOSED SYSTEM	15
	4.4 SIGNATURE CLASSIFYING	15
	4.4.1 BLOCK DIAGRAM	16
	4.5 MODULE DESCRIPTION	16
	4.5.1 INPUT IMAGE	16
	4.5.2 PRE PROCESSING	17
	4.5.3 FEATURE EXTRATION	18
	4.5.4 LOCAL BINARY PATTERN	19
	4.6 CLASSIFIACTION	21
	4.6.1 SVM	21
	4.6.2 FUZZY SIMILARITY MEASURES	22
	4.7 SOFTWARE SPECIFICATION	23
	4.7.1 MATLAB PACKAGES	23
	4.7.2 FEATURE OF MATLAB	24
	4.7.3 DEVELOPMENT ENVIRONMENT	24

	4.7.4 ALGORITHM AND APPLICATION	24
	4.8 INTERFACING WITH OTHER LANGUAGES	25
	4.8.1 MATLAB EDITOR	25
	4.8.2 CODE ANALYZER	25
	4.8.3 MATLAB PROFILER	25
	4.8.4 DIRECTORY REPORTS	25
	4.8.5 DESIGNING GRAPHICAL USER INTERFACE	25
	4.9 ANALYZING AND ACCESSING DATA	26
	4.9.1 DATA ANALYSIS	26
	4.9.2 DATA ACCESS	26
	4.9.3 VISUALIZATION DATA	26
	4.9.4 2-D PLOTTING	27
	4.9.5 3-D PLOTTING AND VISUALIZATION	27
	4.9.6 PERFORMING NUMERIC COMPUTATION	27
	4.10 HARDWARE SPECIFICATION	28
5	RESULT AND DISCUSSION	30
	5.1 RESULT	30
	5.2 DISCUSSION	30
6	CONCLUSION AND FUTURE WORK	31

6.1 CONCLUSION	31
6.2 FUTURE WORK	31
REFERENCES	32
APPENDICES	34
A. SOUCE CODE	34
B. SCREENSHOTS	37
C. PLAGIARISM REPORT	41

LIST OF FIGURES

FIGURE NO	NAME OF THE FIGURE	PAGE NO
3.1	IMAGE PROCESSING SYSTEM	4
3.2	IMAGE PROCESSOR	5
3.4	IMAGE PROCESSING TECHNIQUES	8
3.6.1	DILATION AND EROSION	11
4.4.1	BLOCK DAIGRAM	16

LIST OF ABBREVIATION

QSWTS	QUALIFIED SIGNIFICANT WAVELET TREES
MATLAB	MATRIX LABORATORY
IDWT	INVERSE DISCRETE WAVELET TRANSFORM
C-PRBG	CHAOTIC PSEUDO-RANDOM BIT GENERATOR
LBP	LOCAL BINARY PATTERN
SVM	SUPPORT VECTOR MACHINES

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION OF THE PROJECT

Offline signature verification can be considered a special case of pattern reorganization. Biometric systems are extensively used to establish a person's identity in legal and administrative tasks. They are commonly modeled as Pattern Recognition systems, in which biometric data from an individual is acquired and stored as a "template" for future comparisons, or used to train a classifier that can discriminate if new samples belong to this user. They are useful in automatic verification of signature found on bank checks and documents. The objective of the offline signature verification system is to discriminate if a given signature is related to the user or not (genuine or forgery). Signature verification is a biometric verification which is an important research area targeted at automatic identity verification such as legal, banking and high security environments. Signature verification can be divided into two classes online and offline. Online approach uses a stylus and electronic tablet.

The term digital image 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 or complex numbers represented by a finite number of bits. An image given in the form of a transparency, slide, photograph or an X-ray is first digitized and stored as a matrix of binary digits in computer memory. This digitized image can then be processed and/or displayed on a high-resolution television monitor. For display, the image is stored in a rapid-access buffer memory, which refreshes the monitor at a rate of 25 frames per second to produce a visually continuous display.

1.2 AIM OF THE PROJECT

Biometric systems are extensively used to establish a person's identity in legal and administrative tasks. They are commonly modelled as Pattern Recognition systems, in which biometric data from an individual is acquired (e.g. during an enrolment process), and stored as a "template" for future comparisons, or used to train a classifier that can discriminate if new samples belong to this user. The reliability of these systems has security implications, and in the last decade these systems have been analysed from an Adversarial Machine Learning perspective. From this viewpoint, we consider an active adversary, with its own goals (e.g. getting access to a system), knowledge (e.g. knowing the classifier parameters, or the learning algorithm) and capabilities (e.g. ability to manipulate the training data, or the inputs during test time). In particular, characterize the different components of a biometric system that can be attacked. However, an emerging issue of "Adversarial Examples" pose new security concerns for such systems. This issue refers to adversarial input perturbations specially crafted to induce misclassifications. Those very small perturbations on images (almost imperceptible) could be crafted to mislead a state-of-the-art CNN-based classifier. Moreover, attacks crafted for one model often transfer to other models, meaning that an attacker could train its own surrogate classifier to generate attacks, as long as it has access to data from the same data distribution.

1.3 SCOPE OF THE PROJECT

This issue has been analyzed in many recent papers but the theoretical reasons are not fully understood, and most defenses are weak (i.e. they fail if the attacker knows about the defense). We evaluate this new threat for biometric systems, by characterizing the potential new attacks under a taxonomy of threats to such systems. We consider particular attack scenarios to Offline Handwritten Signature Verification, identifying the attacker's goals, required knowledge and capabilities.

CHAPTER 2

LITERATURE REVIEW

2.1 HANDWRITTEN SIGNATURE IDENTIFICATION USING BASIC CONCEPTS OF GRAPH THEORY (2010)

This paper presents previous work in the field of signature and writer identification to show the historical development of the idea and defines a new promising approach in handwritten signature identification based on some basic concepts of graph theory. This principle can be implemented on both on-line handwritten signature recognition systems and off-line handwritten signature recognition systems.

2.2 OFFLINE SIGNATURE VERIFICATION USING DTW (2017)

In this paper, we propose a signature verification system based on Dynamic Time Warping (DTW). The method works by extracting the vertical projection feature from signature images and by comparing reference and probe feature templates using elastic matching. Modifications are made to the basic DTW algorithm to account for the stability of the various components of a signature.

2.3 OFFLINE SIGNATURE VERIFICATION USING LOCAL INTEREST POINTS AND DESCRIPTORS (2018)

In this article, a new approach to offline signature verification, based on a general-purpose wide baseline matching methodology, is proposed. Instead of detecting and matching geometric, signature-dependent features, as it is usually done, in the proposed approach local interest points are detected in the signature images, then local descriptors are computed in the neighborhood of these points, and afterwards these descriptors are compared using local and global matching procedures

CHAPTER 3

IMAGE PROCESSING SYSTEM

3.1 DIGITIZER

A digitizer converts an image into a numerical representation suitable for input into a digital computer. Some common digitizers are

1. Microdensitometer
2. Flying spot scanner
3. Image dissector
4. Videocon camera
5. Photosensitive solid- state arrays.

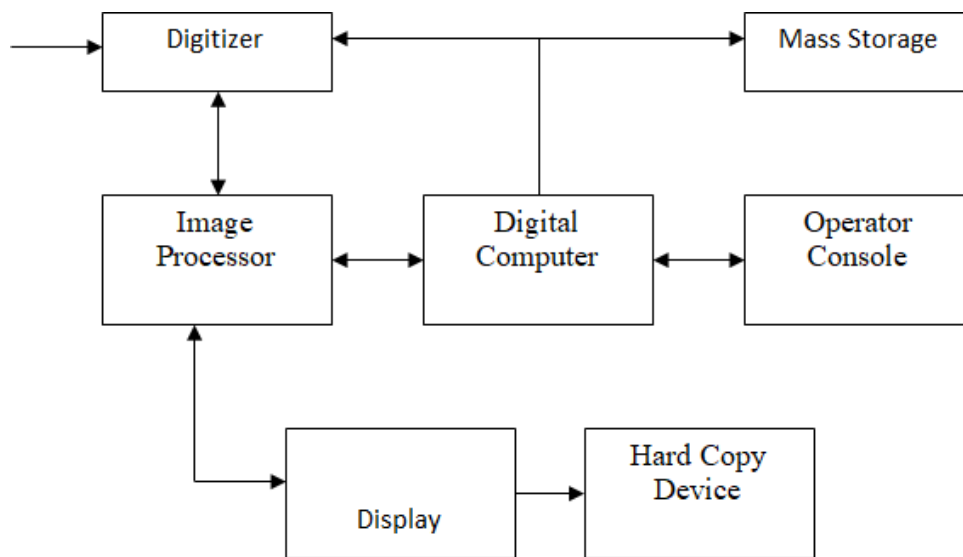


FIG: 3.1 IMAGE PROCESSING SYSTEM

3.2 IMAGE PROCESSOR

An image processor does the functions of image acquisition, storage, preprocessing, segmentation, representation, recognition and interpretation and finally displays or records the resulting image. The following block diagram gives the fundamental sequence involved in an image processing system.

As detailed in the diagram, the first step in the process is image acquisition by an imaging sensor in conjunction with a digitizer to digitize the image.

