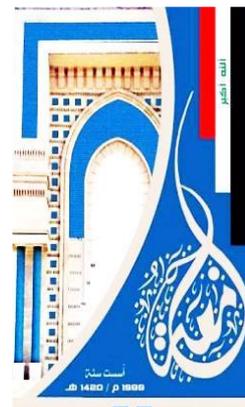


Ministry of Higher Education and
Scientific Research
University of Diyala
College of Science
Department of Computer Science



Deep Semantic Segmentation Based Positive-Unlabeled Convolutional Neural Network for Low Quality Finger Vein Image Pattern Extraction

A Thesis Submitted to the Department of Computer Science \ College of
Sciences \ University of Diyala in a Partial Fulfillment of the Requirements
for the Degree of Master in Computer Science.

By

Ali Salah Hameed

Supervised By

Dr. Adel Abdulwahhab Al-azzawi
Assistant Professor

2022 A.C

1444 A.H

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

وَقُلْ رَبِّ زِدْنِي عِلْمًا

صِدْقَ اللَّهِ الْعَظِيمِ

سورة طه
رقم الآية : ١١٤

ACKNOWLEDGMENT

As I conclude this work, I can only express my sincere thanks and gratitude to my supervisor, who made his knowledge a reference for me, and from his advice the right path for my steps. **Assistant Professor Dr. Adel Abdulwahhab Al-azzawi**, who never failed to help me during the research period.

I extend my thanks, respect, and love to my father and mother, who were examples to me in life, and may God prolong their life and grant them health and wellness.

I also extend my thanks and love to my dear wife, who stood with me throughout my study period.

It is the duty of gratitude to honor that I extend my thanks and appreciation to all members of the College of Science, especially my professors in the Department of Computer Science. Special thanks to all my friends for giving me advice.

Ali Salah

Dedication

To my family with my love

Abstract

Today, finger vein identification is gaining popularity as a potential biometric identification framework solution. Most datasets include a lot of noise and undesirable objects in the background, making it difficult to identify and extract finger vein. Image thresholding it seems to be a suitable approach for the binarization stage that is utilized into some exact finger vein image detection and extraction. In this thesis, two models are proposed to extract the pattern of finger veins, an unsupervised model and a supervised model. The first unsupervised model includes a methodology based on two developmental stages.

The first stage is using a new development of the (K-mean) algorithm, which depends on the difference between the intensity of pixels for points. The centrality determined according to the process of normalizing the image pixels and making them include five levels of intensity, where similar pixels are grouped into one group called a cluster, as the purpose of this process is to isolate the finger area from the background that contains unimportant details.

The second development phase includes dividing the image into blocks and determining the paths of the finger veins pattern using the (GPE and GPO) techniques to obtain an accurate and clear shape of the vessel pattern, as the purpose of the process is to create the vein patterns for each image in the database for the purpose of training it in the second model. The second supervised model is where the predicted masks are evaluated and selected based on the similarity of the predicted binary pattern masks. Ultimately, human identification and validation are made based on predicted pure binary pattern masks using new framework in CNN called U-NET.

The first unsupervised model achieves an accuracy of 98.6% which is substantially higher than other standard unsupervised learning approaches like Fuzzy C-Means (FCM) and k-means algorithms.

The outcomes the second supervised model demonstrate that this model is able to generate a very significant clean finger vein pattern and is able to successfully identify the images of finger veins from the same individual by achieving an average accuracy of 95% of the identification similarity scores between two fingers for the same person.

Contents

Subject	Page
Abstract	I
Contents	III
List of Abbreviations	VI
List of Figures	VI
List of Tables	IX
List of Algorithms	X

Chapter One (General Introduction)		
1.1	Motivation	1
1.2	Biometric Data	2
1.3	Biometric Identification Systems.	5
1.4	Concepts of the Biometric Identification Systems	7
1.5	Advantages and Challenges	8
1.6	Public database.	10
1.7	The Aim of the Study.	11
1.8	Literature Review.	12
1.9	Outline of the Thesis.	14

Chapter Two (Background Theory)		
2.1	Fundamentals of Biometric identification system	16
	2.1.1 Fingerprint-based Biometric Authentication	16
	2.1.2 Finger vein-based Biometric Authentication	17
2.2	Digital Images Types	18
	2.2.1 Binary Image	18
	2.2.2 Gray Scale Image	19
	2.2.3 Color Image	19

2.3	LAB Color Model	20
2.4	Image Processing in Digital Form Operations	21
	2.4.1 Image Enhancement	21
	2.4.2 Image normalization	24
	2.4.3 Image Adjustment	24
	2.4.4 Morphological Operation	25
2.5	Machine learning	25
	2.5.1 Background Clustering Algorithm Theory	26
2.6	Deep learning	28
2.7	Convolution neural network (CNN)	28
2.8	Deep Semantic Segmentation	31
	2.8.1 Semantic Segmentation based Deep Learning Method	31
2.9	Positive Unlabeled Learning Scheme	32
	2.9.1 Auto encoder-based classifier regularization.	33
2.10	Digital Image Quality Metrics	33
	2.10.1 Subjective Quality Assessment	34
	2.10.2 Objective Quality Assessment	35

Chapter Three (Proposed System)		
3.1	Introduction	39
3.2	Research Methodology.	39
3.3	Implementation Scheme	40
3.4	Fully automated finger vein pattern extraction system	41
	3.4.1 First model: Fully Automated Finger Vein Binary Pattern Extraction based Unsupervised Learning Approach	41
	3.4.2 Second Model: Fully Automated Finger Vein Pattern Extraction based Deep Semantic Segmentation.	54

Chapter Four (Experimental Results)		
4.1	Introduction	63
4.2	Implementation Environment	64
4.3	Dataset	64
4.4	Experiential Results of the Pre-processing Stage	65
	4.4.1 Subjective Experimental Results of the Pre-processing-Stage	65
	4.4.2 Quantitative Experimental Results of the Pre-processing Stage Results	67
4.5	Experiential Results of Finger Vein Localization Stage:	70
	4.5.1 Subjective Experimental Results of Image Clustering	70
	4.5.2 Quantitative Experimental Results of Image Clustering.	74
	4.5.3 Experiential Results of Finger Pattern Extraction Stage:	77
	4.5.4 Subjective Miura_Repeated_Line_Tracking Result	77
	4.5.5 Subjective GPE and GPO Result	78
4.6	Experiential Results of the Supervised Learning Model	81
	4.6.1 Training and Testing Scheme.	81
	4.6.2 Fully Automated Binary Mask Prediction Results	82
	4.6.3 Experiential Results of the Finger Vein Identification	90

Chapter Five (conclusions, future works)		
5.1	Conclusions	93
5.2	Future Work	95

References	97
-------------------	-----------

List of Abbreviations

Abbreviation	Meaning
CNN	Convolutional Neural Network
DL	Deep Learning
GPE	Global Pattern Extraction
GPO	Global Pattern Optimization
IOU	Intersections Over Unions
IQA	Image Quality Assessment
ML	Machine Learning
MSE	Mean Square Error
PSNR	Peak-Signal to Noise Ratio
PU	Positive Unlabeled
ROI	Regions Of Interest
SNR	Signal-To-Noise Ratio

List of Figures

1.1	Examples of biometric features information that may be used to verify (identify/verify) an individual's identity are provided.	2
1.2	Enrollment, verification, and identification task block diagrams.	8
2.1	Fingerprint-based Biometric Authentication	17
2.2	Finger vein - based Biometric Authentication	18
2.3	Some examples of binary images	18
2.4	Some samples of gray scale images	19
2.5	Some examples of color images	20
2.6	Learning Styles Influence Machine Learning (a): supervised Algorithms, (b): Unsupervised Algorithms	26
2.7	Convolutional Neural Network model	30
2.8	Semantic Segmentation processes	32

3.1	Research Methodology of Our Proposed System Framework	40
3.2	First Model System Flowchart (Fully automated Finger Vein Pattern Extraction Based Double Optimization Stages).	42
3.3	Stage 1 Preprocessing Stage Flowchart.	45
3.4	Stage 2 Image Clustering Flowchart.)	48
3.5	Stage 2:Post-processing System Flowchart.	52
3.6	All States from the First Model	54
3.7	Fully Automated Finger Vein Pattern Extraction based Deep Semantic Segmentation Flowchart	56
3.8	Deep Semantic Segmentation Model based U-net Structure using positive and unlabeled loss function.	57
4.1	Samples of the Finger Vein Dataset	64
4.2	Experimental Result for Pre-processing Stage.	65
4.3	Mean square error (MSE) of the original finger vein images and for enhanced images.	67
4.4	Peak-Signal-to-Noise-Ratio (PSNR) of the Original and Enhanced images	68
4.5	. Signal-to-Noise-Ratio (SNR) of the Original and Enhanced image.	68
4.6	Different Binary Masks from Algorithms Used.	70
4.7	Different Binary Mask images (a)-(d) are original images, (e)-(h) are FCM Masks, (i)-(l) are standard K-mean Masks, (m)-(p) are Out's Masks, (q)-(t) are new K-mean Masks.	71
4.8	Experimental Result for Post-processing Step.	72
4.9	Quantitative Experimental Results of the Image Localization Stage Based on Using Different Unsupervised Clustering Algorithms.	75

4.10	Different result for First and Second Stages in first Model. (a)Original image, (b) Enhanced image, (c) New K-mean Binary Mask, (d) Localized Finger Vein Image	75
4.11	Different result For The First Step (Miura_Repeated_Line_Tracking Result) from Last Stage.	76
4.12	Finger Vein Direction Map.	77
4.13	Line connection using GPO technique.	78
4.14	Finger Vein Line Extraction Results based on GPE and GPO.	79
4.15	Final Result for all operation in the first model.	79
4.16	Training and Validation Performance Results Based on Using 5 Epochs only (a) Represents the Intersection over Union Curve (IoU), (b) Loss Function Curve.	82
4.17	Training and Validation Performance Results Based on Using 10 Epochs only (a) Represents the Intersection over Union Curve (IoU), (b) Loss Function Curve.	83
4.18	Training and Validation Performance Results Based on Using 15 Epochs only (a) Represents the Intersection over Union Curve (IoU), (b) Loss Function Curve.	85
4.19	Training and Validation Performance Results Based on Using 20 Epochs only (a) Represents the Intersection over Union Curve (IoU), (b) Loss Function Curve.	86
4.20	Training and Validation Performance Results Based on Using 25 Epochs only (a) Represents the Intersection over Union Curve (IoU), (b) Loss Function Curve.	88
4.21	Fully Automated Finger Vein Pattern Extraction based Deep Semantic Segmentation using PU Loss Function Result (a, d, g) Localized Finger Vein Image (b, e, h) Finger Vein Pattern (Ground Truth) (c, f ,i) Predicted Pattern mask. and (j),(m) .	90

4.22	Fully Automated Finger Vein Human Identification Results using the Predicted Masks That Been Generated from the Deep Semantic Segmentation (a),(e) and (i),(l) are the Localized Finger Vein Image For The Same Person , (b),(f) and (j),(m) are The Predicted Mask For The Same Persons and (c), (g), (k),(o) are Localized Images For the Different Persons and (d),(h),(l),(n) are The Predicted Mask For The Different Persons.	92
-------------	---	----

List of Tables

1.1	Comparison of biometric technology. The letter H means high, M means medium, and L means low	6
1.2	Comparisons of the Major Biometrics Techniques	7
1.3	The existing public finger-vein databases	11
3.1	Encoder Operation	61
3.2	Decoder Operation	62
4.1	Objective Quality Assessment of Processing Stage.	67
4.2	Finger Vein Image Clustering Results using 10 Samples randomly selected for different clustering algorithm.	74
4.3	Training and Testing Data Partitions of Our Proposed System.	81
4.4	Training Performance Results based 5 Epochs Only.	81
4.5	Training Performance Results based 10 Epochs Only.	83
4.6	Training Performance Results based 15 Epochs Only.	84
4.7	Training Performance Results based 20 Epochs Only.	86
4.8	Training Performance Results using 25 Epochs Only	87
4.9	Finger vein Testing Performance Result.	89
4.10	Finger Vein Identification/Verification Based Similarity Performance for the same person.	93

List of Algorithms

3.1	Preprocessing Stage Algorithm	45
3.2	Image Adjustment Algorithm	47
3.3	Clustering Step Algorithm	49
3.4	Post Processing Stage Algorithm	51
3.5	Global Map Extraction Algorithm	54
3.6	Semantic Segmentation Algorithm	58

Chapter One

General Introduction

1.1 Motivation

Biometric technology has gotten a lot of attention from the public in recent years in which a remarkable increasing need for biometric system safety and accuracy was observed. Fingerprints, palm prints, iris, gait, facial characteristics, voice recognition, signature, heartbeat, and palm vein have all been suggested as biometric identification methods. However, the majority of these biometric identification methods have a number of flaws, including susceptibility, lighting, facial expression, position, and occlusion. Instead, a more reliable, safe, and convenient biometric technique, the finger vein, has lately been deployed [1].

When it comes to finger vein recognition, convolutional neural networks CNNs have become increasingly popular in the field of image recognition as deep learning technology has advanced and the pixel-level label information obtained by traditional texture extraction techniques has been used to train CNNs. A template matching strategy is used to classify the images of finger veins based on the feature data. Finger vein patterns were extracted using CNN and then restored using Fully Convolutional Network FCN [2].

However, designing a completely automated finger-vein detection system in the absence of labeled data continues to be a difficult undertaking to do. Aside from that, in the most recent suggested systems, the entire data set (the entire finger vein image) has been utilized in order to train the model, which results in some biasing in terms of the data and environmental similarities, such as skin color, backdrop, and other aspects. On the other hand, the finger vein lines should be regarded as the primary pure data because they are the primary pattern upon which it is built. This

work proposes a completely automated, unsupervised learning strategy for the automatic production of training finger vein line data. Our technique eliminates the challenges associated with manually labeling training datasets by employing a fully automated way to build them based on certain complex algorithms and procedures [3].

1.2 Biometric Data

Biometric is a methodological study of measuring and analyzing biological data for the purpose of authentication or identification. Biometrics refers to certain physiological or behavioral characteristic that is uniquely associated to a person. In fact, biometric technology is ancient Egyptian times technology. The Biometrics can be defined as the study of measuring and analyzing the unique physical or behavioral traits, which is used for the purpose of recognizing a person [4]. Figure (1.1) shows typical of existing Biometric features.

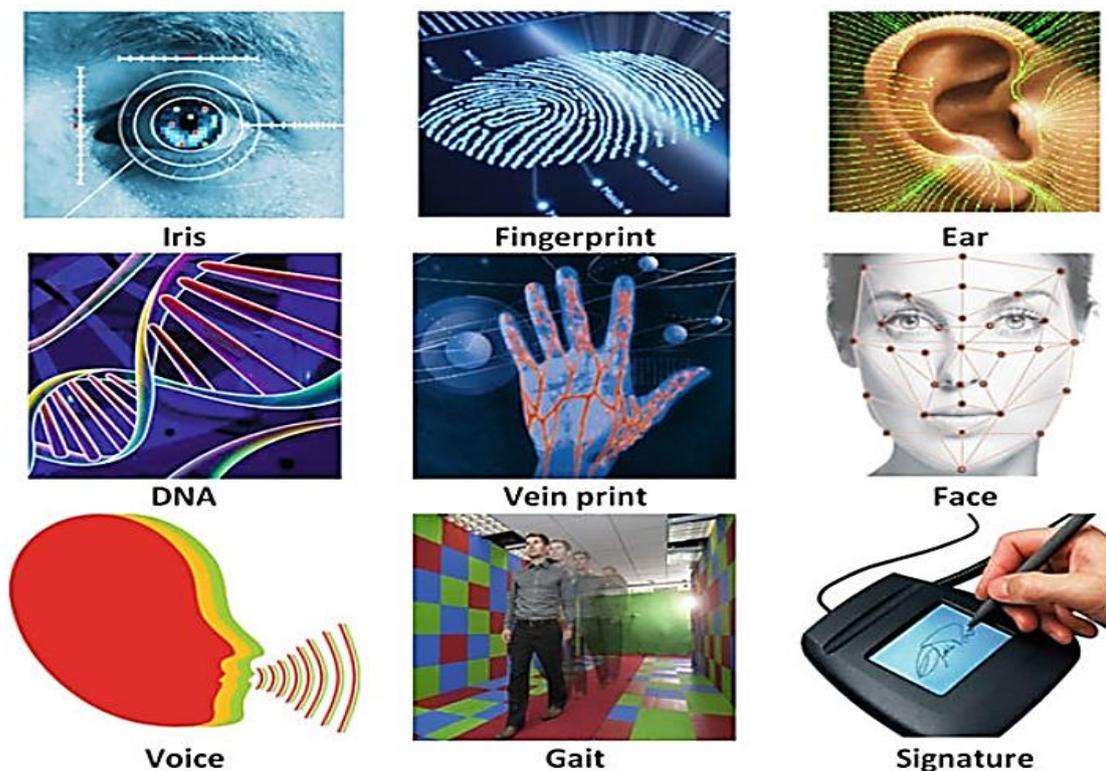


Figure (1.1): Examples of Biometric Features Information that may be used to verify (Identify/Verify) an Individual's Identity are Provided [5].

The characteristics of high acceptance are widely used in different fields. Finger vein recognition is feature-rich, easy to collect and user. The characteristics of high acceptance are widely used in different fields. The general process of finger vein recognition technology includes finger vein image acquisition, finger vein image preprocessing, finger vein feature extraction and recognition, etc [6].

Like fingerprints or iris patterns, finger vein based blood vessel patterns are unique for each individual. Finger vein based blood vessel pattern have high security because the veins are located under the surface of the skin. The fingerprints can be cheated by dummy finger fitted with a copied fingerprint, but the finger vein-based identification system is highly secure for authentication [7].

There are different types of biometric identification system:

1. Fingerprint

Fingerprint is one of the most mature biometric traits and considered legitimate proof of evidence in courts of law all over worldwide. Fingerprints are, therefore, used in forensic divisions worldwide for criminal investigations. More recently, an increasing number of civilian and commercial applications are either using or actively considering using fingerprint-based identification because of a better understanding of fingerprints as well as demonstrated matching performance than any other existing biometric technology [8].

2. Hand Geometry

All biometric techniques differ according to security level, user acceptance, cost, performance, etc. One of the physiological characteristics for recognition is hand geometry, which is based on the fact that each human hand is unique. Finger length, width, thickness, curvatures and relative location of these features distinguish every human being from any other person .Hand geometry is considered to achieve medium security, but

with several advantages compared to other techniques: medium cost as it only needs a platform and medium resolution reader or camera, it uses low-computational cost algorithm which leads to fast results, low template size (from 352 to 1209 bytes), which reduces the storage needs, very easy and attractive to users leading to great user acceptance [9].

3. Iris Recognition

The purpose of ‘Iris Recognition’, a biometrical based technology for personal identification and verification, is to recognize a person from his/her iris prints. In fact, iris patterns are characterized by high level of stability and distinctiveness. Each individual has a unique iris the difference even exists between identical twins and between the left and right eye of the same person [10].

4. Retina Recognition

Retina Recognitions captures and analyzes the patterns of blood vessels on the thin nerve on the back of the eyeball that processes light entering through the pupil. Retinal patterns are highly distinctive traits. Every eye has its own totally unique pattern of blood vessels. The eyes of identical twins are also distinct. The strength of the retina recognition is difficulty in destroying its features [11].

5. Signature Recognition

Signatures biometrics can be classified in to online and offline depending upon the approach. Online signatures verification is more unique and difficult to forge than offline signature verification. Offline signature verification uses shape information while its counterpart uses dynamic features like speed, pressure and capture time of each point on the signature trajectory. In short, online signatures have an extra dimension, which is not available for the offline signatures which make online signature verification more reliable than the other one [12].

6. Voice Recognition

The recognition of speakers is typically divided into two categories: speaker identification and speaker verification or authentication. Speaker identification is the process of determining an unknown speaker's identity by matching his or her voice to the voices in the database of registered speakers. Speaker verification can determine whether a person is what he or she claims to be based on his or her voice sample [13].

1.3 Biometric Identification Systems

The general biometric system constitutes of four modules viz., Sensor, Feature Extraction, and Feature Matching. The efficiency of a biometric system depends on the reliability of the sensor used and the features extracted from the sensed signal. The learning phase and the recognition phase are the two phases of biometric system [4].

1. **Sensor module/ Image acquisition:** to capture the individual's raw biometric data in the form of video, audio and an image or some other signal.
2. **Feature extraction Module:** automated process of extracting distinctive biometric features to generate Template and can be done using Machine learning, Computer Vision and Pattern Recognition techniques.
3. **Database module:** a repository of registered /enrolled biometric information of users and stores various templates of user.
4. **Matching module:** compares the currently extracted features against the stored templates to generate match value or match score, which is computed one to find the similarity between two biometric samples.
5. **Decision-making module:** gives the decision as accepted or rejected by comparing the matching scores with given threshold value.

Table (1.1) shows that the finger vein recognition has a very good balance of all the desirable properties. Every human being possesses finger veins with the exception of any hand-related disabilities; finger vein details are permanent, even if they may temporarily change slightly due to cuts and bruises on the skin or weather conditions. The veins in the finger vein of one person are thought to be distinct from those in the fingers of another. If you're identical twins, you're likely to have distinct veins in your fingers [14].

Table (1.1): Comparison of Biometric Technology, the Letter H means High, M means Medium, and L means Low [4].

Biometric identifier	Universality	Distinctiveness	Permanence	Collectability	Performance	Acceptability	Circumvention
DNA	H	H	H	L	H	L	L
Ear	M	M	H	M	M	H	M
Face	H	L	M	H	L	H	H
Facial thermos	H	H	L	H	M	H	L
Finger vein	M	H	H	M	H	M	M
Gait	M	L	L	H	L	H	M
Hand geometry	M	M	M	H	M	M	M
Hand vein	M	M	M	M	M	M	L
Iris	H	H	H	M	H	L	L
Keystroke	L	L	L	M	L	M	M
Odor	H	H	H	L	L	M	L
Retina	H	H	M	L	H	L	L
Signature	L	L	L	H	L	H	H
Voice	M	L	L	M	L	H	H

Table (1.2) describes the comparative evaluation of the major biometric technologies. Each of these techniques has its own merits and shortcomings. In all, finger vein has the least expensive method, and it is usually used in criminal's identification. Nevertheless, there may be difficulty in image capturing of people due to their work and age which

may make the fingerprint capturing incomplete and accurate. In addition to this, the uses or the application of finger veins are limited as it can be replicated [4].

Table (1.2): Comparisons of the Major Biometrics Techniques[4].

Biometric	Accuracy	Size of Template	Cost	Security Level	Long-term Stability
Finger vein	High	Medium	Low	High	High
Finger print	Medium	Small	Low	Low	Low
Facial Recognition	Low	Large	High	Low	Low
Iris Scan	High	Large	High	Medium	Medium
Speaker Recognition	Low	Small	Medium	Low	Low
Hand Geometry	Low	Large	High	Low	Low

1.4 Concepts of the Biometric Identification Systems

Any biometrics system including two phases first phase is enrollment phase and second is recognition phase. The recognition phase divided to two things which are verification and identification. During the enrollment phase the biometrics data are captured and generate digital image then Preprocessing apply to digital image for removing unwanted data and apply the post-processing than store this data in database. In the case of identification process the finger vein acquired from one person is compared with all the finger veins which store in database. Also it is known as (1: M) matching. It is used in the process of seeking the criminals. In the verification process the person's finger vein is verified from the database by using matching algorithms. Also it is known as (1:1) Matching. It is the comparison of a claimant finger vein against enroll finger vein, initially the person enrolls his/her finger vein into verification system, and the result show whether the finger vein which take from the user is matching with the

finger vein store as a template in database or not match [15].The Enrollment, Identification and Verification process shown in the Figure (1.2).

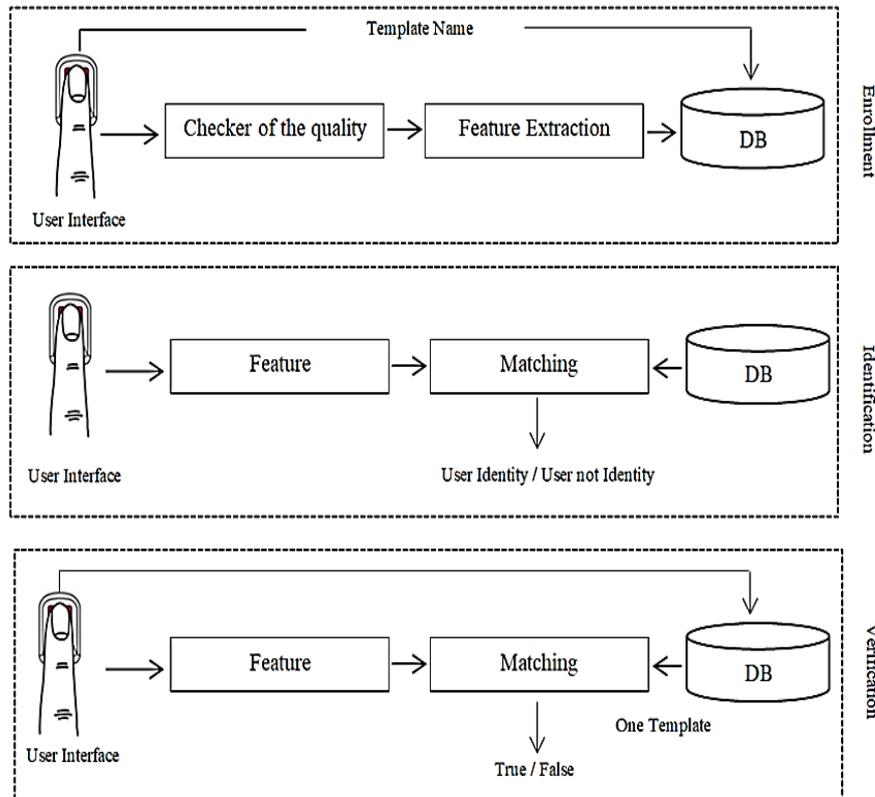


Figure (1.2): Process of Enrollment, identification and verification [15].

1.5 Advantages and Challenges

Since a decade ago, biometric recognition method based on vascular patterns has received much attention among the researchers and technologists. The most popular vascular patterns are hand-vein and finger-vein. A finger-vein biometric system used a specifically designed device to capture image of the vein patterns. The vein is inside of the human's finger. Finger-vein features exhibit several other excellent advantages and these include [16] :

1. The finger-vein patterns are unique for every individual, including identical twins. Thus, it offers good dissimilarity between each individual.

2. The finger-vein patterns are permanent. It does not change with time.
3. The finger-vein patterns are not obscured nor it easily to be replicated or damaged because it is located underneath the human's skin. In another word, it is invisible to human's eyes.
4. The finger-vein patterns acquisition is contemplated to be very user-friendly. The vein pattern images are captured non-invasively. The device using a contactless sensors concept ensuring convenience and hygiene for the user.
5. Every individual commonly has ten available fingers. Therefore, if something unforeseen incident happens to any one of the fingers, other fingers are used as replacement for authentication.
6. Finger-veins can only be captured from a living body; hence, if a person is dead, it is impossible to steal his identity.

Regardless of the advantages mentioned above, there are challenges that still need to be dealt with in order to achieve the higher performance required in real-world deployments of finger-vein biometric identification systems [16].

1. The alteration of the finger vein pattern shape occurs due to physiological growth. The vein pattern is constant between the ages of 20 to 50 years since there is no important development adult stage. Hence, this means it is necessary to be replacing the enrollment template every 5 or 10 years.
2. Changes in finger vein patterns also occur from shrink of vascular system. This is because of falloff in bones and muscles due to growth age.
3. Image acquisition using near-infrared imaging that can temporal change thickness and contrast of finger veins taken because of variations in the volume of veins blood is another challenge. This

occurs because of extreme physical activity or ambient temperature, which temporally effect of physiological changes.

4. The research still on-going if disease can affect the finger vein pattern; they believe that diseases like diabetes, tumors, hypertension, or metabolic disorders can affect finger vein.
5. Vein recognition efficiency can be badly affected by the development in age and the physical revolutions in the body. The low environment temperature raises the blood viscosity, which leads to finger vein contract, that is, it grows into smaller in width. This contract has reduced surface area finger veins standard, and thus produces the poor image patterns. Though, in atmosphere elevated temperature, the blood becomes less viscous, that leads to finger veins enlargement.

1.6 Public database

There are several publically available finger-vein databases as shown in Table (1.3). However, these databases are not considered as standard databases for finger-vein application. These databases provide samples for more than 100 subjects except for UTFV database (60 subjects). Some of the samples provided are low-quality samples with a high degree of noise exist. In addition, some of the samples are terribly skewed (misaligned). Therefore, some of the databases may suit to one's applications and some may not. In any fingerprint capturing device, a person's finger is normally guided throughout the capturing process. Similarly, in finger-vein capturing device, there is a specific space to place a finger. Therefore, misaligned/skewed finger-vein images which are provided in by publically available database are irrelevant and there is no significant reason to use them as the database that we will be working on is the database named (SDMULA-HMT), the details of which are mentioned in the table below. The device used can be defined as it is a device that will capture the human finger-vein image and will be used for biometric security purposes such as

authentication, verification and identification. A near-infrared light (NIR) will be emitted by a bank of NIR Light Emitting Diodes (LEDs) which will penetrate the finger and are absorbed by the hemoglobin in the blood. The areas in which the NIR rays are absorbed (i.e. Veins) thus appear as dark regions in an image conveyed by a CCD camera located on the opposite side of the finger [17].