The next step is the preprocessing step where the image is improved being fed as an input to the other processes. Preprocessing typically deals with enhancing, removing noise, isolating regions, etc. Segmentation partitions an image into its constituent parts or objects.

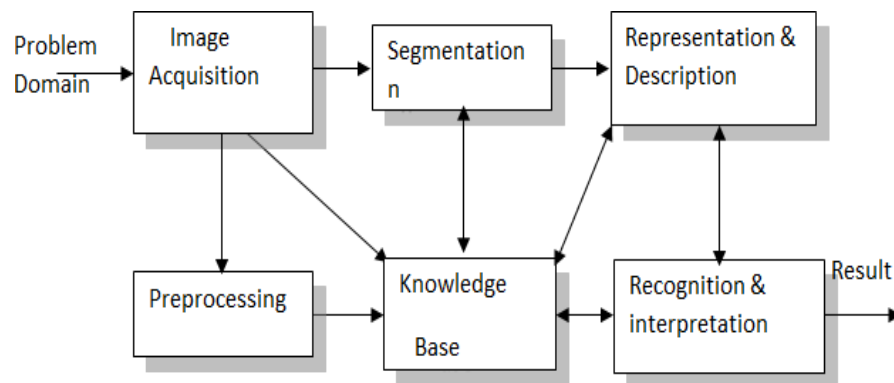


FIG: 3.2 IMAGE PROCESSOR

3.2.1 Segmentation

The output of segmentation is usually raw pixel data, which consists of either the boundary of the region or the pixels in the region themselves. Representation is the process of transforming the raw pixel data into a form useful for subsequent processing by the computer.

Description deals with extracting features that are basic in differentiating one class of objects from another. Recognition assigns a label to an object based on the information provided by its descriptors. Interpretation involves assigning meaning to an ensemble of recognized objects.

3.2.2 Knowledge Base

The knowledge about a problem domain is incorporated into the knowledge base. The knowledge base guides the operation of each processing module and also controls the interaction between the modules. Not all modules need be necessarily present for a specific function. The composition of the image processing system depends on its application. The frame rate of the image processor is normally around 25 frames per second.

3.2.3 Digital Computer

Mathematical processing of the digitized image such as convolution, averaging, addition, subtraction, etc. are done by the computer. Digital signatures are a standard element of most cryptographic protocol suites, and are commonly used for software distribution, financial transactions, contract management software, and in other cases where it is important to detect forgery or tampering.

Digital signatures are equivalent to traditional handwritten signatures in many respects, but properly implemented digital signatures are more difficult to forge than the handwritten type. Digital signature schemes, in the sense used here, are cryptographically based, and must be implemented properly to be effective

3.2.4 Mass Storage

The secondary storage devices normally used are floppy disks, CD ROMs etc. The number of required signature cards depends on the time period during which a defined amount of files is supposed to be signed.

3.2.5 Hard Copy Device

The hard copy device is used to produce a permanent copy of the image and for the storage of the software involved. A method of authentication and non-repudiation of hard-copy documents includes affixing a physical manifestation of a digital signature to a hard-copy document.

The physical manifestation of a digital signature is converted to an electronic digital signature, which is compared to a public key to authenticate the hard-copy document. Description deals with extracting features that are basic in differentiating one class of objects from another.

3.2.6 Operator Console

The operator console consists of equipment and arrangements for verification of intermediate results and for alterations in the software as and when require. The operator is also capable of checking for any resulting errors and for the entry of requisite data.

3.3 IMAGE PROCESSING FUNDAMENTAL

Digital image processing refers processing of the image in digital form. Modern cameras may directly take the image in digital form but generally images are originated in optical form.

They are captured by video cameras and digitalized. The digitalization process includes sampling, quantization. Then these images are processed by the five fundamental processes, at least any one of them, not necessarily all of them.

3.4 IMAGE PROCESSING TECHNIQUES

Digital image processing is the use of a digital computer to process digital images through an algorithm. As a subcategory or field of digital signal processing, digital image processing has many advantages over analog image processing.

It allows a much wider range of algorithm to be applied to the input data and can avoid problems such as the build-up of noise and distortion during processing. Since images are defined over two dimensions digital image processing may be modelled in the form of multi dimension systems.

Images acquired by satellites are useful in tracking of earth resources; geographical mapping; prediction of agricultural crops, urban growth and weather; flood and fire control; and many other environmental applications. Space image applications include recognition and analysis of objects contained in image obtained from deep space-probe missions.

Some techniques are used in digital image processing include:

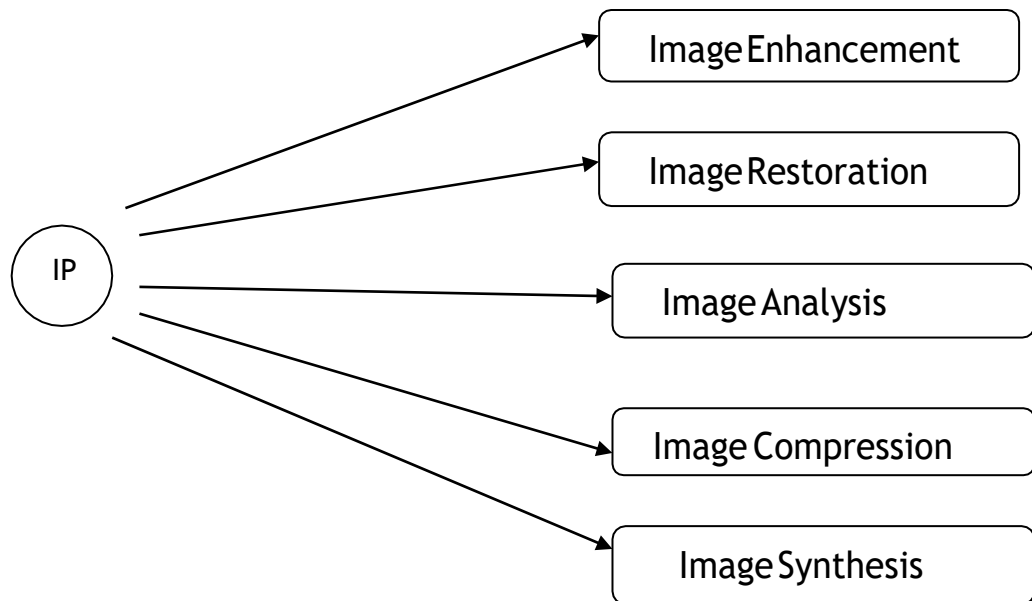


FIG: 3.4 IMAGE PROCESSING TECHNIQUES

3.4.1 Image Enhancement

Image enhancement operations improve the qualities of an image like improving the image's contrast and brightness characteristics, reducing its noise content, or sharpen the details. This just enhances the image and reveals the same information in more understandable image. It does not add any information to it.

3.4.2 Image Restoration

Image restoration like enhancement improves the qualities of image but all the operations are mainly based on known, measured, or degradations of the original image.

Image restorations are used to restore images with problems such as geometric distortion, improper focus, repetitive noise, and camera motion. It is used to correct images for known degradations.

3.4.3 Image Analysis

Image analysis operations produce numerical or graphical information based on characteristics of the original image. They break into objects and then classify them.

They depend on the image statistics. Common operations are extraction and description of scene and image features, automated measurements, and object classification. Image analyze are mainly used in machine vision applications.

3.4.4 Image Compression

Image compression and decompression reduce the data content necessary to describe the image. Most of the images contain lot of redundant information, compression removes all the redundancies. Because of the compression the size is reduced, so efficiently stored or transported.

The compressed image is decompressed when displayed. Lossless compression preserves the exact data in the original image, but Lossy compression does not represent the original image but provide excellent compression.

3.4.5 Image Synthesis

Image synthesis operations create images from other images or non-image data. Image synthesis operations generally create images that are either physically impossible or impractical to acquire.

3.5 APPLICATIONS OF DIGITAL IMAGE PROCESSING

Digital image processing has a broad spectrum of applications, such as remote sensing via satellites and other spacecrafts, image transmission and storage for business applications, medical processing, radar, sonar and acoustic image processing, robotics and automated inspection of industrial parts.

The proposed method provided significantly improved results compared to the state-of-the-art methods considering two different off-line signature datasets.

The main advantage of the proposed model is that it allows the design and integration of a model for a new individual using only genuine signatures with the same parameters as before, without any need of re-tuning all the parameters.

3.5.1 Medical Applications

In medical applications, one is concerned with processing of chest X-rays, cineangiograms, projection images of Tran's axial tomography and other medical images that occur in radiology, nuclear magnetic resonance (NMR) and ultrasonic scanning. These images may be used for patient screening and monitoring or for detection of tumours' or other disease in patients.

3.5.2 Satellites Imaging

Images acquired by satellites are useful in tracking of earth resources; geographical mapping; prediction of agricultural crops, urban growth and weather; flood and fire control; and many other environmental applications. Space image applications include recognition and analysis of objects contained in image obtained from deep space-probe missions.

3.5.3 Communication

Image transmission and storage applications occur in broadcast television, teleconferencing, and transmission of facsimile images for office automation, communication of computer networks, closed-circuit television based security monitoring systems and in military communications.

3.5.4 Radar Image System

Radar and sonar images are used for detection and recognition of various types of targets or in guidance and manoeuvring of aircraft or missile systems. It is used in scanning, and transmission for converting paper documents to a digital image form, compressing the image, and storing it on magnetic tape. It is also used in document reading for automatically detecting and recognizing printed characteristics.