Table (1.3): The existing public finger-vein databases[17].

Database	No of images	No of Subject	Image No. per Finger	Image resolution	Format
THU-FVFD1 [18]	440	220	1	720x576pxl	.BMP
UTFV [19]	1440	60	4	672x380pxl	8 bit gray scale .PNG
MMCBNU_6000 [20]	6000	100	10	480x640pxl	.BMP
HKPU-FV [21]	6264	156	12/6*	513x256pxl	.BMP
SDMULA-HMT [22]	3816	106	6	320x240pxl	.BMP

1.7 The Aim of Thesis

Instead of other biometric systems, the aim is to build an integrated system to identify people based on finger vein patterns using two advanced technologies. The first technique is using an optimized K-mean algorithm based on the difference between the intensity of the pixels of the image that we have made some improvements to during the pre-processing stage in which we want to isolate the finger area from the background that contains information that we do not need because the finger pattern is hidden under the finger area. After extracting the finger pattern, we try, using the second technique which is called (GPE) Global Pattern Extraction, to obtain an

accurate shape for this pattern by knowing the path and directions of the finger vein lines and knowing the angle of curvature of each line in the pattern, thus improving the shape of the pattern to be ready for training using a new framework in deep learning, which is the U-Net framework that makes a cut At the pixel level and by using the loss PU positive unlabeled function equation of the positive and negative approach, which does not require the labeling of all images, and there will be automatic label, which saves a lot of effort and time in the process of filling data by hand such as in traditional deep learning and thus we get trained patterns and test the system after the training process to know the identity of the person through the similarity of the veins of the fingers of the same person.

1.8 Literature Review

Many researchers showed their interest about finger vein recognition systems some of the published works related to the objectives of this thesis are given in the following:

- Zhang et al. proposed a single finger vein image enhancement based gray level grouping model (GLC). Gabor filtering is the main tool that is used to enhance the finger vein images. In this approach, the experimental results show that circular Gabor filter (CGF) based gray-level grouping (GLG) is able to enhance the finger vein image effectively and improved the image for the recognition stage [23].
- Yu et al. proposed a histogram equalization approach to increase the contrast of the resulting image. In addition, a fuzzy-based multi-threshold algorithm, which considers the characteristics of the vein patterns and the skin region, this fuzzy-based multi-threshold algorithm is not only straightforward, but it also increases the contrast between the vein patterns and the background [24].

- Yusoff et al. proposed method resampling to reduce the number of pixels in the vein image, thus reducing the cost of computation. Then, Difference of Gaussian (DoG) and thresholding are used to segment the vein image. Finally, thinning is applied to get single line veins. In addition to that, image thresholding is the main approach that has been used in those method which they based on the manually threshold value selection for the finger vein image banalization approach [25].
- Ruqaiya at el. proposed and develop globalization approach for finger vein recognition system-based K- nearest neighbor algorithm (KNN) classification approach and finger vein image analysis tools. They rely on the KNN to verify and calculate the performance of the recognition system after some globalized features are extracted from the finger vein images. Experimentally, the proposed system for finger vein verification system based KNN achieved accuracy starting from 55.84% till 92.21% [26].
- Yu at el. proposed a convolutional neural network (CNN) based local feature descriptor for finger vein recognition. The proposed system showed that the convolutional neural network based local feature descriptor (CNN-CO) can exploit and extract the discriminative finger fine features. Basically, the complex CNN filter (kernels) is strong enough to find the corresponding local features base Gabor filter banks. The experimental results of the proposed system show that finger vein recognition system based local feature descriptor demonstrated effectiveness performance results when it has been applied to two public finger vein datasets [27].
- Manmohan at el. proposed a finger vein recognition system based maximum edge position detection. The proposed system is based on the feature collection stage that use the sign and the magnitude of the finger vein edges which illustrate and capitalize the differences of the gradient

feature values in different directions. The proposed maximum values of the finger vein edges based different gradient directions show that the classification accuracy has achieved 89.47% and approximately error rate up to 12.48% [28].

- Manisha Sap kale et. al. proposed an embedded finger vein recognition and classification system for human authentication. The proposed system based on a novel finger vein image recognition system. The proposed algorithm is based on using the Gabor filtering for finger vein feature extraction and based on the distance majority for finger vein classification. The experimental results showed that finger vein recognition become more reliable, easier and accurate [29].

1.9 Outline of the Thesis

The other chapters in this thesis are as follows:

- **Chapter 2:** Background Theory which describe and discuss the main background theory that is used in a research as well as the main tools, method, and the approaches of the digital image processing as well as some fundamentals.
- **Chapter 3:** Design and Implementation of the Proposed System which describe and discuss the whole approach of the proposed system and discuss each implanted step as well. Also, have described some algorithms, pseudocode and flowchart that are used in the proposed system.
- **Chapter 4:** Experimental Results which lists and discuss the main results they are achieved in the proposed system using different criteria's such as figures, tables, and graphs.
- **Chapter 5:** Conclusion and Suggestions for the Future Works. Finally, we list some conclusion points that are achieved through our research as well as some suggestions for the future works.

Chapter Two

Background Theory

2.1 Fundamentals of Biometric Identification System

A biometric system is a technology which takes an individual's physiological, behavioral, or both traits as input, analyzes it, and identifies the individual as a genuine or malicious user. Each human being is unique in terms of characteristics, which make him or her different from all others. The physical attributes such as fingerprints, finger veins, color of iris, color of hair, hand geometry, and behavioral characteristics such as tone and accent of speech, signature, or the way of typing keys of computer keyboard etc., make a person stand separate from the rest [30].

2.1.1 Fingerprint-Based Biometric Authentication

Using fingerprints as a biometric means of identifying is the oldest and most often used method because of its extensive user acceptance, accuracy, security and cost-effectiveness. At the global level, the oriented pattern displayed by the ridge flow provides important information for fingerprint analysis at three degrees of detail. At the local level, minutiae are the most significant elements that ensure the fingerprint's uniqueness; they are characterized by regions with local ridges discontinuities. Pores and ridge contours are addressed at a finer level. To develop a recognition procedure, a fingerprint analysis algorithm may employ one or more levels of information. Some particular uses of fingerprint exploitation, such as gender identification and the finding of an individual's ancestors, are now being explored [31].

Figure (2.1) illustrates Fingerprint - based Biometric Authentication, as the basis of work is that the device captures the fingerprint and changes it into a digital form to be employed in the areas of computer science.

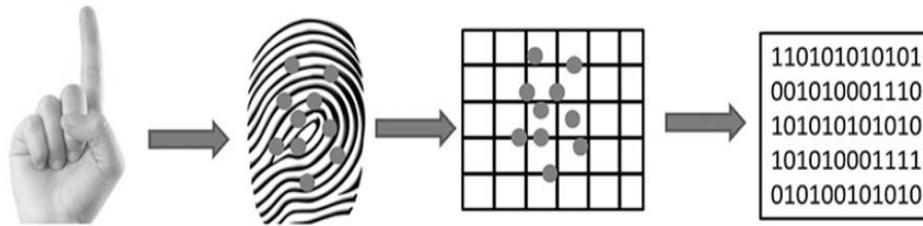


Figure (2.1): Fingerprint-Based Biometric Authentication [30].

2.1.2 Finger Vein-Based Biometric Authentication

Finger vein biometric authentication is a recent identification system in this modern era. This technology is used for wide variety of applications including credit card authentication, automobile security, employee time and attendance tracking, computer and network authentication, and so on. Like fingerprints or iris patterns, finger vein based blood vessel patterns are unique for each individual. Finger vein based blood vessel pattern have high security because the veins are located under the surface of the skin. The fingerprints can be cheated by dummy finger fitted with a copied fingerprint, but the finger vein based identification system is highly secure for authentication. The iris pattern recognition is known for low error rates of authentication, but some users feel psychological resistance to the direct application of light rays into their eyes. In addition to this, precise positioning of the eyes is required for accurate iris authentication. So the iris authentication system is provided with high-cost position adjustment mechanisms for the accurate recognition. For authentication application the pattern of the finger vein is stored in a database [32].

The finger is placed on an attester terminal which contains a near-infrared, light emitting diode light source and a monochrome charge coupled device camera. The hemoglobin present in the blood absorbs the near infrared light emitting diode light and makes the vein to appear as dark pattern. The recorded image is digitized and stored in the database. During authentication, the finger vein is scanned and is compared with the image in the database [7].

Figure (2.2) named Finger vein -based Biometric Authentication. Its working idea is to convert the unique vein pattern into a digital form that may be employed in many disciplines of computer science, such as artificial intelligence and machine learning.

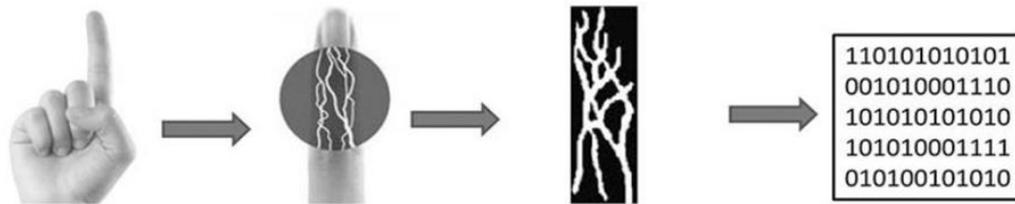


Figure (2.2): Finger vein - Based Biometric Authentication [30].

2.2 Digital Images Types

There are different types of images in the term of digital image processing include:

2.2.1 Binary Image

There are just two potential pixel intensities for binary images: black and white. Because it only contains black and white, it is sometimes described to it as a 1- bit monochromatic image. Office papers, handwritten writing, line drawings, technical graphics, etc. are all examples of typical binary image applications. The outcome of the scanning process is a series of pixels, each of which can be either black or white. It's a zero-to-one ratio: black pixels are (0), white pixels are (1) [33]. The examples of binary images are shown in the following figure (2.3).



Figure (2.3): Examples of Binary Images.

2.2.2 Gray Scale Image

Gray-scale images are referred to as monochrome, or one-color image. They contain brightness information only, no color information. The number of different brightness level available. The typical image contains 8-bit/pixel (data, which allows us to have (0-255) different brightness (gray) levels). The 8-bit representation is typically because the byte. Here are some samples of grayscale images shown in Figure (2.4).

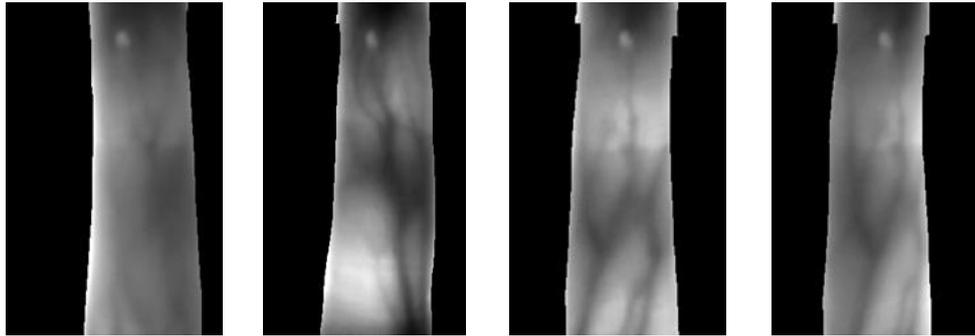


Figure (2.4): Gray Scale Images[22].

Equation (2.1) illustrate the conversion of color image to gray scale and this equation will applied to each pixel in an image converted.

$$Gray(x, y) = \frac{R(x, y) + G(x, y) + B(x, y)}{3} \quad (2.1)$$

Where $R(x, y)$ is denudated as a red is channel and $G(x, y)$ is a green channel and $B(x, y)$ is a blue channel of an image [34].

2.2.3 Color Image

Color images can be modeled as three-band monochrome image data, where each band of data corresponds to a different color. The actual information stored in the digital image data is the gray-level information in each spectral band. Typical color images are represented as red, green, and blue (RGB images) using the 8-bit monochrome standard as a model, the corresponding color image would have 24-bits/ pixel [34]. Figure (2.5) illustrate some example of color images.

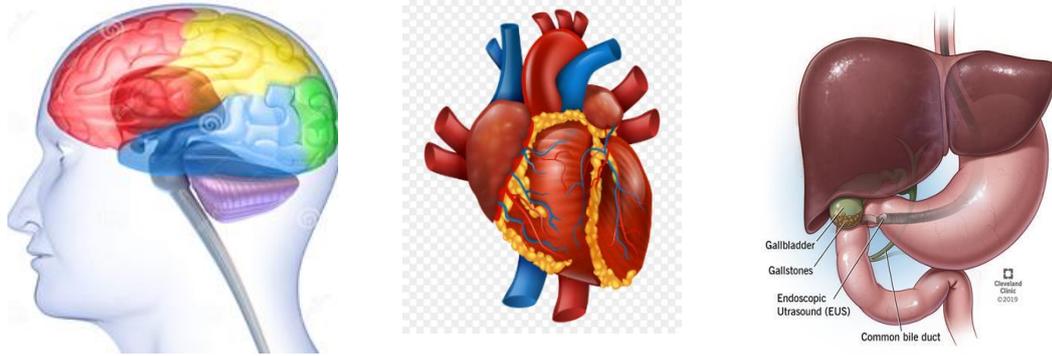


Figure (2.5): Examples of Color Images[35].

2.3 LAB Color Model

For both image modification (tone and contrast editing) and compression, The "*Lab color space*" system is a good decouples of intensities (as indicated by L^* luminosity) as well as color (a^* for the red color negative green color and b^* for green color negative blue color). Studies show that the $L^* a^* b^*$ color system separates brightness information from color information more than any other color system. Here are the formulae for "*Lab color space*" components.

$$L^* = 116 f\left(\frac{Y}{Y_n}\right) - 16 \quad (2.2)$$

$$a^* = 500 \left(f\left(\frac{X}{X_n}\right) - f\left(\frac{Y}{Y_n}\right) \right) \quad (2.3)$$

$$b^* = 200 \left(f\left(\frac{Y}{Y_n}\right) - f\left(\frac{Z}{Z_n}\right) \right) \quad (2.4)$$

Where, being $t = \frac{X}{X_n} = \frac{Y}{Y_n}$ or $\frac{Z}{Z_n}$ and $f(t) = \sqrt[3]{t}$ if $t > \epsilon^3$ and $f(t) = \left(\frac{t}{3 \times \epsilon^2}\right) + \left(\frac{4}{29}\right)$ otherwise. Where $\epsilon = \frac{6}{29}$ and X, Y, Z define the color incentive considered and X_n, Y_n, Z_n define a specified white achromatic reference illuminant [36].

2.4 Image Processing in Digital Form Operations

Digital image processing includes a variety of techniques, including image enhancement, image analysis, and image compression. The talk will be about image enhancement.

2.4.1 Image Enhancement

Image improvements are used to make an image appear more appealing to the viewer. Adjusting the image's brightness or contrast may be necessary if it's too dark or too bright. The term "spatial filtering" refers to an image operators that use a fixed integer matrix of the same size to alter the gray value of each *pixel* (x, y) based on the pixels value in a square neighborhood centered at (x, y) . The integer matrix is mentioned to as a filter, mask, kernel, or a window, depending on the context. Moving the filter mask around an image is all that is required to do spatial filtering. The filter's response at each *pixel* (x, y) is determined based on a specified relationship at each pixel. using Equation (2.5)[37], we can see how the spatial domain process often works [38].

$$I(x, y) = T [C(x, y)] \quad (2.5)$$

In this case, the production image is $I(x, y)$, the contribution image is $C(x, y)$, and the operator T is specified over a region around points (x, y) .

Different types of spatial filters are used in digital image processing to improve images, such as:

1. Gaussian filter

Gaussian filtering is used to blur images, eliminate noise, and enhance the image features. A linear filter, a Gaussian filter is used. Equation (2.6) [37] summarizes the main characteristics of the Gaussian function utilized in the image enhancement process.

$$G(x) = 1/\sqrt{2\pi\sigma^2} e^{-x^2/2\sigma^2} \quad (2.6)$$

As $G(x)$ is the output image and σ is the standard deviation of the distribution, and has a significant impact on its actions. The mean of the distribution is assumed to be 0 [39].

2. Median Filter

For signal smoothing, a median filter is a nonlinear filter. For reducing impulsive-type noise from a signal, it is excellent. Digital signal processing systems frequently utilize two-dimensional versions of this filter to eliminate noise and speckles from images. Sliding a window across an image does the median filtering. The filtered image is created by setting the input window's median value to the output image's center, and then subtracting that value from the input window's median value. When utilizing the median filter, instead of the average operation, the gray levels in a nearby area are replaced. Median filtering is a method of arranging gray values in increasing or decreasing order, then selecting the middle one in each row[40].

$$\hat{O}(x, y) = \underset{(r,c) \in SS_{xy}}{\text{median}}[v(r, c)] \quad (2.7)$$

As $\hat{O}(x, y)$ is output image, $v(r, c)$ is the value vector for sub- image, SS_{xy} is a sub image (region) centered on points (x, y) [37].

3. Average (Mean) Filter

The average (or mean) filtering method reduces the amount of intensity fluctuation between adjacent pixels in order to smooth the images. When using an average filter, it moves across the image pixel by pixel, exchanging each pixel's original value with the regular value of its neighbors [41].

$$Y(i) = \frac{1}{M} \sum_{j=0}^{M-1} x[i + j] \quad (2.8)$$

Where M is the number of point used in the average filter, x denotes the input, y denotes the output signal [37].

4. Soper Operator Filters

Differential filters can be used to enhance various aspects of a mask's design. Convolution masks are applied to lines in the horizontal, vertical, and diagonal directions and this kernel highlights the limits in all directions [42].

$$\begin{array}{cccc}
 \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 0 \end{bmatrix} &
 \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix} &
 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} &
 \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} \\
 \text{Vertical} & \text{Horizontal} & \text{Diagonal 1} & \text{Diagonal 2}
 \end{array}$$

5. Wiener filter

Wiener used for smoothing and removing noise from images. The inverse filtering and noise smoothing procedures effectively reduce a total mean square error during acquisition and transmission; images are often corrupted by noise. Denoising is therefore the basic problem in image processing application. The purpose of denoising is then to reduce noise level, while preserving the image features such as edges and textures and so on as accurately as possible. Many image denoising methods have been proposed in the literature. Among these methods, Wiener filter has received an increasing attention because of its simplicity and effectiveness. Up to now, a vast literature has emerged on image denoising based on Wiener filter [43]. Equation (2.9) denotes the Wiener filter numerical representation [44].

$$\text{Wiener}(f_1, f_2) = \frac{H(f_1, f_2) \times S_{xx}(f_1, f_2)}{|H(f_1, f_2)|^2 S_{xx}(f_1, f_2) + S_{\eta\eta}(f_1, f_2)} \quad (2.9)$$

For the original image and for the added noise, the power spectrums are $S_{xx}(f_1, f_2)$ and $S_{\eta\eta}(f_1, f_2)$ respectively. $H(f_1, f_2)$ is the blurring filter.

2.4.2 Image normalization

Changing the range of pixel intensity levels is called normalizing in image processing. For example, images with low contrast caused by glare might benefit from this technology. Contrast stretching or histogram stretching is another term for normalization. Dynamic range expansion is a more generic

term used in data processing domains like digital signal processing[45]. Equation (2.10) depicts the fundamental idea of an image normalizing implemented based on the formal Z-score function [46].

$$x' = \frac{x - \bar{x}}{\sigma} \quad (2.10)$$

As \bar{x} is the mean of the intensity pixels values, and σ is the standard deviation.

2.4.3 Image Adjustment

Adjusting the levels, contrast, gamma, hue, saturation, and brightness of an image are all examples of image adjustment. An overexposed image can be corrected by using these techniques. It is also possible to enhance other images with the right correction [47]. Equation (2.11) depicts the fundamental idea of an image adjustment [37].

$$X_{ij} = \frac{L_{out} + (H_{out} - L_{out}) \times (X_{ij} - L_{in})}{(H_{in} - L_{in})} \quad (2.11)$$

There are two parameters in the output image: the lower and higher bounds of intensity level, L_{out} and H_{out} . L_{in} and H_{in} represent the input images lowest and greatest intensity levels, respectively.

2.4.4 Morphological Operation

Morphological image processing is a collection of nonlinear operations related to the shape or morphology of features in an image. A morphological operation on a binary image creates a new binary image in which the pixel has a non-zero value. Morphological operations transform the image. In this paper, erosion is applied to detect the tumor. The erosion of A by B is given by the expression:

$$A \ominus B = \langle (i, j): B_{(i, j)}(A) \rangle \quad (2.12)$$

Where, A is the binary image, B is the structuring element and (i, j) is the center pixel of structuring element [48].

2.5 Machine learning

Machine learning (ML) is used to teach machines how to handle the data more efficiently. Sometimes after viewing the data, we cannot interpret the extract information from the data. In that case, we apply machine learning. With the abundance of datasets available, the demand for machine learning is in rise. Many industries apply machine learning to extract relevant data. The purpose of machine learning is to learn from the data. Many studies have been done on how to make machines learn by themselves without being explicitly programmed. Many mathematicians and programmers apply several approaches to find the solution of this problem which are having huge data sets [49].

Machine Learning relies on different algorithms to solve data problems. Data scientists like to point out that there's no single one-size-fits-all type of algorithm that is best to solve a problem. The kind of algorithm employed depends on the kind of problem you wish to solve, the number of variables, the kind of model that would suit it best and so on [50].

In supervised learning, we need to have input and output data for the system to predict. In an unsupervised system, we need only input and the output is obtained by finding relationships between data and using algorithms to create these relationships in order to get the output.

Figure (2.6) shows learning styles influence of (ML), there are two types of machine learning supervised (classification) and unsupervised (clustering).

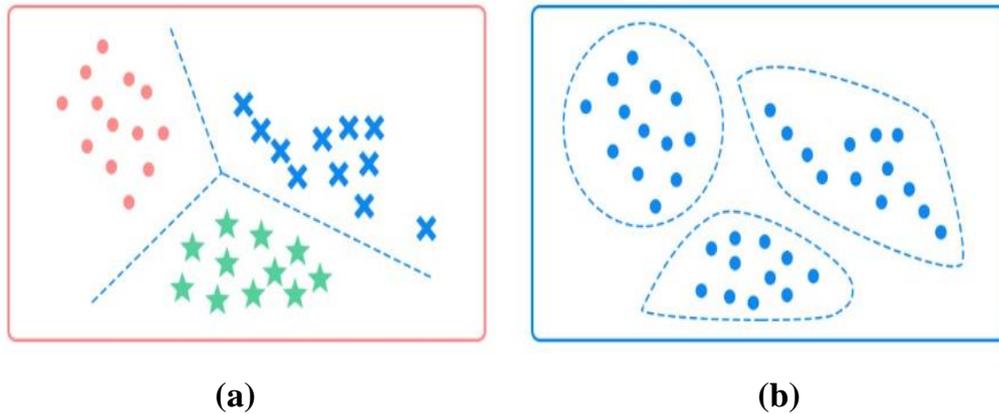


Figure (2.6) Learning Styles Influence Machine Learning (a): supervised Algorithms, (b): Unsupervised Algorithms [51].

2.5.1 Background Clustering Algorithm Theory

In this section, we will talk about some of the clustering algorithms used in the field of machine learning; there are many types of clustering algorithms:

1. K-means Clustering

An adaptive technique, as defined by K means, is used to determine the optimal outcome for each repeat. The number of clusters necessary then is calculated. It is necessary to split the data points into K groups in order to use this method of clustering, smaller groupings with a greater K mean more specificity. An increase in the number of cluster with such a smaller K value is more likely. Labels are generated as a result of the algorithm. The K categories are used to categorize each piece of information. In k-means clustering, the centroid of each group is identified. The cluster's beating heart, the centroids gather and include the points that are most closely located to their centers. This method uses a distance measure to reduce the total number of clusters (iteration after iteration) via an integration technique. Finally, we need a huge cluster with all of the objects. Let $[z_1, z_2, z_3, \dots, z_q]$ consist of a set of the vector data (samples). These vectors are written as follows:

$$Z = \begin{pmatrix} Z_1 \\ Z_2 \\ Z_3 \\ \cdot \\ \cdot \\ \cdot \\ Z_n \end{pmatrix} \quad (2.13)$$

Using k-means clustering, the objective is to divide the set Q of reflections into k [$k \leq Q$] splitting cluster set $C = [c_1, c_2, c_3, \dots, c_k]$ so that the next standard of optimality is content:

$$\sum_{i=1}^k \sum_{Z \in C_i} \|Z - M_i\|^2 \quad (2.14)$$

Mean vector (or centroid) of samples in set C_i are M_i , and Arg is the argument vector norm. Typically, As a result of the application of the Euclidean norm, the familiar Euclidean distance between a sample in C_i and the value of M_i is given the name $\|Z - M_i\|^2$. In words, this equation says that we are interested in finding the sets $C = [c_1, c_2, c_3, \dots, c_k]$ such that the sum of the distances from each point in a set to the mean of that set is the minimum value [52].

2. Fuzzy C-mean Clustering

Using this approach, each data point associated with each cluster center is assigned membership based on a cluster's distance from the data point. The more the data's affinity for a specific cluster center, the closer it is to the cluster center itself. There's no doubt about it: the summing of each data point's membership should be one. The membership and cluster centers are modified based on a formula after each repetition.

$$\sum_{i=1}^N \sum_{j=1}^C u_{ij}^r \|X_i - c_{ij}\|^2 \quad 1 \leq R \leq \alpha \quad (2.15)$$

Where R is anything larger than 1; any real number greater than 1; $\|*\|$ any norm that expresses similarity between any measured data and any center [53].

2.6 Deep learning

Deep learning is nothing but many classifiers working together, which are based on linear regression followed by some activation functions. Its basis is the same as the traditional statistical linear regression $WT X + b$ approach. The only difference is that there are many neural nodes in deep learning instead of only one node which is called linear regression in the traditional statistical learning. These neural nodes are also known as a neural network, and one classifier node is known as a neural unit or perception. Another contrasting point need to be noticed is that in deep learning there are many layers between the input and the output. A layer can have many hundreds or even thousands of neural units. The layers which are in between the input and the output known as the hidden layers and the nodes are known as the hidden nodes. The draw-back of the traditional machine learning classifiers is that we need to write a complex hypothesis by ourselves, while in the deep neural network it is generated by the network itself, which makes it a powerful tool for learning nonlinear relationships effectively [54].

2.7 Convolution neural network (CNN)

CNNs was first proposed in the 1980s. It is inspired by the cat's cortex. The LeNet-5 system was a classical model of a convolution neural network. Its error rate was only 0.9% on the MNIST data-set. It had been widely used to identify handwritten checks on banks, but it did not recognize large images. With the development of Graphics Processing Unit (GPU) technology, researchers used an efficient GPU supported program to solve the ImageNet problem in 2012, which made the convolution neural network again popular. In fact, one of the bottlenecks of deep neural nets was that it took a long time for training because of the many hidden nodes in its network. But as the

GPUs become faster in parallel computing, this bottleneck was overcome. At present, the convolution neural network is a hot topic in the field of voice data analysis and image recognition. The convolution neural network has a network structure with share permission, which makes it closer to the biological neural network. This network structure in the convolution neural network can effectively reduce the complexity of the network model and also reduce the number of the weights. Mainly, it is more efficient to deal with high dimensional images, which can directly consider the image as the input of the entire network and effectively avoid the complex feature extraction and reconstruction of the traditional algorithm. In the process of image recognition, the convolution neural network has a high degree of invariance in scaling, tilting, translating, and other forms of image deformation [54].

Below is the structure of convolution neural network:

1. **Convolutional layer** the input image or the last feature map is convolved by a randomly generated kernel (or filter) of size (height, width, channel). The feature map F_k is calculated as:

$$F_k = \left(\sum_i W_{ki} * X_i \right) + b_k \quad (2.16)$$

Where W_{ki} the sub-kernel of the i th channel is, X_i is the i_{th} input channel, $*$ represents the convolution operator, and b_k is a bias term. Since the convolutional layer is associated with L input channels, X contains $M \times M \times L$ values, each kernel W_{ki} contains $N \times N \times L$ weights. Accordingly, the number of parameters in a convolution block composed of K feature maps is equal to $K \times M \times M \times L$.

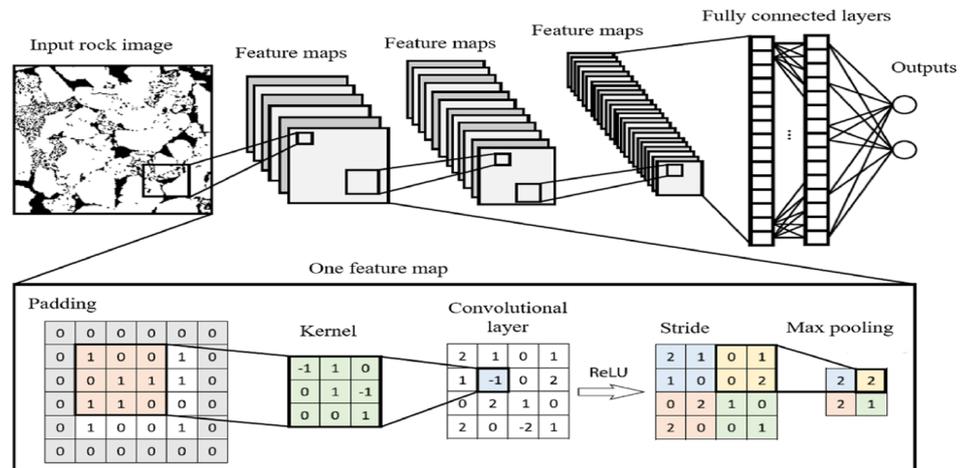


Figure (2.7): Convolutional Neural Network model[55].