3.5.5 Defences and Intelligence

It is used in reconnaissance photo-interpretation for automatic interpretation of earth satellite imagery to look for sensitive targets or military threats and target acquisition and guidance for recognizing and tracking targets in real-time smart-bomb and missile-guidance systems.

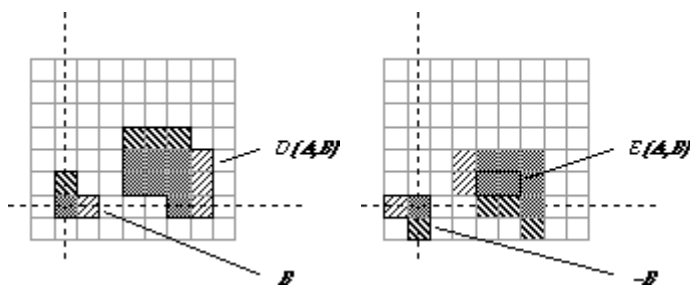
3.6 FUNDAMENTAL OPERATION

The fundamental operations associated with an object are the standard set operations union, intersection, and complement plus translation. Translation - Given a vector x and a set A , the translation, $A + x$, is defined as an image as an (amplitude) function of two, real (coordinate) variables $a(x,y)$ or two, discrete variables $a[m,n]$.

An alternative definition of an image can be based on the notion that an image consists of a set (or collection) of either continuous or discrete coordinates. In a sense the set corresponds to the points or pixels that belong to the objects in the image.

3.6.1 Dilation and Erosion

From these two operations we define the fundamental mathematical morphology operations dilation and erosion. These two operations are illustrated in figures below for the objects defined



A binary image containing two object sets A and B . The three pixels in B are "color-coded" as is their effect in the result.

3.6.2 Opening and Closing

We can combine dilation and erosion to build two important higher order operations. For the opening with structuring element B and images A , A_1 and A_2 where A_1 is a subimage of $(A_1 A_2)$. For the opening with structuring

element B and images A, $A \ominus B_1$ and $A \ominus B_2$, they can be considered as the reason for defining erosion with $-B$ instead of B.

3.6.3 Hit and Miss Operations

The hit-or-miss operator was defined by Serra but we shall refer to it as the hit-and-miss operator and define it as follows. Given an image A and two structuring elements B_1 and B_2 , the set definition Boolean definition are where B_1 and B_2 are bounded, disjoint structuring elements. Note the use of the notation from two sets are disjoint if $B_1 \cap B_2 = \emptyset$, the empty set.

In an important sense the hit-and-miss operator is the morphological equivalent of template matching, a well-known technique for matching patterns based upon cross-correlation. Here, we have a template B_1 for the object and a template B_2 for the background.

3.6.4 Summary of the Basic Operations

The results of the application of these basic operations on a test image are illustrated below. The various structuring elements used in the processing are defined. The value "-" indicates a "don't care". All three structuring elements are symmetric. The opening operation can separate objects that are connected in a binary image.

The closing operation can fill in small holes. Both operations generate a certain amount of smoothing on an object contour given a "smooth" structuring element. The opening smooths from the inside of the object contour and the closing smooths from the outside of the object contour. The hit-and-miss example has found the 4-connected contour pixels. An alternative method to find the contour is simply to use the relation

3.6.5 Classification

In machine learning and statistics, classification is the problem of identifying to which of a set of categories (sub-populations) a new observation belongs, on the basis of a training set of data containing observations or instances.

- Examples are assigning a given email to the "spam" or "non-spam" class, and assigning a diagnosis to a given patient based on observed characteristics of the patient (sex, blood pressure, presence or absence of certain symptoms, etc).
- Other classifiers work by comparing observations to previous observations by means of a similarity or distance function. An algorithm that implements classification, especially in a concrete implementation, is known as a classifier.
- The term "classifier" sometimes also refers to the mathematical function, implemented by a classification algorithm, that maps input data to a category.
- Terminology across fields is quite varied. In statistics, where classification is often done with logistic regression or a similar procedure, the properties of observations are termed explanatory variables (or independent variables, regressors, etc).
- The categories to be predicted are known as outcomes, which are considered to be possible values of the dependent variable. In machine learning, the observations are often known as instances, the explanatory variables are termed features (grouped into a feature vector).

CHAPTER 4

METHODOLOGY

4.1 SIGNATURE DETECTING

The main objective of this project is to detecting and classifying the fake and real handwritten signature verification using different image processing techniques. The objective of signature verification systems is to discriminate if a given signature is genuine (produced by the claimed individual), or a forgery (produced by an impostor). ... The aim of such systems is to recognize a person based on physiological or behavioral traits.

4.2 EXISTING SYSTEM

In this paper they investigated the impact of adversarial examples on biometric systems, in particular by identifying threats to Offline Handwritten Signature Verification under the point of view of Adversarial Machine Learning. Their experiments indicate that the issue of adversarial examples presents new threats to such systems in several scenarios, including both systems using handcrafted feature extractors and systems that learn directly from image pixels.

4.2.1 *Existing System Advantage*

- Accuracy is low.
- Due to low accuracy the system gives wrong output

4.3 PROPOSED METHOD

In this paper an efficient off-line signature verification method based on an interval symbolic representation and a fuzzy similarity measure is proposed. In the feature extraction step, a set of Local Binary Pattern (LBP) based features is computed from both the signature image and its under-sampled bitmap.

Interval-valued symbolic data is then created for each feature in every signature class. As a result, a signature model composed of a set of interval values (corresponding to the number of features) is obtained for each individual's handwritten signature class.

A novel fuzzy similarity measure is further proposed to compute the similarity between a test sample signature and the corresponding interval-valued symbolic model for the verification of the test sample.

4.3.1 Proposed System Advantage

- The proposed method provided significantly improved results compared to the state-of-the-art methods considering two different off-line signature datasets.

4.4 SIGNATURE CLASSIFYING

Signature verification is a biometric verification which is an important research area targeted at automatic identity verification such as legal, banking and high security environments. Signature verification can be divided into two classes online and offline.

Online approach uses a stylus and electronic tablet. A method of offline signature verification based on the bi-interval valued symbolic representation is presented. A novel fuzzy similarity measure is further proposed to compute the similarity between a test sample signature the corresponding interval-valued symbolic model for the verification of the test sample.

4.4.1 Block Diagram

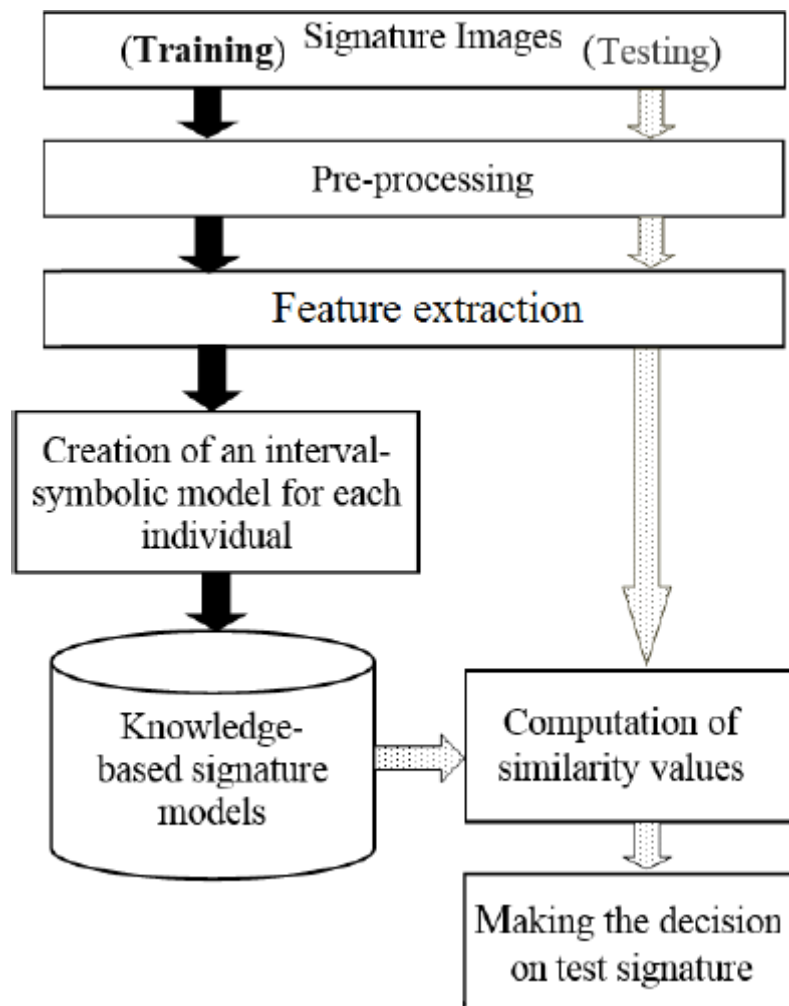


FIG: 4.4.1 BLOCK DAIGRAM

4.5 MODULE DESCRIPTION

4.5.1 Input Image

Read and Display an input Image. Read an image into the workspace, using the read command or camera. In image processing, it is defined as the action of retrieving an image from some source, usually a hardware-based source for processing. It is the first step in the workflow sequence because, without an image, no processing is possible. The image that is acquired is completely unprocessed.