2. **Padding:** In order to obtain a convolutional layer of the same size as the input matrix, a zero padding cells containing some rows and columns with zero value are added around the input matrix.
3. **ReLU:** (Rectified Linear Unit) layer The ReLU layer is an activation function that changes negative values to zero.
4. **Maximum pooling:** the pooling layer, also known as down sampling, combines all values in the pooling window into one value in the next layer. For example, the maximum pooling uses the maximum or average pool to calculate the average value in each pool cluster.
5. **Stride Convolution** and maximum merge are performed with an offset of "n" pixels, which is called stride. A stepping window larger than one pixel will lead to a smaller image. It controls the size of feature maps in convolution and max pooling layers.

2.8 Deep Semantic Segmentation

The semantic image segmentation task consists of classifying each pixel of an image into an instance, where each instance corresponds to a class. This task is a part of the concept of scene understanding or better explaining the global context of an image. Image segmentation is a computer vision task in which we label specific regions of an image according to what's being shown.

More specifically, the goal of semantic image segmentation is to label each pixel of an image with a corresponding class of what is being represented. Because we're predicting for every pixel in the image, this task is commonly referred to as dense prediction. The goal is simply to take an image and generate an output such that it contains a segmentation map where the pixel value (from 0 to 255) of the input image is transformed into a class label value (0, 1, 2, ... n) [56].

2.8.1 Semantic Segmentation Based Deep Learning Method

Semantic Segmentation often requires the extraction of features and representations, which can derive meaningful correlation of the input image, essentially removing the noise. Different semantic segmentation methods that use CNN as the core architecture. The architecture is sometimes modified by adding extra layers and features, or changing its architectural design altogether. CNN consists of a convolutional layer, a pooling layer, and a non-linear activation function. When it comes to semantic segmentation, we usually don't require a fully connected layer at the end because our goal isn't to predict the class label of the image. In semantic segmentation, our aim is to extract features before using them to separate the image into multiple segments. However, the issue with convolutional networks is that the size of the image is reduced as it passes through the network because of the max-pooling layers. To efficiently separate the image into multiple segments, we need to up sample it using an interpolation technique, which is achieved using deconvolution layers [57].

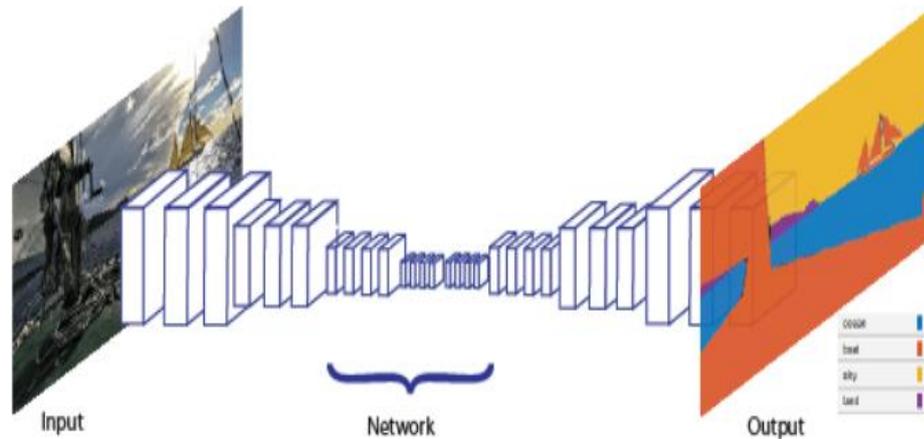


Figure (2.8): Semantic Segmentation processes [58].

2.9 Positive Unlabeled Learning Scheme

The use of CNNs to train image classifiers using images of finger veins has been proposed in recent work on image classification. Because of lack of the training dataset and the Ground Truth (actual labels) of the correct finger vein lines for the human finger vein images, researchers use the whole finger vein images to train their model instead of using the actual finger vein pattern regarding to the time- and effort-intensive process. Also, because of the difficulty of manually labeling a representative collection of finger vein patterns regarding to the positive and negative label's location, to fulfill acceptable performance criteria, the number of tagged negatives must be 10 times greater than the number of positives. Researchers developed a classifier for positive and negative areas using only a few identified positive regions (pixels-based) and the rest of the unlabeled regions (other pixels). In order to decrease classification errors on labeled data (finger vein lines), Mainly, by the PU learning scheme overfitting can be reduced and trained high-accuracy finger vein pattern identification with extremely few labeled data points (pixels) [59].

2.9.1 Auto Encoder-Based Classifier Regularization.

When including the auto encoder component, the classifier is divided into the following two components: an encoder network composed of all layers except the final linear layer and the linear classifier layer. These networks was denoted as f and c , respectively, with the full network, g , being given by $g(x) = c(f(x))$. Furthermore, a deconvolution (also called transposed convolutional) decoder network, d , which takes the output of the feature extractor network and returns a reconstruction of the input image $\bar{x} = d(f(x))$. The objective function is then modified to include a term penalizing the expected reconstruction error over all images in the dataset; d , with weight γ and the GE binomial equation will be used in second models of the proposed system [59].

$$E_{x \sim P} [L(c(f(x)), 1)] + \lambda \sum_{k=1}^N q(k) \log p(k) + \gamma E_{x \sim D} [||x - d(f(x))||_2^2] \quad (2.16)$$

Where the $E_{x \sim P} [L(c(f(x)), 1)] + \lambda \sum_{k=1}^N q(k) \log p(k)$ is the section of positive unlabeled loss function used during the training to obtain the pattern region. Instead of labeling two classes then one class will be labeled and the other will be the probability of the first class subtracted from one value. The second section $\gamma E_{x \sim D} [||x - d(f(x))||_2^2]$ in equation above represent the decoder operation.

2.10 Digital Image Quality Metrics

Digital images are rapidly finding their way into our daily lives due to the explosion of information in the form of visual signals. This images/video often passes through several processing stages before they reach to their end-users. In most cases, these end-users are human observers. Through different processing stages, as acquisition, compression, and transmission, images are subjected to different types of distortions which degrade the quality of them. For example, in image compression, loss compression schemes introduce

blurring and ringing effects, which leads to quality degradation. To maintain, control, and enhance the quality of images, it is essential for image communication, management acquisition, and processing systems to assess the quality of images at each stage. Image's quality metrics have become more and more popular in the image processing community. The definition of evaluation mechanisms to assess the quality of image plays a major role in the overall design of video communication system [60].

There are two main classes of quality assessment: subjective and objective. The subjective assessment involves the use of human observers, while the objective assessment is without human observers, for example using measurement devices or algorithms.

2.10.1 Subjective Quality Assessment

The most reliable method for assessing the quality of images is through subjective testing since human observers are the ultimate users in most of the multimedia applications. In subjective testing a group of people are asked to give their opinion about the quality of each image. Subjective quality assessment involves the processes of viewing environment setup, subject's recruitment, and subjects grading, and processing of the subjective results [61].

The MOS is a subjective quality measurement obtained from several human observers, has been regarded for many years as the most reliable form of quality measurement, which is defined according to the following equation:

$$MOS = \frac{1}{S} \sum_{i=1}^s ip(i) \quad (2.17)$$

Where i is denoted as the image score; $P(i)$ is denoted as the image score probability, and S is denoted as the number of observers. It consists of (MOS) of five degrees to evaluate the quality of the image is: Excellent quality, good quality, Acceptable quality, poor quality and Unacceptable quality [62].

Subjective quality assessment methods provide curate and reliable measurement of the quality of visual signals. However, these methods suffer from different drawbacks that limit their applications.

2.10.2 Objective Quality Assessment

Objective image quality measures play important roles in various applications in the image processing, and they are numerical measures. A good objective measure reflects the distortion on image due to blurring, noise, compression, sensor inadequacy and any source can be distorted the image. Sometime objective image quality assessment models typically require the access to a reference image that is assumed to have perfect quality. Objective assessment relies on computational models that can predict the image quality observations of humans. The accurate objective image quality model predicts the image quality sensation of an average human observer in other words, strong correlations to subjective observations are essential when defining a good objective quality model [63].

There are good quality standards, whether at the image level or at the level of a complete learning model. We review some of them:

1. Signal to Noise Ratio (SNR)

SNR is defined as the ratio of the average signal value to the standard deviation of the signal value. Higher SNR value showed a better-quality image and low SNR indicates the certain region of image weakness relative to background noise [64].

$$SNR = 10 \log_{10} = \frac{\text{mean} [I(x, y)]}{SD [I(x, y)]} \quad (2.18)$$

Where $I(x, y)$ is input image, SD is the standard deviation.

2. Mean Square Error (MSE)

MSE is one of the mathematically based metrics. It denotes the difference between the reference and test images. MSE can be calculated using the following equation [65]:

$$MSE = \frac{1}{MN} \sum_{j=1}^M \sum_{i=1}^N (I_{ref}(i,j) - I_{tst}(i,j))^2 \quad (2.19)$$

Where M and N size of image, $I_{ref}(i,j)$ is original image, $I_{tst}(i,j)$ Enhanced image.

3. Peak-Signal to Noise Ratio (PSNR)

PSNR is the ratio of maximum possible power of a signal and power of distortion, and it is calculated by:

$$PSNR = 10 \log \left(\frac{D^2}{MSE} \right) \quad (2.20)$$

Where D denotes the dynamic range of pixel intensities, MSE is the Mean squared error which denotes the power of the distortion [66].

4. Accuracy (Recognition Rate):

Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. Formally, accuracy has the following definition As expressed in (2.21) [67].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.21)$$

Where TP = True Positives, TN = True Negatives, FP = False Positives, and FN = False Negatives.

5. Recall (Sensitivity):

In an imbalanced classification problem with two classes, precision is calculated as the number of true positives divided by the total number of true positives and false positives. As provided in (2.22) [68].

$$Recall = \frac{TP}{TP + FP} \quad (2.22)$$

6. Precision:

Precision is a good measure to determine, when the costs of False Positive is high. For instance, email spam detection. In email spam detection, a false positive means that an email that is non-spam (actual negative) has been identified as spam (predicted spam) [69].

$$Precision = \frac{TP}{TP + FN} \quad (2.23)$$

7. F1-Measurement:

F1 Measurement is needed when you want to seek a balance between Precision and Recall, so what is the difference between F1 Measurement and Accuracy then? We have previously seen that accuracy can be largely contributed by a large number of True Negatives which in most business circumstances, we do not focus on much whereas False Negative and False Positive usually has business costs (tangible & intangible) thus F1 Measurement might be a better measure to use if we need to seek a balance between Precision and Recall AND there is an uneven class distribution (large number of Actual Negatives) [70].

$$F1 - Measurement = 2 \times \frac{2TP}{2TP + FP + FN} \quad (2.24)$$

8. The Dice coefficients:

There are two possible ways to evaluate a model for semantic segmentation: during training, which is known as validation, or after training which is known as testing. A metric, or equations, between the actual mask and the expected mask must be calculated at all times. Using these indicators, we can determine whether or not the model is learning properly. That is, a measure of dice coefficient is the product of the intersections of ground truth and expected mask divided by the sum of ground truth and expected mask, which may be employed as a metric [71].

$$Dice = 2 \times \frac{y \cap \hat{y}}{y + \hat{y}} \quad (2.25)$$

Where y the true (ground truth) is mask and \hat{y} is the predicted mask.

9. The Intersections over Unions(IoU):

To define the term, in Machine Learning, IoU means Intersection over Union - a metric used to evaluate Deep Learning algorithms by estimating how well a predicted mask or bounding box matches the ground truth data. Additionally, for your information .To evaluate a model using the IoU metric, you need to have; the ground truth annotation (either masks or bounding boxes); And the model's predictions of a similar type as the ground truth ones [72].

$$IoU = \frac{y \cap \hat{y}}{y \cup \hat{y}} \quad (2.26)$$

Where y represent the true or ground truth image mask and \hat{y} represent the predicted mask that learned the model and predicted it.

Chapter Three

Design and Implementation of Proposed System

3.1 Introduction

In this chapter, the proposed system will be discussed and explain. In general, two models are proposed here. The first model is designed and implemented based unsupervised learning approach. The second model is designed and implemented based supervised learning approach using new architecture in CNN called U-NET model and new loss function for deep semantic segmentation called PU (Positive Unlabeled). The first model is built on two developments. The first development is to isolate the finger area ROI (Region of interest) from the background using the developed K-mean algorithm, which showed much better results when compared with other clustering algorithms such as K-mean, Fuzzy c- mean, and Otsu thresholding algorithms. The second development was using a new technique called Global Pattern Extraction (GPE) and Global Pattern Optimization (GPO) which determines the path of lines that represent hidden veins inside the ROI. The binary pattern masks generated from the first model will serve as training data for the CNN Framework to predict binary pattern mask. Finally, the predicted masks are evaluated and identified based the similarity between the predicted binary pattern masks. At the end, the human is identified and verified based on the pure predicted binary pattern masks.

3.2 Research Methodology

Designing a completely automated finger-vein detection system in the absence of labeled data continues to be a difficult undertaking to do. Aside from that, in the most recent suggested systems, the entire data set (the entire finger vein images) has been utilized in order to train the model, which results in some biasing in terms of the data and environmental similarities, such as skin color, backdrop, and other aspects. On the other hand, the finger vein

lines should be regarded as the primary pure data because they are the primary pattern upon which it is built. This work proposes a completely automated, unsupervised learning strategy for the automatic production of training finger vein line data. This technique eliminates the challenges associated with manually labeling training datasets by employing a fully automated way to build them based on certain complex algorithms and procedures. Figure (3.1) shows a research methodology for this proposal, which is designing and implementing a fully automated finger vein binary pattern extraction and identification.

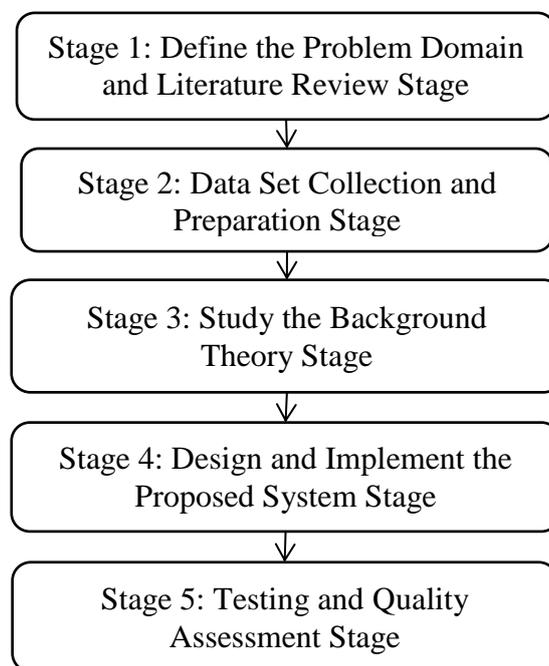


Figure (3.1): Research Methodology of Our Proposed System Framework.

3.3 Implementation Scheme

In this thesis, to implement the fully automated finger vein pattern extraction based positive-unlabeled loss function; two programming platforms have been used. The first platform is MATLAB with windows 10, and the second platform is Python 3.1 with Linux. Many building libraries and toolboxes have been used using image processing toolbox in MATLAB to design and implement the first model (Fully Automated Finger Vein Binary Pattern Extraction based Unsupervised Learning Approach). Also, Tensor

Flow layout has been used to design and implement the second model (Fully Automated Finger Vein Pattern Extraction based Deep Semantic Segmentation).

3.4 Fully Automated Finger Vein Pattern Extraction Approach.

In general, the proposed system is divided in to two models. The first one is designed to generate the training dataset automatically based on unsupervised learning while the second model is designed based on using a deep learning approach for fully automated finger vein pattern prediction. In the second model the training dataset that is obtained from the first model to train a deep semantic segmentation model. Then the trained model is used later on the testing dataset to automatically predict and extract the finger vein pattern.

3.4.1. First Model: Fully Automated Finger Vein Binary Pattern Extraction Based Unsupervised Learning Approach

The first proposed model includes two optimization steps to design fully automated finger vein pattern extraction based on unsupervised learning .The first step is to isolate and enhance the finger area ROI and the second step is to obtain clean and clear finger vein pattern using GPE and GPO techniques. In this model not all the testing cases are passed (binary masks are successfully generated) some cases are failed in this model. For this reason, we used this model on a successful case to generate a decent finger vein binary masks that will used as a training dataset in second model. Then, the other failed cases are isolated to be a testing dataset. There are three stages in this model include preprocessing stage, finger vein localization stage and pattern extraction stage. The initial stage of this model is to enhance the finger vein images by smoothing and sharpening the images using different tools and filters. The second stage involve new K-mean algorithm addressed the

problems of regular K-mean algorithm represented by Clustering Destabilization, and Clusters Randomly Initialization. The new K-mean approach bases on the intensity differences as an activation function instead of using the distance metric which will solve the most significant problems that we talked about in the previous section.

The final stage involves creating GPO and GPE. GPE based on the finger vein image map that determines the direction of each line in the finger pattern. An existing approach is used to extract the finger pattern lines called *miura_repeated_line_tracking* which produces distorted and inaccurate patterns with separate lines. Finally, GPO optimization in this stage is used to correct and enhanced the extracted finger vein lines and the separate lines are connected. The general flowchart of the first model is illustrated in Figure (3.2).

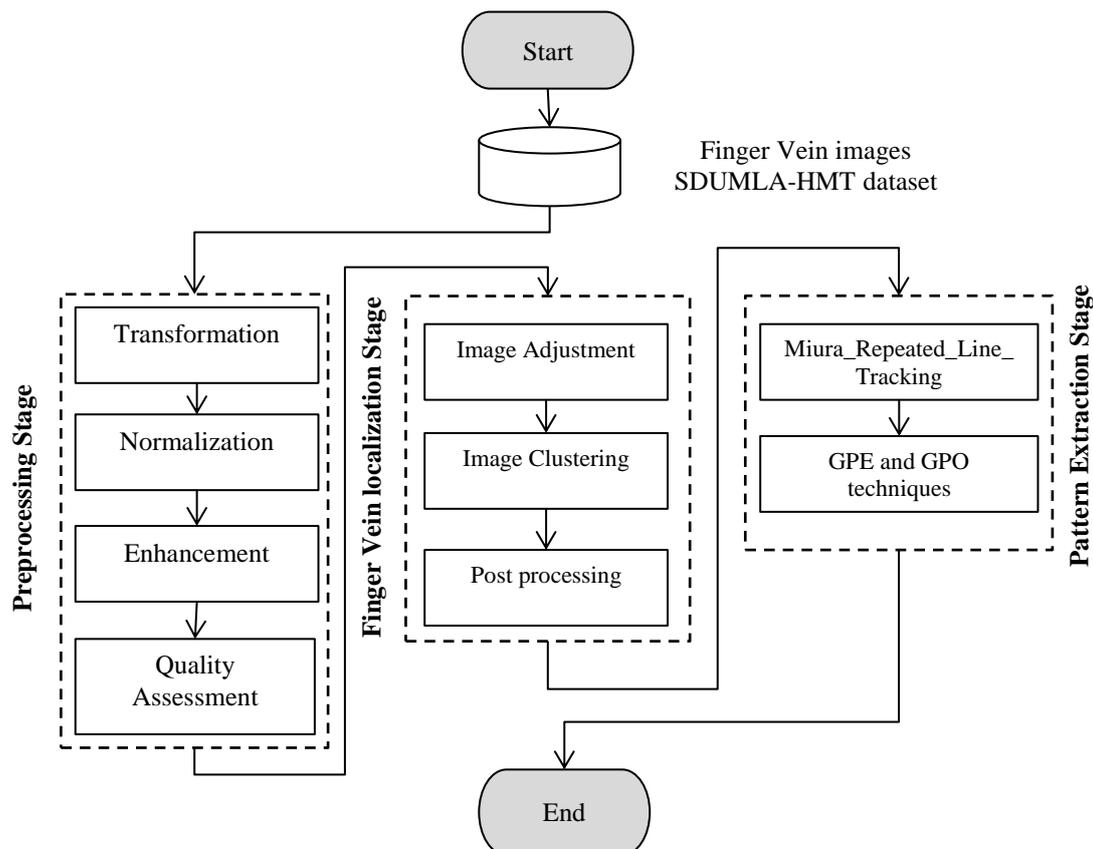


Figure (3.2): First Model System Flowchart (Fully automated Finger Vein Pattern Extraction Based Double Optimization Stages).

A. Stage 1: Preprocessing Stage

In this stage, the finger vein images are preprocessed first. The proposed system is based on using a low-quality finger vein (SDUMLA-HMT) dataset [22]. Here it is proposed some image preprocessing steps that are selected and designed to work well to enhance and improve the finger vein images dataset. As the results, the preprocessing steps eliminate many of the effects of changing and remove the finger vein image noise while still preserving and keeping most of the appearance details that needed to be kept.

1. Step 1: Image Transformation and Image Normalization

At this step, the image is converted from a color image to a gray image, the sequence of converting the image is from colored to LAB colored in order to isolate the lighting from the alpha and beta where the L represent the light and A for Alpha and B for Beta of the chromatic space for this type of transformation, and therefore we worked on the alpha and beta part to try to distinguish the shape of the vein and make it dark and distinct from the rest of the image details. The equations of LAB color space are illustrated in section (2.3) in chapter two. The finger vein image values will be changed from three channels to one channel and the image intensity values in a range between [0...255]. Conversion of the color image to gray image required each pixel in the image consists of three principal colors (*red, green, and blue*). Each color component is represented by one byte. The gray value is obtained by using the equation (2.1), section (2.2.2) in chapter two. This step is applied on all image pixels in order to convert the color image to gray image; the main steps of the pre-processing stage are based on changing the image intensity level or by normalizing the image levels.

After transformation and conversion processes, image normalization will be used. We projected the original intensity level which is typically being between $[0, L - 1]$ which in the most cases has 255. By normalized the image intensity, we will have only 10 deference as it is normalized to be

between $[0, 1]$. Then, we used the mid-range intensity shrinking to adjust the intensity to have 5 intensity levels instead on 10 ones. In this case, we are pretty sure that there are only 5 total clusters in the whole domain which is used to determine the clusters number. This step "image normalization" is implemented based on calculating the mean value of the intensity "gray scales values" that is denoted as \bar{x} which represents the higher distributed intensity, and the local variance that is denoted as σ that represents the other distributed intensity levels. The local image normalization of the finger vein images is described in equation (2.10), section (2.4.2) in chapter two.

2. Step 2: Image Enhancement

The technique of modifying an image such that the output is better suited than the original is known as an image enhancement. At this step wiener filter is used for smoothing and removing noise from finger vein images. The inverse filtering and noise smoothing procedures effectively reduce a total mean square error. The Wiener filter can be expressed in Equation (2.9), section (2.4.1) in chapter two.

3. Step 3: Image Quality Assessment (IQA)

The goal of an (IQA) is to mimic what a human sees when looking at an image. Many approaches for evaluating the effectiveness of image enhancement, generation, and recovery models were developed as a result of their increasing popularity.

In contrast, the vast majority of IQA systems are geared at predicting overall image quality. The result of quality assessment will be discussed in chapter four. Figure (3.3) shows that the image is read from the database file and then entered into the while loop. As long as the condition is met, we will move to the implementation of the pre-processing stages of the original image, and these stages are converting the image to a gray image and controlling the intensity levels of the pixels of the image, and then we evaluate the original image and enhanced image, as we will discuss the results

in Chapter Four, and the Algorithm (3.1) shows the steps used to complete this stage.

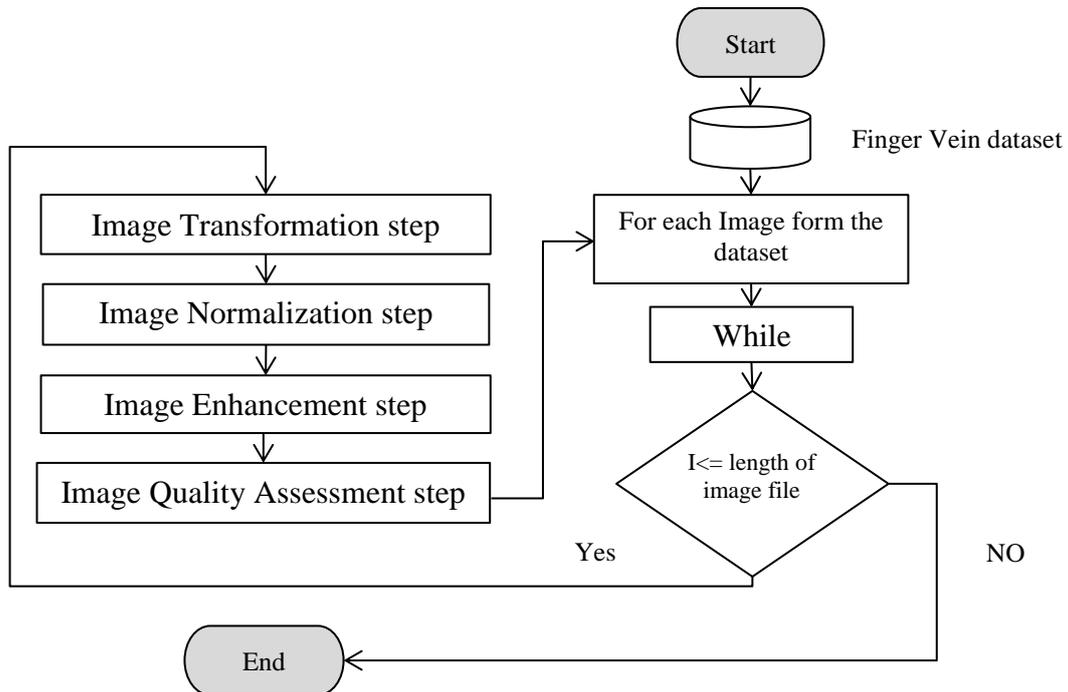


Figure (3.3): Stage 1 Preprocessing Stage Flowchart.

Algorithm (3.1) : Preprocessing Stage Algorithm	
Inputs: I_O : Original Finger Vein Image	
Output: I_P : Preprocessed / Enhanced Image	
1	Begin
2	For each Image from the dataset I_O.
3	Import the image // read the $I_{Original}$ image.
4	Convert image to grayscale using equation (2.1).
5	Calculate the global normalization using equation (2.10).
6	Do image enhancement using Wiener filter illustrated in equation (2.9).
7	Calculate the Mean Square error (MSE) using equation (2.18) in chapter 2.
8	Calculate the Peak Signal to Noise Ratio (PSNR) using equation (2.19) in chapter 2.
9	Calculate the Signal to Noise Ratio (SNR) using equation (2.20) in chapter 2.
10	End For loop
11	Return I_P the enhanced image.
12	End.

B. Stage 2: Finger Vein Image Localization Stage:

In this stage, the whole finger vein image is localized which the finger object is isolated from its background based unsupervised learning approach. Localization stage includes three steps: image intensity adjustment step, clustering step, and post- processing step.

1. Step 1: Image Adjustment Step:

Image correction or adjustment is a popular technique for improving images. The major objective of this step is to execute image intensity level quantification after normalization the finger vein image. In this step the image adjustment maps intensity values in Original image to new values in adjusted image such that values between lower value and high value map to values between 0 and 1 with [0] referring to black and [1] referring to white. The image adjustment is given in equation (2.11), section (2.4.3) in chapter two.

The main idea of modifying the pixel values is to get five levels of pixel values and prepare them for the next step, which is the clustering step, as these five levels represent the central points or centroid to start with the new K-mean or optimized K-mean algorithm. The pixel level in the proposed step of the first model that we are working on is between (0.5-0.9) where each value is a central point ($C_1 = 0.5, C_2 = 0.6, C_3 = 0.7, C_4 = 0.8, C_5 = 0.9$) to achieve the localization process for the finger area and with high efficiency. The following Algorithm (3.2) shows the sequence of the first step in the second stage, which is the clustering step of creating a binary mask to isolate the finger region.

Algorithm (3.2) Image Adjustment Algorithm	
Inputs: I_p : Preprocessed image, $row = 320$, $column = 240$.	
Output: I_{adj} : Adjusted image.	
1	Begin
2	Import the enhanced images that come from Algorithm (3.1)
3	Calculate the low intensity level L_{in}
4	Calculate the high intensity level H_{in}
5	Calculate the length image $length = row \times column$
6	For $i = 1$ to $length$
7	Apply equation (2.11) described in section (2.4.3) in chapter 2 for each pixel in the image.
8	End for
9	New intensity levels L_{out} and H_{out} obtained.
10	Return adjusted image I_{adj} .
11	End.

2. Step 2: Image Clustering Step

In this step, an unsupervised learning approach is used based image clustering. In terms of isolating the finger vein image from its background, an advance image segmentation approach needs to be used here.

The main idea behind the new K-mean optimization is based on using the fixed initial clusters centroid instead on the random initialization ones. To achieve this step, we assume that the intensity distribution is represented as $P(i ; d)$, which i are the intensity values and d is the intensity distribution. In this case, i represents the different intensity levels from different images that are illustrated as $\{I_1, I_2 \dots I_n\}$ in which each pixel has certain intensity value $\{x^{(1)}, x^{(2)} \dots x^{(L)}\}$ where L is denoted as the largest intensity value. In this case, each Cluster center (initial center) is represented as $\{\theta_1, \theta_2 \dots \theta_K\}$ that belong to the suggested fixed number of clusters K . Mainly, each pixel is assigned to each group (cluster center) based on $\{\langle x^{(1)}, U_1 \rangle, \langle x^{(2)}, U_2 \rangle, \dots, \langle x^{(n)}, U_n \rangle\}$ and each center (cluster center) is calculated in this case based on step size that is fixed incrementally increased based on the different of the intensity values .