4.5.2 Pre-processing

Pre-processing is a common name for operations with images at the lowest level of abstraction both input and output are intensity images. The aim of pre-processing is an improvement of the image data that suppresses unwanted distortions or enhances some image features important for further processing. Image pre-processing methods use the considerable redundancy in images. Neighboring pixels corresponding to one object in real images have essentially the same or similar brightness value. Thus distorted pixel can often be restored as an average value of neighboring pixels.

➤ Resizing the Input Image

All the input images are resized into same dimensions. If the specified size does not produce the same aspect ratio as the input image, the output image will be distorted.

➤ Converting the Color Format

For many applications of image processing, color information doesn't help us. If you get into the business of attempting to distinguish colors from one another, then one reason for converting RGB image to BLACK AND WHITE or GRAYSCALE formats in image.

➤ RGB Color

The RGB color model is an additive color model which red, green, and blue light are added together in various ways to reproduce a broad array of colors. The name of the model comes from the initials of the three additive primary colors, red, green, and blue. RGB is a device-dependent color model different devices detect or reproduce a given RGB value differently, since the color elements (such as phosphors or dyes).

Their response to the individual R, G, and B levels vary from manufacturer to manufacturer, or even in the same device over time. Thus an RGB value does not define the same color across devices without some kind of color management.



RGB color image

➤ **Grayscale**

In photography and computing, a grayscale or grayscale digital image is an image in which the value of each pixel is a single sample, that is, it carries only intensity information. Grayscale images are distinct from one-bit bi-tonal black-and-white images.



Gray scale color image

4.5.3 Feature Extration

In machine learning, pattern recognition and in image processing, feature extraction starts from an initial set of measured data and builds derived values (features) intended to be informative and non-redundant, facilitating the subsequent learning and generalization steps, and in some cases leading to better human interpretations. Feature extraction is related to dimensionality reduction.

- When the input data to an algorithm is too large to be processed and it is suspected to be redundant (e.g. the same measurement in both feet and meters, or the repetitiveness of images presented as pixels), then it can be transformed into a reduced set of features (also named a feature vector).
- Determining a subset of the initial features is called feature selection. The selected features are expected to contain the relevant information from the input data, so that the desired task can be performed by using this reduced representation instead of the complete initial data.
- In the feature extraction step, a set of Local Binary Pattern (LBP) based features is computed from both the signature image and its under-sampled bitmap. Interval-valued symbolic data is then created for each feature in every signature class.
- As a result, a signature model composed of a set of interval values (corresponding to the number of features) is obtained for each individual's handwritten signature class.
- A novel fuzzy similarity measure is further proposed to compute the similarity between a test sample signature and the corresponding interval-valued symbolic model for the verification of the test sample.

4.5.4 Local Binary Pattern

LBP features compute co-occurrence of pixel values in predetermined neighborhoods. LBP method is commonly used in object recognition with good success and we expected it also to be useful in offline signature verification

- Furthermore, since LBP is a texture feature, we expected it to be complementary to the HOG features that are also used in this thesis.
- An example work that combines HOG and LBP features successfully was designed to detect partially occluded humans in scenes [97], reaching the best human detection performance on the INRIA dataset.
- LBP features are used in signature verification as well. In this thesis, LBP features are extracted only in Cartesian coordinates. We utilized the LBP method by different approaches explained below

- The LBP operator was extended to use neighborhoods of different sizes (Ojala et al. 2002). Using a circular neighborhood and bilinearly interpolating values at non-integer pixel coordinates allow any radius and number of pixels in the neighborhood.
- The gray scale variance of the local neighborhood can be used as the complementary contrast measure. In the following, the notation (P,R) will be used for pixel neighborhoods which means P sampling points on a circle of radius of R.
- In the LBP approach for texture classification, the occurrences of the LBP codes in an image are collected into a histogram. The classification is then performed by computing simple histogram similarities.
- However, considering a similar approach for facial image representation results in a loss of spatial information and therefore one should codify the texture information while retaining also their locations.
- One way to achieve this goal is to use the LBP texture descriptors to build several local descriptions of the face and combine them into a global description. Such local descriptions have been gaining interest lately which is understandable given the limitations of the holistic representations.
- These local feature based methods are more robust against variations in pose or illumination than holistic methods.
- (LBP) is a simple yet very efficient texture operator which labels the pixels of an image by thresholding the neighborhood of each pixel and considers the result as a binary number.
- Due to its discriminative power and computational simplicity, LBP texture operator has become a popular approach in various applications. It can be seen as a unifying approach to the traditionally divergent statistical and structural models of texture analysis.
- Perhaps the most important property of the LBP operator in real-world applications is its robustness to monotonic gray-scale changes caused, for example, by illumination variations.
- Another important property is its computational simplicity, which makes it possible to analyze images in challenging real-time settings. The basic idea for developing the LBP operator was that two-dimensional surface textures can be described by two complementary measures.

4.6 CLASSIFICATION

Image classification refers to the task of extracting information classes from a multiband raster image. The resulting raster from image classification can be used to create thematic maps.

The recommended way to perform classification and multivariate analysis is through the Image Classification toolbar. There are many classification algorithms available and some classification algorithm that are given below,

4.6.1 SVM (*Support Vector Machine Classification*)

- The approach proposed given Compared to the previous year, the results were slightly higher. The most up-to-date approaches when it comes to Offline signature datasets from two separate sources.
- QSWTS provide both invisibility and significant resistance against lossy transmission and compression. The inverse discrete wavelet transform (IDWT) is applied to provide the stego-object.
- The proposed model's main benefit is that it facilitates the development and incorporation of a model for a new person using just a few lines of code.
- Genuine signatures that are similar parameters as before, without the need for any changes many of the parameters are being re-tuned.
- In the SVM classifier, it is easy to have a linear hyper-plane between these two classes. But, another burning question which arises is, should we need to add this feature manually to have a hyper-plane. No, the SVM algorithm has a technique called the kernel trick.
- The SVM kernel is a function that takes low dimensional input space and transforms it to a higher dimensional space i.e. it converts not separable problem to separable problem.
- It is mostly useful in non-linear separation problem. Simply put, it does some extremely complex data transformations, then finds out the process to separate the data based on the labels or outputs you've defined.
- In Python, scikit-learn is a widely used library for implementing machine learning algorithms. SVM is also available in the scikit-learn library and we follow the same structure for using it(Import library, object creation, fitting model and prediction).

- Now, let us have a look at a real-life problem statement and dataset to understand how to apply SVM for classification
- Support Vector Machine” (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems.
- In the SVM algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate.
- we perform classification by finding the hyper-plane that differentiates the two classes very well (look at the below snapshot).

4.6.2 Fuzzy Similarity Measures

- One of the most common BTP methods is the crypto-biometric FV scheme. This proposal is provided as an example.
- A cryptographic construction that is error-tolerant which an unsorted set (e.g. a list of) biometric characteristics) are used to encrypt/decrypt a hidden key for the purpose of accessing an The vault is unbreakable.
- This method not only protects the key while also protecting it to the haphazardly organised set The FV school district A user-specific key K , of length M bits, is randomly generated to protect a biometric feature set (a_1, a_2, \dots, a_n) . R is an error-correcting code.
- The Solomon (RS) method is used on K and N . The provided redundancy is concatenated to K , resulting in an encoded key K of length N . Bits (NM) is a unit of measurement for the number of bits in a So there's a degree polynomial P .
- The letter L (being $L n$) is constructed. Using K as a set of coefficients The polynomial projections a_1, \dots, a_2 are determined for each element of A , yielding a set of genuine polynomial projections.
- The fuzzy similarity measure (distance measure) is a measure that depicts the closeness (difference) among fuzzy sets. It proposed the Pythagorean fuzzy similarity measures for dealing the multi-attribute decision-making problems.

- Though the methods discussed in the previous section could predict the similarity of fuzzy numbers they fail to correctly give the similarity measure in certain situations.
- Here we present a new similarity measure based on fuzzy difference of distance of points of fuzzy numbers rather than geometric distances used by the existing methods.
- We see that from pattern sets given in section 5 the current fuzzy similarity measure not only overcomes the drawback of the earlier methods it also gives the similarity measure with better accuracy
- The fuzzy similarity measure presented here satisfies other properties which reduces the computational work. The relevant properties we consider for the similarity measures depend on the usefulness within the domain of research but they are not considered as complete.

4.7 SOFTWARE SPECIFICATION

4.7.1 Matlab

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 can be saved in a text file, typically using the MATLAB Editor, as a script or encapsulated into a function, extending the commands available.

4.7.2 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.
- Mathematical functions for linear algebra, statistics, Fourier analysis, filtering, optimization, and numerical integration.
- 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++, Fortran, Java™, COM, and Microsoft Excel.
- 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 particular classes of problems in these application areas.

4.7.3 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.

4.7.4 Algorithm and Applications

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

4.8 INTERFACING WITH OTHER LANGUAGES

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**).

4.8.1 *Matlab Editor*

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

4.8.2 *Code Analyzer*

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

4.8.3 *Matlab Profiler*

Records the time spent executing each line of code. 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.

For fast execution of heavy matrix and vector computations, MATLAB uses processor-optimized libraries. For general-purpose scalar computations, MATLAB generates machine-code instructions using its JIT (Just-In-Time) compilation technology.