In this case, instead of using the standard [0-255] band of 256 intensity levels, this method uses the [0-1] range from 10 intensity levels; subsequently reducing the number of degrees to 5 using image adjustment step by shrinking levels to five only. Figure (3.4) illustrates the flowchart of the second step in the second stage (image clustering) in the first model. Algorithm (3.3) describes the essential steps of the finger vein image clustering stage.

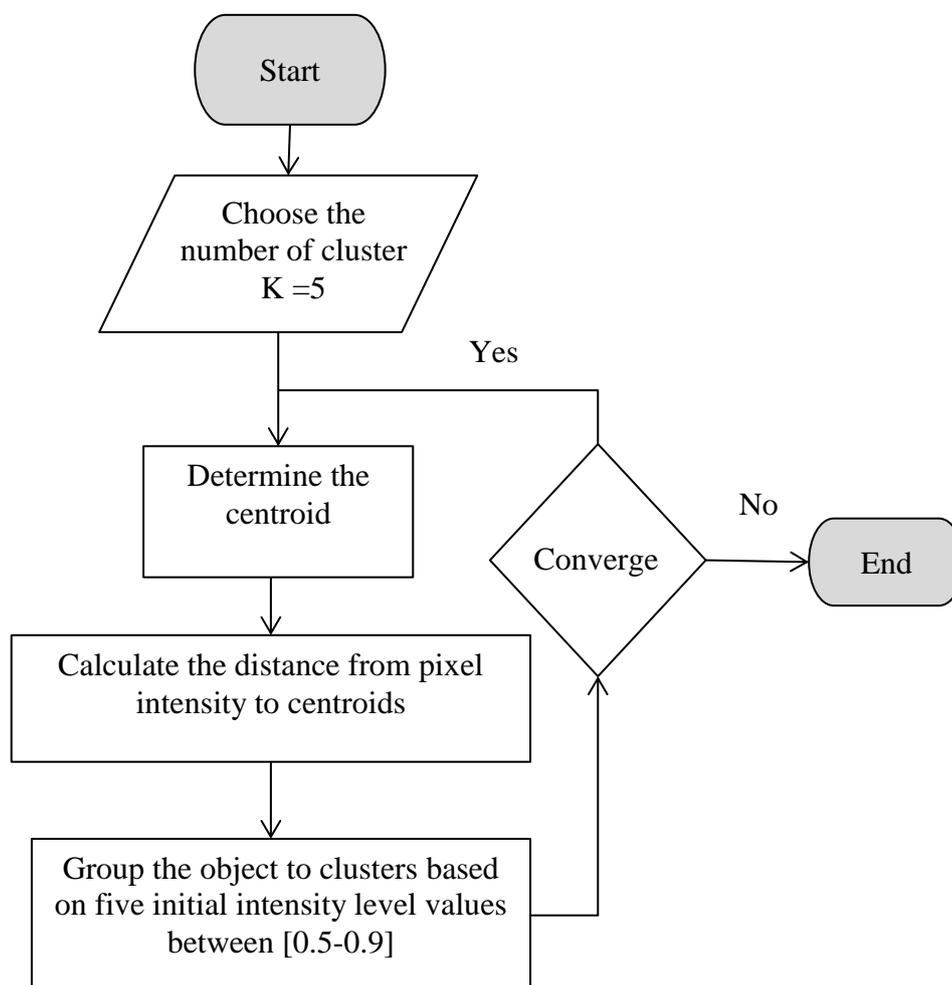


Figure (3.4): Stage 2 Image Clustering Flowchart.

Algorithm (3.3) Clustering Step Algorithm	
Input : Enhanced Image [image dimension is (320 *240)]	
Output : Binary Mask	
1	Begin
2	For loop for each image in the enhanced dataset.
3	Choose number of cluster from 1- 5, $K = 5$, K represent the number of cluster.
4	Initialize each cluster by default value=1
5	Find the global maxima and the global minima values of cluster
6	Determine the initial range space search position
7	Determine the number of steps using the difference between max and min value of centers divided by the number of cluster $K=5$.
8	Incremental value of the clustering steps.
9	Define the center for each cluster.
10	Increment the center by the number of step calculated before.
11	Initialize the update value.
12	Determine mean values for each cluster.
13	Update the means by assign the center value for each cluster.
14	Sample the 1D space image according to the new center.
15	Collect the pixels which are close to each other using the obsolete value of the difference between the current pixel and the current centroid.
16	Collect Cluster 1, Cluster 2, Cluster 3, Cluster 4, and Cluster 5 based on index value.
17	Construct the image clusters by reshaping the five cluster in one cluster (Final Cluster)
18	Binary mask = Final Cluster.
19	End for loop
20	End

The main idea from algorithm (3.3) above is to get a binary mask that isolates the finger area ROI from the background, as we compare the work of this algorithm with other Clustering algorithms.

The difference between the developed new K-mean algorithm and the normal K-meam algorithm is that we used a fixed number of centroid instead of random points, and thus we addressed the problem of instability and randomness that the normal algorithm suffers from, and the other difference is that we used the absolute difference between the value of the current pixel and the value of the center instead of Euclidean distance that the normal algorithm used.

For more clarification, as for example, if the difference between the current pixel and the current center is 1, it means that this pixel will be classified into the first group and if it 2 is classified into the second group and so on. Thus, we find that this method is fixed and not random in classifying pixels on groups.

After distributing the pixels of the image into five groups, in the last step of the algorithm, these groups are combined to obtain a binary mask, where the white color in this mask represents the finger area and the black color represents the background surrounding the finger. By watching the results we obtained, we found that the new optimized K-mean algorithm showed much better results than other clustering algorithms. We will discuss these results in the fourth chapter.

3. Step 3: Post-Processing Step:

The last step of the finger vein localization stage is the post-processing step that is illustrated in Figure (3.5). The irregular forms are built using an efficient clustering method that takes into account the degree to which the intensity values of the pixels in the background and the foreground are comparable. In addition, the final picture of the cluster finger veins is improved by the application of dilation and erosion to get rid of any remaining uneven binary structures. The primary dilatation and erosion of an image equation is given in equation (2.12), section (2.4.4) in chapter two. Algorithm (3.4) shows the steps we have taken to complete the last task in this stage.

Algorithm (3.4) Post Processing Stage Algorithm	
Inputs: I_B : Clustering binary mask image	
Output: I_{NB} : Final Binary Mask Image.	
1	Begin
2	For each (j) image I_B from algorithm (3.3)
3	Import the image I_B // read the image.
4	Specify the $P = 20000$ pixel from the I_B binary mask
5	Get the location of white pixels.
6	Removes all connected components smaller than P .
7	Create a square structuring element with width $W = 11$ pixels.
8	Erode image by structuring element.
9	Dilate image by structuring element.
10	The new binary mask after the morphological operation is I_{NB} .
11	End for j loop
12	End.

After obtaining binary masks for each image in the database, this data or images will be saved in a new folder called the new optimized K-mean algorithm binary masks, Where the post-processing algorithm will be applied in the sequence shown in the algorithm above , where we will read the image from the folder we mentioned earlier and set the value of $P = 20000$ which means removes all connected components (objects) that have fewer than P pixels from the binary image (Binary Mask), producing another binary image. This operation is known as an area opening which is an image processing tool. By experimenting and changing the parameters of these tools, we determined another morphological process, which is that we define a kernel, which is a square-structured element with an angle $W = 11$. Through experimentation, we found that this value gives an optimal shape for the binary mask.

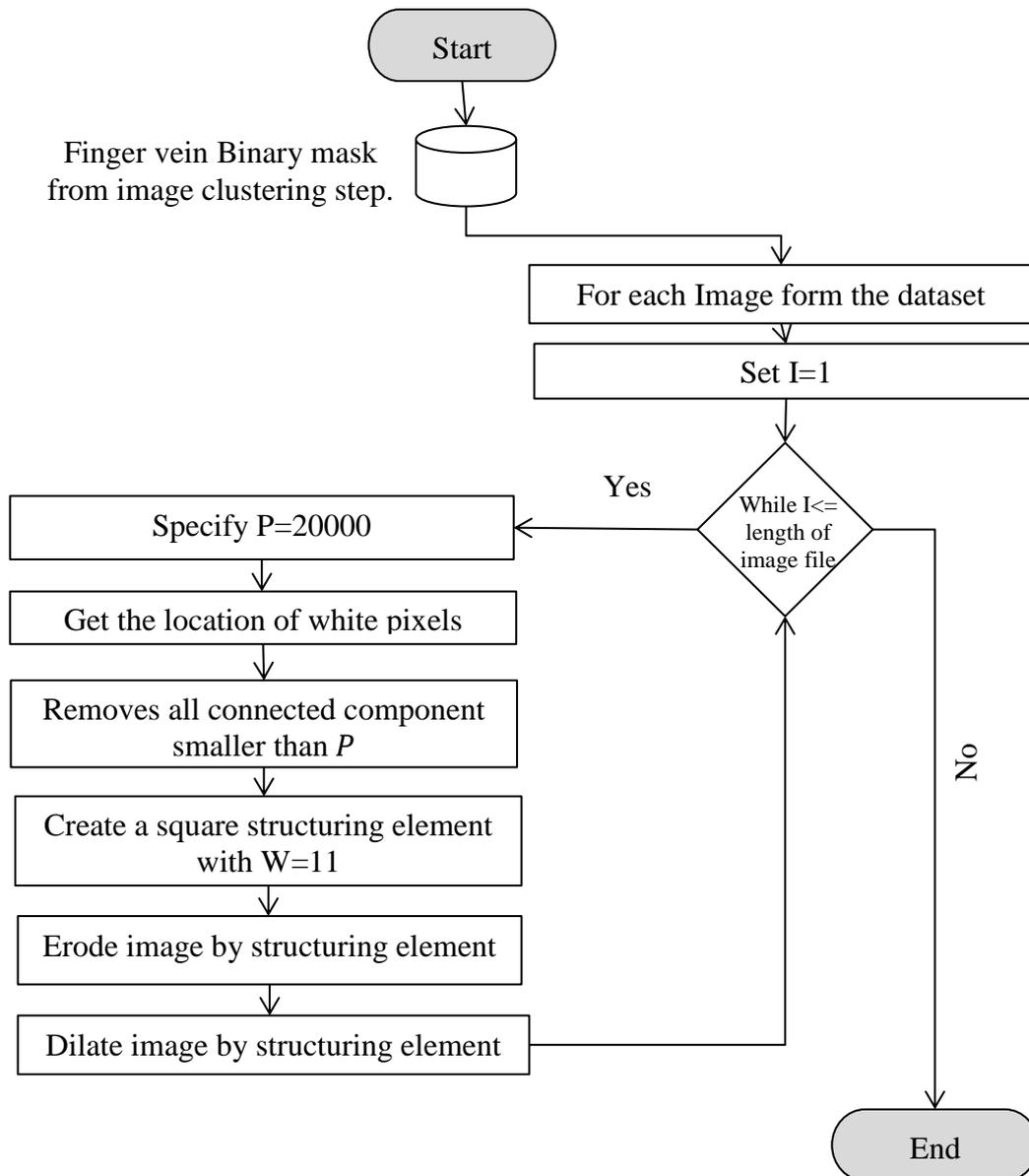


Figure (3.5): Stage 2:Post-processing System Flowchart.

C. Stage 3: Finger Vein Pattern Extraction:

At this stage, we will discuss about extracting the features that we need, which are the hidden vein pattern shape inside the finger area, as we want to get an accurate and clean shape of the finger veins.

There is an approach used previously by researchers, this approach is called "miura_repeated_line_tracking", this method will try to extract the center lines of the veins. This is done by calculating local maximum curvatures in cross-sectional profiles of a vein image. The method is supposed to be robust against varying vein widths and non-uniform brightness.

Repeated line tracking: This algorithm will start at several random points and will try to track a line. If a pixel has been visited by the line tracking algorithm multiple times it is likely to be a vein [73].

The problem with this approach is that the shape of the finger vein pattern is distorted and has many separate lines, knowing that the shape of the image resulting from this approach is the form of a binary image, so until we address the problem of distortion in the lines of the pattern Finger veins, the usual method is using a morphological operation, but the problem here is that each line of curvature is different from the other line of the same pattern and each time there is a different angle. When applying the morphological operation, it is necessary to determine the appropriate angle to get rid of the distortion in the image, so we will get a more distorted shape if we use this method. Based on what we have mentioned, the idea here is how to get the direction and paths of these lines that represent the pattern of the finger vein; this will be explained by the GPE algorithm in the next part.

There are three steps in this stage. The first step is a previous approach that we touched upon and referred to its source in the previous two sections, so we will start from the second step. The third step we will talk about in the second step.

1. Step 2: Global Pattern Extraction (GPE)

After normalizing the images, an estimation of the finger vein lines directions is computed that is based on specific finger vein characteristics such as the quality and the image intensity.

Finger vein images are represented by the orientation image, which reflects their properties. Ridges and furrows invariant coordinates are specified by the localized feature orientation. Algorithm (3.5) displays the major phases of the technique of estimating the feature orientation.

Algorithm (3.5) Global Map Extraction Algorithm**Inputs:** I_L : Represent the Localized Finger Vein Image.**Output:** I_{orient} : Position of Local Ridges

- 1 **Begin**
- 2 **For each (i) image I_L from the algorithm (3.4).**
- 3 Import the image I_L // read the image with dimensions 240×320 .
- 4 Images were divided into $w \times w$ blocks from the normalized image
- 5 Each block will have its dimensions 23×31
- 6 Smoothed the finger vein image using Gaussian filter... with standard deviation σ choose from [3-31]
- 7 Compute gradient along x-axis and y-axis
- 8 Compute the first and second derivative along x and y axis
- 9 Smoothed the first and second derivative along x and y axis using Gaussian filter... with standard deviation σ choose from [3-31].
- 9 Determine theta value in degree and then convert to radian.
- 10 Specify line length in each block
- 11 Determine placement of orientation vectors
- 12 Return the angel of each block Which represents the grouping of pixels in the same direction
- 13 Return I_{orient} Position of Local Ridges
- 14 **End**

This is an example of all the steps that we touched on in the first model, which are shown in the following Figure (3.6).