4.8.4 *Directory Reports*

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

4.8.5 *Desining Graphical User Interface*

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 ActiveX[®] controls. Alternatively, you can create GUIs programmatically using MATLAB functions.

4.9 ANALYZING AND ACCESSING DATA

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.

4.9.1 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

4.9.2 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.

4.9.3 Visualization 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.

4.9.4 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.

4.9.5 3-D Plotting and 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.

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

4.9.6 Performing Numeric Computation

MATLAB contains mathematical, statistical, and engineering functions to support all common engineering and science operations. These functions, developed 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.

4.10 HADWARE SPECIFICATION

- Data Exploration, Acquisition, Analyzing and Visualization Engineering drawing and Scientific graphics Analyzing of algorithmic designing and development Mathematical functions and Computational functions Simulating problems prototyping and modeling Application development programming using GUI building environment.
- Using MATLAB, you can solve technical computing problems faster than with traditional programming languages, such as C, C++ and FORTRAN additionally this toolbox supports offloading computationally intensive workloads to the campus compute cluster.
- MATLAB is one of a few languages in which each variable is a matrix (broadly construed) and "knows" how big it is. Moreover, the fundamental operators (e.g. addition, multiplication) are programmed to deal with matrices when required.
- The MATLAB environment handles much of the bothersome housekeeping that makes all this possible.

- 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.
- Since so many of the procedures required for Macro-Investment Analysis involves matrices, MATLAB proves to be an extremely efficient language for both communication and implementation.
- Processor : Pentium Dual Core 2.00GHZ
- Hard Disk : 500 GB
- RAM : 4GB (minimum)
- Keyboard : 110 keys enhanced

CHAPTER 5

RESULT AND DISCUSSION

5.1 RESULT

In this paper we investigated the impact of an off-line signature verification method based on an interval symbolic representation and a fuzzy similarity measure is proposed. In the feature extraction step, a set of Local Binary Pattern (LBP) based features is computed from both the signature image and its under-sampled bitmap. Interval-valued symbolic data is then created for each feature in every signature class.

5.2 DISCUSSION

As a result, a signature model composed of a set of interval values is obtained for each individual's handwritten signature class. A comparison of our results with some recent signature verification methods available in the literature was provided in terms of average error rate. we noted that the proposed method always outperforms when the number of training samples is eight or more. On the other hand, the global SVM improves the performance when used in conjunction with user SVMs. Classifier combination is applied at score level to combine the decisions of the six classifiers.

CHAPTER 6

CONCLUSION AND FUTUREWORK

6.1 CONCLUSION

In this paper we investigated the impact of an off-line signature verification method based on an interval symbolic representation and a fuzzy similarity measure is proposed. In the feature extraction step, a set of Local Binary Pattern (LBP) based features is computed from both the signature image and its under-sampled bitmap. Interval-valued symbolic data is then created for each feature in every signature class.

As a result, a signature model composed of a set of interval values (corresponding to the number of features) is obtained for each individual's handwritten signature class. A comparison of our results with some recent signature verification methods available in the literature was provided in terms of average error rate and we noted that the proposed method always outperforms when the number of training samples is eight or more.

6.2 FUTURE WORK

While state-of-art in offline signature verification achieves around 10-15% EER in various databases, the performance of these systems would be expected to be significantly worse with signatures collected in real life scenarios. In the future, systems research needs to concentrate on increasing the robustness of systems towards larger variations encountered in real life.

Another issue is to allow the system work well with few number of references, such as three as is the case in many banking operations or even with one reference. Importance of user-based score normalization becomes significant with such extreme cases. Developing a simpler and better score normalization method is a part of our future work.

REFERENCES

- [1] A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 14, no. 1, pp. 4–20, 2004.
- [2] N. K. Ratha, J. H. Connell, and R. M. Bolle, "An analysis of minutiae matching strength," in *International Conference on Audio-and Video- Based Biometric Person Authentication*. Springer, 2001, pp. 223–228.
- [3] B. Biggio, g. fumera, P. Russu, L. Didaci, and F. Roli, "Adversarial Biometric Recognition : A review on biometric system security from the adversarial machine-learning perspective," *IEEE Signal Processing Magazine*, vol. 32, no. 5, pp. 31–41, Sep. 2015.
- [4] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *International Conference on Learning Representations*, 2014.
- [5] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and Harnessing Adversarial Examples," in *International Conference on Learning Representations*, 2015.
- [6] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, "Distillation as a defense to adversarial perturbations against deep neural networks," in *Security and Privacy, IEEE Symposium on*. IEEE, 2016, pp. 582–597.
- [7] F. Tram`er, A. Kurakin, N. Papernot, D. Boneh, and P. McDaniel, "Ensemble Adversarial Training: Attacks and Defenses," in *International Conference on Learning Representations*, 2018.
- [8] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *Security and Privacy (SP), 2017 IEEE Symposium on*. IEEE, 2017, pp. 39–57.

- [9] “Adversarial Examples Are Not Easily Detected: Bypassing Ten Detection Methods,” in Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, 2017.
- [10] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, “Towards Deep Learning Models Resistant to Adversarial Attacks,” International Conference on Learning Representations, 2018.
- [11] M. Barreno, B. Nelson, R. Sears, A. D. Joseph, and J. D. Tygar, “Can machine learning be secure?” in Proceedings of the 2006 ACM Symposium on Information, computer and communications security. ACM, 2006, pp. 16–25.
- [12] M. Barreno, B. Nelson, A. D. Joseph, and J. D. Tygar, “The security of machine learning,” *Machine Learning*, vol. 81, no. 2, pp. 121–148, 2010.
- [13] B. Biggio, G. Fumera, and F. Roli, “Security evaluation of pattern classifiers under attack,” *IEEE transactions on knowledge and data engineering*, vol. 26, no. 4, pp. 984–996, 2014.
- [14] B. Biggio and F. Roli, “Wild patterns: Ten years after the rise of adversarial machine learning,” *Pattern Recognition*, vol. 84, pp. 317– 331, Dec. 2018.
- [15] A. K. Jain, K. Nandakumar, and A. Nagar, “Biometric Template Security,” *EURASIP J. Adv. Signal Process*, vol. 2008, pp. 113:1–113:17, Jan. 2008.

APPENDICES

A. SOURCE CODE

```
clc;
clear;
close all;
warning('off','all')
%%%%%%%%%% READ THE INPUT IMAGES FROM THE DATASET
%%%%%%%%%%
[f,d]=uigetfile('*.jpg');
image=imread([d f]);
figure,imshow(image),title('INPUT IMAGE');
%%%%%%%%%% TO CONVERT THE IMAGE TO GRAYSCALE
%%%%%%%%%%
gray=rgb2gray(image);
figure,imshow(gray),title('GRAY CONVERTED IMAGE');
%%%%%%%%%% TO INCREASE THE CONTRAST %%%%%%%%%%%
thresh=graythresh(gray);
figure,imshow(thresh),title('ENHANCED IMAGE');
%%%%%%%%%% TO COVERT THE IMAGE TO BLACK AND WHITE
IMAGE %%%%%%%%%%%
binary=im2bw(gray,thresh);
figure,imshow(binary),title('BLACK AND WHITE IMAGE');
%%%%%%%%%% TAKE THE COMPLEMENT %%%%%%%%%%%
com=imcomplement(binary);
figure,imshow(com),title('COMPLEMENT IMAGE');
%%%%%%%%%% TO CROP THE IMAGE %%%%%%%%%%%
for i=1:size(com,1)
r=com(i,:);
m=max(r);
if m==1
break;
end
```

```

end
minrow=i;
for i=1:size(com,1)
r=com(size(com,1)-i,:);
m=max(r);
if m==1
break;
end
end
maxrow=size(com,1)-i;
for i=1:size(com,2)
r=com(:,i);
m=max(r);
if m==1
break;
end
end
mincolumn=i;
for i=1:size(com,2)
r=com(:,size(com,2)-i);
m=max(r);
if m==1
break;
end
end
maxcounn=size(com,2)-i;
cropped=imcrop(binary,[mincolumn minrow maxcounn-mincolumn
maxrow-minrow]);
figure,imshow(cropped),title('CROPPED IMAGE');
%%%%%%%%%EXTRACT THE LBP(LOCAL BINARY PATTERN
FEATURES)%%%%%%%%
com=imcomplement(binary);
r=1;p=8;
rlbp_trains1 = rlbp(binary,r,p);

```

```

rlbp_trains = rlbp(gray,r,p);
features1 = extractLBPFeatures(binary);
features = extractLBPFeatures(gray);
feat1=[rlbp_trains1 rlbp_trains];

%% Creation of interval-valued symbolic model
load features.mat
symbC=zeros(10,4);
for i=1:size(feat,1)
m=median(feat(i,:));
tho=std(feat(i,:));
f1k=m-2*tho;
f2k=m+2*tho;
symbC(i,:)=[f1k f2k m tho];
end
%% computing similarity values and the verification process
m1=median(feat1);
tho1=std(feat1);
fTk=m1-2*tho1;
for ii=1:2
PK=[];
if ii==1
symbC1=symbC(1:10,:);
else
symbC1=symbC(11:20,:);
end
for i=1:size(symbC1,1)
if symbC1(i,1)<fTk || fTk>symbC1(i,2)
pk=0;
PK=[PK;pk];
end
if (symbC1(i,3)-symbC1(i,4))<=fTk<=(symbC1(i,4)+symbC1(i,4))
pk=1;
PK=[PK;pk];

```

```

end
if symbC1(i,1)<=fTk<(symbC1(i,3)-symbC1(i,4))
pk=(fTk)/((symbC1(i,3)-symbC1(i,1)));
PK=[PK;pk];
end
if (symbC1(i,3)+symbC1(i,4))<fTk<symbC1(i,2)
pk=(symbC1(i,2)-fTk)/(fTk-(symbC1(i,3)+symbC1(i,4)));
PK=[PK;pk];
end
end
me=mean(PK);
S=std(PK);
theta(ii)=me+3.74*S;
end
fid=max(theta);
S=find(theta==fid);

%%%%%%%%%%CLASSIFICATION %%%%%%%%%%%
label=ones(20,1);
label(11:20,:)=2;
model=fitcecoc(feats,label);
S=predict(model,feat1);

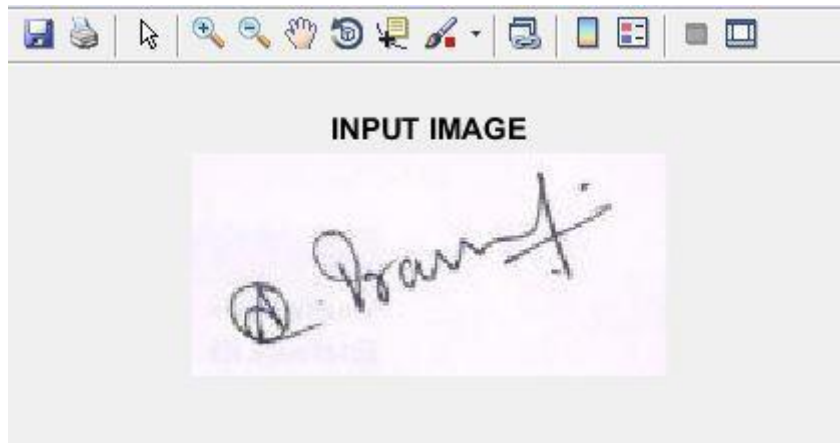
if S==2
disp('Forged Signature');
msgbox('Forged Signature');
else
disp('Orginal Signature');
msgbox('Orginal Signature');end
%%%%%%%%%%

```

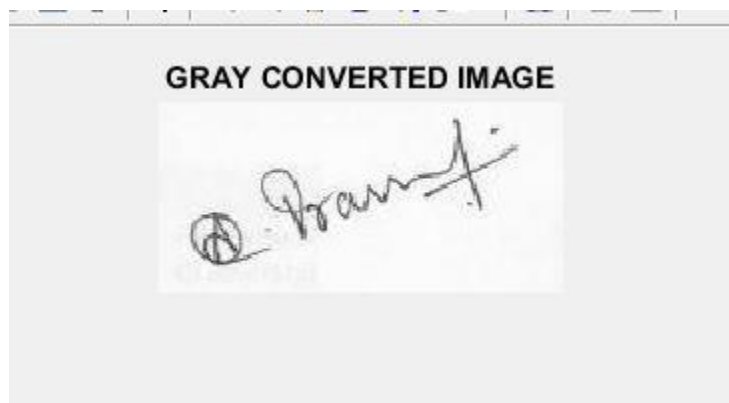

B. SCREENSHOTS

FAKE AND ORIGINAL IMAGE SIGNATURE

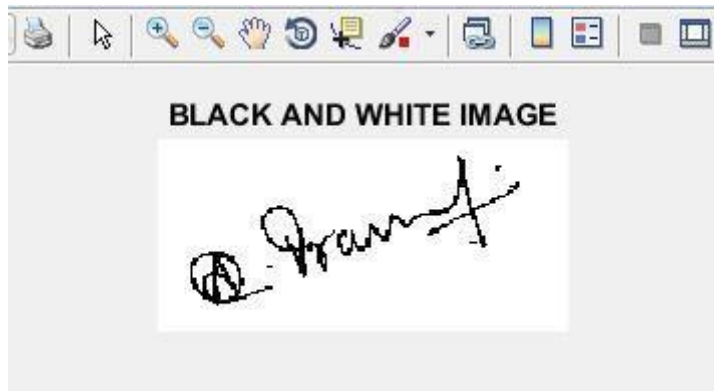
➤ *INPUT IMAGE*



➤ *GRAY CONVERTED IMAGE*



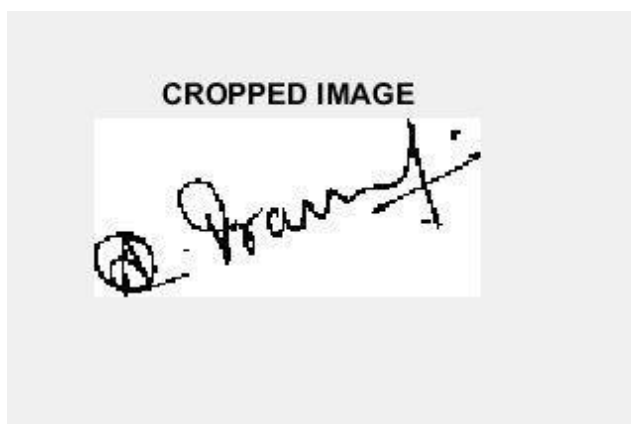
➤ **BLACK AND WHITE IMAGE**



➤ **COMPLEMENTED IMAGE**



➤ **CROPPED IMAGE**



FINAL OUTPUT

➤ FORGED SIGNATURE

```
81
82 %%%%%%%%%%EXTRACT THE LBP(LOCAL BINAR
83 com=imcomplement(binary);
84 r=1;p=8;
85 rlbp_trains1 = rlbp(binary,r,p);
86 rlbp_trains = rlbp(gray,r,p);
87 features1 = extractLBPFeatures(binary);
88 features = extractLBPFeatures(gray);
89 feat1=[rlbp_trains1 rlbp_trains];
90
91 %% Creation of interval-valued symbolic model
92 load features.mat
93
94 symbC=zeros(10,4);
95 for i=1:size(feat,1)
96     m=median(feat(i,:));
97     th=std(feat(i,:));
98     f1k=m-2*tho;
99     f2k=m+2*tho;
```

Command Window

Forged Signature

f >>

➤ ORIGINAL SIGNATURE

```
81
82 %%%%%%%%%%EXTRACT THE LBP(LOCAL BINAR
83 com=imcomplement(binary);
84 r=1;p=8;
85 rlbp_trains1 = rlbp(binary,r,p);
86 rlbp_trains = rlbp(gray,r,p);
87 features1 = extractLBPFeatures(binary);
88 features = extractLBPFeatures(gray);
89 feat1=[rlbp_trains1 rlbp_trains];
90
91 %% Creation of interval-valued symbolic model
92 load features.mat
93
94 symbC=zeros(10,4);
95 for i=1:size(feat,1)
96     m=median(feat(i,:));
97     th=std(feat(i,:));
98     f1k=m-2*tho;
99     f2k=m+2*tho;
```

Command Window

Original Signature

f >>

C. BASE PAPER AND PLAGIARISM REPORT

SIGNATURE VERIFICATION USING MATLAB WITH IMAGE PROCESSING

Miss.Suruthi¹, Miss. Thamizharasi², Mrs. Vimali³

^{1,2}U.G Scholar, Department of Information Technology

³Assistant Professor, Department of Information Technology

Sathyabama Institute of Science and Technology, Chennai, India

sujithramohan891@gmail.com, Vimalijs.balu@gmail.com, thamizhmari1999@gmail.com

Abstract— Based on a fuzzy logic and an interval symbolic representation, an effective off-line signature verification method has been developed. Similarity estimation is proposed. During the method of feature extraction, a selection of local features based on binary patterns (LBP) are derived both of the signatures picture and the as well as it under sampled bitmap Then, for each function in each signature class, interval valued symbolic data is generated. As a result, a signature model comprised of multiple signatures was developed a set of values that are separated by intervals (equivalent to the number of characteristics for each) is acquired. In this paper, we compare our findings to some recent signature verification methods that have been published in the scientific literature When it comes to the average error rate, we discovered the following average error rate the proposed strategy is still relevant It outer shapes as the number of training sessions increases.

Keywords—Local bitmap pattern, fuzzy method, under sampled, bitmap Error rate, Average.

I. INTRODUCTION

The use of biometric technology is widespread. To assess in legal and administrative activities, a person's identity is essential. They're very popular. An individual's biometric data is collected (for example, during the enrollment process) and used to train or saved as a "template" for potential comparisons. Discriminating classifier in Pattern Recognition systems. If this user has new samples, they will be listed here. The security implications of these systems' dependability are important.

Adversarial Machine Learning was used to analyze the data. [16]The point of view from this vantage point, we believe an aggressive opponent and its own objectives (for example, gaining access to a system) information (for example, knowing where a system) and experience (for example, knowing how to use a system). Where a system is) (e.g., knowing the location of a system).Data for training, or inputs during testing) and abilities (Controlling the atmosphere, for example.)[15](Classifier parameters or learning algorithm parameters) and abilities (For example, the ability to manipulate the environment).

[17]The various components of a system are defined in detail by Ratcheted and later Baggie et al. A biometric device that is vulnerable to attack.[18] None the less, a new issue of "Adversarial Examples" is on the way this raises new security issues for these systems.

This problem is about adversarial input.[19] Perturbations that are explicitly designed to cause Errors in classification Szegedetal.[20]Demonstrated that very small (almost imperceptible) image perturbations could be used to deceive a human.

II. LITERATURE SURVEY

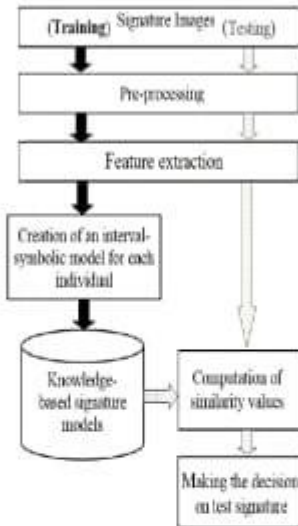
[1]In this article, we will look at they looked into the establishment of a fixed-length Scheme of the Fuzzy Vault (FV) Application-specific dynamic models Verification of signature (DSV) is a method of ensuring that only one person has signed a document. [2]The signature has 15 global characteristics. Considered in the development of templates it is a theory based on the premise that any perceived object is saved as a "template" in the brain.[4]Memory that lasts a long time Knowledge that comes in is compared to these models in order to come up with a perfect match. In this paper, we'll look at; they looked into biometric structures and the effects of adversarial examples specifically using Identifying Possible Risks to Offline Handwritten Documents. [3]At the point of sale, signature authentication is performed.

[5]Machine Learning with Adversarial Behavior from the Adversarial Machine Learning's Point of View, Their research demonstrates that in some cases, the new examples of adversarial examples are presented in this issue. Challenges to such structures.[6]Systems that are both handcrafted and automated function extractors and learning systems directly from the pixels of an image.

Offline Handwritten Signature Authentication is exposed to complex attack situations, which we define. [7]The assailant's objectives necessary information and competences Signature verification is a form of biometric verification in the field of automated verification this is a hot subject verification.[10]Legal, banking, and high-level verification environments that are stable authentication.[9]There are two forms of verification internal and external. Both on and off the internet. A web-based technique is used. Electronic tablet and stylus. CNN-based classifier that is cutting edge.[8]Attacks designed for one model often spread to others. Other models suggest that a cyber- assailant may build a surrogate classifier of its own and launch As long as it has access to a computer, it can launch attacks.The target's data. However, the theoretical causes remain unknown, and most defenses are ineffective that is, they fail if the attacker is successful. [11]The intruder is aware of the defense.

[14] This paper is structured in such a way that we begin by formalising the problem at hand and listing the most important datasets for evaluating such systems. [12]The techniques used for each operation of the pipeline for training a system are then defined. [13]Preprocessing, Feature Extraction, and Model Training are discussed, followed by a summary of recent progress and possible research areas.

SYSTEM ARCHITECTURE



III. METHODOLOGY

The signature verification task has a lot of intra-class variability. As compared to physical biometric characteristics like fingerprints or iris, handwritten signatures from the same individual often display a lot of variance between samples. Consider if expert forgeries are a possibility. These forgeries are targeted at a particular audience. In this case, the user's signature is often imitated. As a consequence, expert forgeries are commonplace. They mimic genuine signatures to a large degree.

Another important obstacle for training an automated signature verification system is the existence of partial information during training. We only have access to valid signatures for users who have registered in the system during training in practise. During operations, however, we would like the framework to be able to accept genuine signatures while refusing forgeries. This is a challenging job because for users who have signed up for the scheme, there is no information available to train a classifier to learn what exactly distinguishes a genuine signature from a forgery.

A. Input Image

This function reads and displays a picture to be used as an input using the camera or the read command, load an image into the workspace. It is used in the field of image processing Image is classified as the activity of obtaining information from an originator. Image obtained from a third-party source, typically as a processing source based on hardware It's real the very first step in the workflow and there is no processing without an image it's likely. [The image that is obtained has not been processed in any way.



B. Data Preprocessing

Pre-processing is a term that refers to operations on pictures at the most basic the degree of abstraction, with input as well as performance being images. Images of high strength the aim of pre-processing is to improve the quality of the final product. To improve data from a picture by suppressing or eliminating unwanted distortions and enhancing some key picture features.

Further sorting the major redundancy of images is exploited by image pre- processing methods. Neighboring pixels have same color as each other. In real- life pictures, one object essentially a brightness value that is the same or ider Consequently, Sometimes. A picture can be reconstructed fit blurred pixel. The average value of pixels in close proximity.

TABLE I
COMMONLY USED SIGNATURE DATASETS

Dataset Name	Users	Genuine signatures	Forgeries
CEDAR [36]	55	24	24
MCYT-75 [20]	75	15	15
GPDS Signature 160 [17]	160	24	30
GPDS Signature 960 Grayscale [82]	861	24	30
GPDS Synthetic Signature [19]	4000	24	30
Brazilian (PUC-PR) [21]	60 + 108	40	10 simple, 10 skilled

C. Resizing the Image

All of the images in the feedback are resized to the same proportions. The output image will be distorted if the input aspect ratio does not match the required dimension.

D. Converting the color Format

Color information is useless in many image processing applications. If you're in the business of trying to fix problems, you're in for a long haul. If you can tell one color from another, you can move on to the next stage. Converting an RGB image to a CMYK image is one of the reasons for doing so, GRAYSCALE or BLACK AND WHITE in image formats.

GRAY CONVERTED IMAGE



BLACK AND WHITE IMAGE



E. Feature Extraction

Feature extraction begins with an initial collection of calculated data in pattern recognition, and machine learning creates derived values (features) that are supposed to be non-redundant and informative facilitating subsequent learning and development measures of generalization, and, in some circumstances, Dimensionality reduction is linked to feature extraction.

When an algorithm's input data is too big to handle process and is considered to be obsolete (for example, a similar calculation in two places) both of the feet and meters, or the monotony of seeing all pixels as photographs) so it's possible. Transformed into a smaller set of characteristics. A function vector is another name for it. Feature selection is the method of deciding a subset of the initial features. The features that have been chosen are planned to include all relevant details so that the desired mission can be performed from the input data can be carried out with the aid of this reduced as compared to the full initial details.

A collection both the signature image and the collection of features based on the Local Binary Pattern (LBP) are computed the background image in the feature extraction stage. As well as after that, interval valued symbolic data is generated for each signature class under a sampled bitmap has its own collection of features. As a result, each function is given a signature model, which is made up of a set of values that are separated by intervals (corresponding to the number of features).

Handwritten signatures are a type of signature that is written by a person. A new fuzzy similarity test is also implemented. Proposed a method for measuring the similarity of a sample signature and a real signature for testing and the test's corresponding symbolic interval-valued interval-model verifications an example.

IV. Classification

The process a method for extracting information groups from a raster image with multiple bands is known as image classification. The raster that results is as follows: It is possible to use image classification to create a thematic map the proposed a method of classification and Image analysis is used for multivariate analysis. Toolbar for classification. There are various classification algorithms available, as well as some that are provided as follows

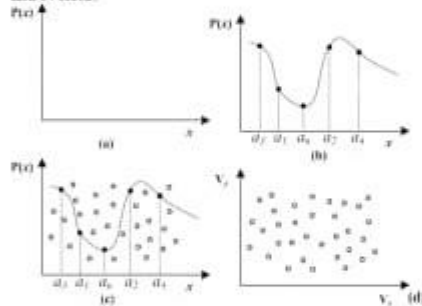
• SVM (SUPPORT VECTOR MACHINE)

It is an acronym for the approach proposed given Compared to the previous year, the results were slightly higher. The most up-to-date approaches when it comes to Offline signature datasets from two separate sources. The proposed model's main benefit is that it facilitates the development and incorporation of a model for a new person using just a few lines of code. Genuine signatures that are similar parameters as before, without the need for any changes many of the parameters are being re-tuned.

• Fuzzy Similarity Measures

This proposal is provided as an example. A cryptographic construction that is error-tolerant which is unbreakable and uses unsorted collection (e.g. a list of) biometric characteristics to encrypt/decrypt a hidden key for the purpose of accessing the vault.

This Not only does this technique keep the key safe, but it also keeps the key safe. Secures it against the haphazardly organized collection. The school district of FV to protect K is a user-specific key with a length of M bits. Produced from the biometric feature set (a1, a2, an). Created at random. R) is a code that corrects errors? On K and N, the Solomon (RS) approach is used. Concatenating the given redundancy to K yields a central K of length N that has been encoded. Fig 3 Redundancy level of the K and N field.



A few (NM) is a unit of measurement for the number of bits in a So there's a degree polynomial P. The letter L (being L.n) is built. When K is used as a set of coefficients, for each element of A, the polynomial projections a1, ..., a2 are calculated. Yielding a genuine collection of polynomial projections, p-G .fig 3 (a) points To keep genuine arguments secret, points of chaff that don't cross paths with both P denotes a polynomial equations.

A is a group of things, and B is a group of things. Randomly produced finally, there is a vault collection V I. G is made up of collection and chaffing combined a few points. (b)Biometric B is the probe prototype provided during authentication if the template B overlaps the template A, the vault must be decoded. Recover K. If A can define B in a meaningful way, then B can as well.

V has a lot of genuine points to make. There isn't much of There is enough of a gap between sets A and B that the redundancy that exists in (k). Enables the erroneously specified points must be corrected, P is a polynomial must be effectively a reconstruction, and thus K, the key that goes with it, must be found. Reconstructed the polynomial successfully, at the very least, L.l genuine points are needed. V has been established.

V. Experimental Result

As a result, for each individual's handwritten signature class, a signature model composed of a set of interval values is obtained. In terms of average error rate, we compared our results to some recent signature verification methods published in the literature. When the number of training samples is eight or more, the proposed method often outperforms. On the other hand, the global SVM improves the performance when used in conjunction with user SVMs. Classifier combination is applied at score level to combine the decisions of the six classifiers.



VI. CONCLUSION

The effect of using an interval symbolic r representation for off-line signature verification was explored in this paper. Also proposed is a fuzzy similarity metric. A collection of Local features is extracted in the feature extraction stage. Features based on Binary Patterns (LBP) are derived from both the signature picture and the as well as it's under sampled bitmap. Then, for each function in each signature class, interval valued symbolic data is generated. When it comes to signatures, to accomplish this, we plan to use to merge the originals, Use the XOR logic feature. Password-protected design (independent of until the vault is created (the user key). Finally, we'll use an encryption method to protect the vault we've built. In this manner, Renewability and unlink ability can be accomplished. Although the protection is being strengthened, this target is being reached. As a consequence, the initial power is reduced. On the system's output transformation should be examined.

VII. Reference

- [1] Jain, A. K., Ross, A., & Prabhakar, S. (2004). An introduction to biometric recognition. *IEEE Transactions on circuits and systems for video technology*, 14(1), 4-20.
- [2] Ratha, N. K., Connell, J. H., & Bolle, R. M. (2001, June). An analysis of minutia-matching strength. In *International Conference on Audio and Video-Based Biometric Person Authentication* (pp. 221-228). Springer, Berlin, Heidelberg.
- [3] Biggio, B., Fumera, G., & Roli, F. Adversarial biometric recognition: A review on biometric system security from the adversarial machine-learning perspective. *IEEE Signal Processing Magazine*, 2015 Aug 12;32(5):31-41.

- [4] Papernot, N., McDaniel, P., Wu, X., Ju, S., & Swami, A. (2016, May). Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE Symposium on Security and Privacy (SP)* (pp. 582-597). IEEE.
- [5] Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D., McDaniel, P., Ensemble adversarial training. *Attacks and defenses arXiv preprint arXiv:1705.07304*, 2017 May 19.
- [6] Carlini, Nicholas, and David Wagner. "Towards evaluating the robustness of neural networks." *2017 IEEE Symposium on Security and Privacy (S&P)*. IEEE, 2017.
- [7] Biggio, B., Fumera, G., & Roli, F. (2013). Security evaluation of pattern classifiers under attack. *IEEE transactions on knowledge and data engineering*, 25(4), 984-996.
- [8] Jain, A.K., Nandakumar, K. and Nagar, A., 2008. Biometric template security. *EURASIP Journal on advances in signal processing*, 2008, pp.1-17.
- [9] Sivasubari, A., P. Ajitha, Immanuel Rajkumar, and S. Poornadhari. "Emotion recognition system for autism disorder people." *Journal of Ambient Intelligence and Humanized Computing* (2019): 1-7.
- [10] Ajitha, P., and G. Ganasekaran. "An approach of opinion mining for online marketing using sentiment thesaurus and concept search engine." In *2015 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCCCT)*, pp. 208-213. IEEE, 2015.
- [11] Sivasubari, A., P. Ajitha, and R. M. Gomathi. "Light weight security scheme in wireless body area sensor network using logistic chaotic scheme." *International Journal of Networking and Virtual Organisations* 22, no. 4 (2020): 433-444.
- [12] Ray, Soumya, and A. Sivasubari. "Design of Mobile Robot Teleoperation System using Virtual Reality." In *2020 International Conference on Communication and Signal Processing (ICCCSP)*, pp. 1173-1175. IEEE, 2020.
- [13] Jitila, Y. B., Rajalakshmi, V., Gladence, L. M., & Ann, V. M. (2020). Food Consumption Monitoring and Tracking in Household Using Smart Container. In *Proceedings of the Third International Conference on Computational Intelligence and Informatics* (pp. 693-700). Springer, Singapore.
- [14] Jitila, Y. B., Alan, M. S., & Singh, P. D. (2019). Cloud-Based Scheme for Household Garbage Collection in Urban Areas. In *Advances in Big Data and Cloud Computing* (pp. 539-546). Springer, Singapore.
- [15] Vimala, J. S., Sankara Srinivasulu, J. Jabez, and S. Gowri. "Hand Gesture Recognition Control for Computers Using Arduino." In *Data Intelligence and Cognitive Informatics*, pp. 569-578. Springer, Singapore, 2021.
- [16] Vimala, J. S., S. Srinivasulu, and S. Gowri. "IoT based bank security system." *Int J Recent Technol Eng* (2019): 2324-2327.
- [17] Gowri, S. and Divya, G., 2015, February. Automation of garden tools monitored using mobile application. In *International Conference on Innovation Information in Computing Technologies* (pp. 1-6). IEEE.
- [18] Akshaya, R., N. Niroshma Raj, and S. Gowri. "Smart Mirror-Digital Magazine for University Implemented Using Raspberry Pi." In *2018 International Conference on Emerging Trends and Innovations in Engineering And Technological Research (ICETIETR)*, pp. 1-4. IEEE, 2018.
- [19] J. Jose and T. Sasipraba, "Indoor air quality monitors using BJT sensors and LPWAN," *2019 3rd International Conference on Trends in Electronics and Informatics (ICEI)*, Tirunelveli, India, 2019, pp. 635-637.
- [20] "An Efficient System to Predict and Analyze Stock Data using Hybrid Techniques" Jithina Jose, SujachandrakulapamthiMata, B.KeerthiSarditha *International Journal of Recent Technology and Engineering (IJRTTE)* ISSN: 2277-3878, Volume-8 Issue-2, July 2019.

PALGIARISM REPORT

thamizhmari1999@gmail.com.docx

ORIGINALITY REPORT

14%	11%	12%	10%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.check-plagiarism.com Internet Source	2%
2	www.groundai.com Internet Source	1%
3	hellanicus.lib.aegean.gr Internet Source	1%
4	Submitted to University College London Student Paper	1%
5	theses.eurasip.org Internet Source	1%
6	Submitted to University of Technology, Sydney Student Paper	1%
7	Submitted to European University of Lefke Student Paper	1%
8	Submitted to Higher Education Commission Pakistan Student Paper	1%
9	Yahaya Isah Shehu, Anne James, Vasile	

Palade. "Detecting an Alteration in Biometric Fingerprint Databases", Proceedings of the 2nd International Conference on Digital Signal Processing - ICDSP 2018, 2018
Publication **1%** |


10 Submitted to University of Birmingham
Student Paper **1%** |

11 www.springerprofessional.de
Internet Source **1%** |

12 Alireza Alaei, Srikanta Pal, Umapada Pal, Michael Blumenstein. "An Efficient Signature Verification Method Based on an Interval Symbolic Representation and a Fuzzy Similarity Measure", IEEE Transactions on Information Forensics and Security, 2017
Publication **1%** |

45

ACCEPTANCE LETTER



ICICCS

5th International Conference on
Intelligent Computing and Control Systems
ICICCS 2021 | May 06 - 08, 2021
iciccs.com/2021/ | iccs.conf19@gmail.com

Letter of Acceptance

TO
Miss.Suruthi, Miss. Thamizharasi, Mrs. Vimali, Assistant Professor,
Department of Information Technology
Sathyabama Institute of Science and Technology, Chennai, India.

Paper ID- ICICCS192

Subject: Acceptance to the International Conference on Intelligent Computing and Control Systems [ICICCS 2021], 06-08, May 2021, Madurai, India.


Dear Author,

We are happy to inform you that your paper titled "**SIGNATURE VERIFICATION USING MATLAB WITH IMAGE PROCESSING**" has been accepted for the oral presentation at the International Conference on Intelligent Computing and Control Systems [ICICCS 2021] to be held on **6 – 8, May 2021** at Vaigai College of Engineering, Madurai, Tamil Nadu, India.

With the evident of its previous publications, ICICCS 2021 is also dedicated for the publication in **IEEE Xplore Digital library**. As a result of the peer review process, the technical conference program committee is pleased to inform you that your paper is shortlisted and accepted for the presentation in conference event and formally accepted for inclusion in IEEE Xplore. We appreciate if you could submit the final version of manuscript at your earliest convenience, in order to ensure a novel and timely publication of your research paper.

Once again, on behalf of the conference committee we extend our warm gratitude to welcome you at our conference site.

Yours' Sincerely,



Prof. P. Sugumaran
Vice Principal,
Vaigai College of Engineering,
Madurai, India.

Our Previous Publications
[ICICCS 2020 Publication Link](#)
[ICICCS 2019 Publication Link](#)
[ICICCS 2018 Publication Link](#)
[ICICCS 2017 Publication Link](#)

Proceedings by