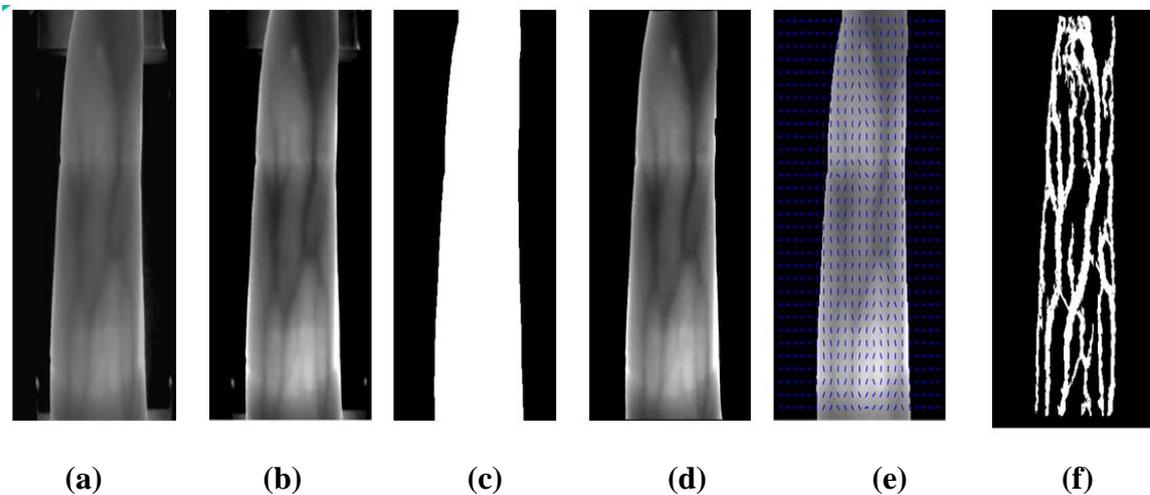


Figure (3.6): All States from the First Model

As (a) is Original finger vein image, (b) is Enhanced image, (c) is Binary mask from new k-mean algorithm, (d) is Localized Finger Vein Image, (e) is GPE Image, (f) is Final Optimized Pattern GPO.

3.4.2. Second Model: Fully Automated Finger Vein Pattern Extraction based Deep Semantic Segmentation.

The main idea of using the second model is first to do another optimization process using deep learning approach and secondly to address some cases where the first model was not able to get a good shape for the finger patterns.

In this approach, first, we proposed a fully automated deep learning model using dataset that is automatically generated by using first model (Fully Automated Finger Vein Binary Pattern Extraction based Unsupervised Learning Approach). These data are images of the finger vein pattern that we extracted for each image in the database so that we train the model to recognize the identity of people through the vein pattern and not the image as a whole because the basis on which the comparison process was built to find out the identity of a person is the vein pattern that represents the unique characteristic that distinguishes each One person over another instead of comparing the image as a whole.

Second, propose a non-traditional deep learning model for finger vein identification. The proposed deep semantic segmentation model based Unlabeled Positive loss function extracts the binary finger vein lines and identify the test the original finger vein images based on the similarity between those patterns (finger vein lines). The general flowchart of the second model is illustrated in Figure (3.7).

3.4.2.1 Deep Fully Automated Finger Vein Pattern Extraction Designed Model.

Deep semantic segmentation of finger vein patterns using CNNs and PU learning is presented and illustrated in Figure (3.8). The second model (Fully Automated Finger Vein Pattern Extraction based Deep Semantic Segmentation.) Evaluate the binary patterns automatically by selecting it for

good training datasets while the others are isolated to be a testing dataset. Then, the fully automated training dataset are labeled based on the pixels coordinated to be positive/negative samples as is shown in Figure (3.8) (a).

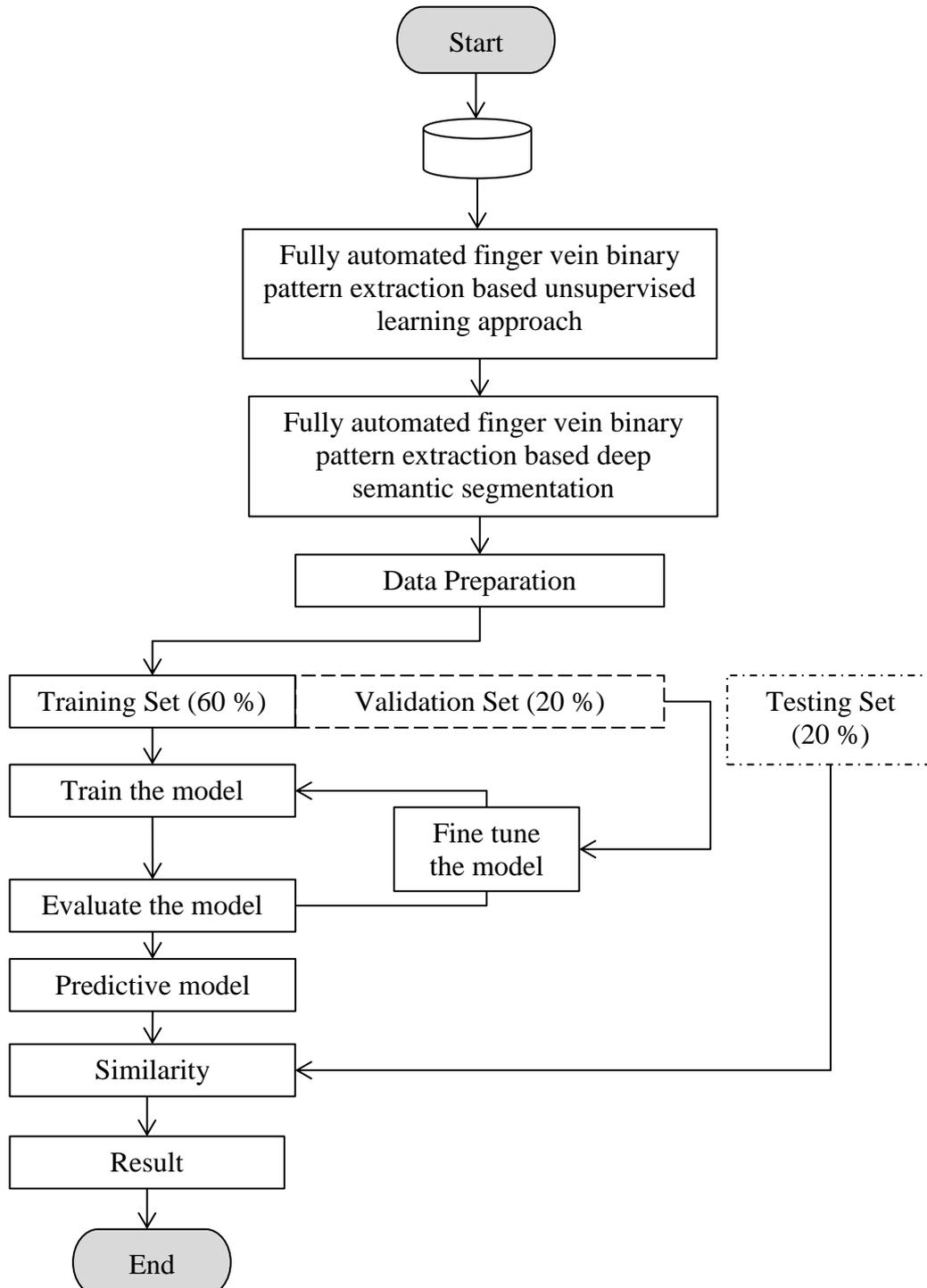


Figure (3.7): Fully Automated Finger Vein Pattern Extraction based Deep Semantic Segmentation Flowchart

train the input images. The numbered algorithm (3.6) shows the steps of the second model:

Algorithm (3.6) Semantic Segmentation Algorithm	
INPUT: 1. Finger Vein Pattern ant its image 2. Fully labeled output for supervised training	
Output: Predicted binary mask	
1	Begin
2	Install All Libraries Required For Training dataset
3	All images have 320 pixels height and 240 pixels width
4	During the Training the dimension will be 128×128 in the first convolution
5	load repo with data if it is not exists
6	Use helper function for data visualization
7	Read images, apply augmentation and preprocessing transformations.
8	Convert string names to class values on masks
9	Extract certain classes from mask
9	Apply augmentations
10	Apply preprocessing
11	Create segmentation model with pre trained encoder
12	Prepare Hyper parameter before training ,Optimizer="Adam Optimizer" ,Learning Rate="0.0001",Number of epochs="25",Batch size ="32"
13	Test best saved model
14	Load best saved checkpoint
15	Create test dataset
16	Evaluate model on test set
17	Visualize predictions , test dataset without transformations for image visualization
18	Return the predicted mask
19	End

3.4.2.2 Classifier training based positive-unlabeled data:

As is shows in Figure (3.8) (b), in this model, a pre-trained approach was used based on a new architecture in the (CNN) convolutional neural network, which is called U-NET. This approach consists of two parts, the first is called the encoder and the second is called the decoder. Before explaining the processes involved in the deep learning model, I would like to point out a set of terms.

1. 2@Conv layers means that two consecutive Convolution Layers are applied.
2. C_1, C_2, \dots, C_N are the output tensors of Convolutional Layers.
3. U_1, U_2, \dots, U_N are the output tensors of up-sampling (transposed convolutional) layers

4. In the Encoder, the size of the image gradually reduces while the depth gradually increases. Starting from $128 \times 128 \times 3$ to $8 \times 8 \times 256$.
5. In the decoder, the size of the image gradually increases and the depth gradually decreases. Starting from $8 \times 8 \times 256$ to $128 \times 128 \times 3$.

A. Contraction Path (Encoder):

The size of the input image is $320 \times 240 \times 3$ however, we will use input image with size of $128 \times 128 \times 3$. At the first convolution we have 2@Conv and 16@ 3×3 filters and Padding from the type 'same'. After the process of multiplying the entered image by the number of existing filters, we will get a new output C_1 after completing the first two layers. And now the dimensions of this process have become $128 \times 128 \times 16$.

Now we apply the max pooling layer to the tensor C_1 with block size of 2×2 and Stride =2 which mean which means moving two pixels Since the purpose of the pooling operation is to make a down sampling.

As the tensor that resulted from the pooling operation, we called it P_1 for clarification only. After this operation, the dimensions of the image became $64 \times 64 \times 16$ and the same operations will be repeated on the tensor P_1 .

At the Second convolution we have 2@Conv and 32@ 3×3 filters and Padding from the type 'same'. After the process of multiplying P_1 image by the number of existing filters, we will get a new output C_2 after completing the second two layers. And now the dimensions of this process have become $64 \times 64 \times 32$. Now again we apply the max pooling layer to the tensor C_2 with block size of 2×2 and Stride =2 and the tensor that resulted from the second pooling operation, it is called P_2 after this operation, the dimensions of the image became $32 \times 32 \times 32$ and the same operations will be repeated on the tensor P_2 .

At the third convolution we have 2@Conv and 64@ 3×3 filters and Padding from the type 'same'. After the process of multiplying P_2 image by

the number of existing filters, we will get a new output C_3 after completing the third two layers. And now the dimensions of this process have become $32 \times 32 \times 64$. Now again we apply the max pooling layer to the tensor C_3 with block size of 2×2 and Stride =2 and the tensor that resulted from the third pooling operation, it is called P_3 after this operation, the dimensions of the image became $16 \times 16 \times 64$ and the same operations will be repeated on the tensor P_3 .

At the fourth convolution we have 2@Conv and 128@ 3×3 filters and Padding from the type 'same'. After the process of multiplying P_3 image by the number of existing filters, we will get a new output C_4 after completing the fourth two layers. And now the dimensions of this process have become $16 \times 16 \times 128$. Now again we apply the max pooling layer to the tensor C_4 with block size of 2×2 and Stride =2 and the tensor that resulted from the fourth pooling operation, it is called P_4 after this operation, the dimension of the image became $8 \times 8 \times 128$ and the same operations will be repeated on the tensor P_4 .

At the fifth convolution we have 2@Conv and 256@ 3×3 filters and Padding from the type 'same'. After the process of multiplying P_4 image by the number of existing filters, we will get a new output C_5 after completing the fifth two layers. And now the dimensions of this process have become $8 \times 8 \times 256$. Now again we apply the max pooling layer to the tensor C_5 with block size of 2×2 and Stride =2 and the tensor that resulted from the fifth pooling operation, it is called P_5 after this operation, the dimension of the image became $8 \times 8 \times 256$, as the down sampling process will end in the encoder part and the image will go to (FCN) Fully Connected Layer $1 \times N$. As the number of layers increases, we repeat the same operations until we reach the middle layer (FCN), which prepares the image for us to work on up sampling operation in decoder part.

B. Expansion Path (Decoder):

The sequence of operations in this part is in reverse of what we did in the previous part (Encoder). For Up sampling Transposed Convolution layer are used, and the parameters for each Transposed Convolution layer are such that the height and width of the image are doubled while the depth (number of channels) is halved. I will only touch on one process, and the rest of the operations in subsequent layers will be of the same principle.

After (FCN) $1 \times N$ vectors up sampling will begin where the dimensions of C_5 were $8 \times 8 \times 256$ after adding a skip connection from the opposite layer in the encoder part of the bottom of the U-net Structure. To clarify this point the U_6 for example will be come from $U_6 = U_6 + C_4$, and so on for the rest of the layers in the decoder part until we reach the dimensions of the original image and get a dimension $N \times 1$ So the pixels are categorized. Note that for each pixel we get a value between 0 and 1, where 0 represents negative and 1 represents Positive which mean the activation function used at output layer is sigmoid function. We take 0.5 as the threshold to decide whether to classify a pixel as 0 or 1.

The following table (3.1) shows a summary of what happened in the layers of the encoder model used:

Table (3.1): Encoder Operation

Encoder					
Input Image	Convolution	Filter size	Padding	Max pooling	Output image
$128 \times 128 \times 3$	2 Convolution	$16 @ 3 \times 3$	same	2 Stride	$64 \times 64 \times 16$
$64 \times 64 \times 16$	2 Convolution	$32 @ 3 \times 3$	same	2 Stride	$32 \times 32 \times 32$
$32 \times 32 \times 32$	2 Convolution	$64 @ 3 \times 3$	same	2 Stride	$16 \times 16 \times 64$
$16 \times 16 \times 64$	2 Convolution	$128 @ 3 \times 3$	same	2 Stride	$8 \times 8 \times 128$
$8 \times 8 \times 128$	2 Convolution	$256 @ 3 \times 3$	same	2 Stride	FCN and Up sampling

The following table (3.2) shows a summary of what happened in the layers of the decoder model used:

Table (3.2): Decoder Operation

Decoder					
Input Image	Convolution	Filter size	Padding	Max pooling	Output image
FCN and Up sampling	2 Convolution	256@3 ×3	same	2 Stride	8×8 ×128
8×8 ×128	2 Convolution	128@3 ×3	same	2 Stride	16×16 ×64
16×16 ×64	2 Convolution	64@3 ×3	same	2 Stride	32×32 ×32
32×32 ×32	2 Convolution	32@3 ×3	same	2 Stride	64 ×64 ×16
64 ×64 ×16	2 Convolution	16@3 ×3	same	2 Stride	128 ×128 ×3

3.4.2.3 Pattern Extraction Based Loss Function:

It is possible to retrieve finger vein pixel coordinates and probabilities from a trained CNN classifier. We begin by applying the classifier to each finger vein image pixel-based area and calculating the per pixel predicted probability. Iteratively selecting high score pixels and removing their neighbor from concept as finger vein center is then used to generate dense predictions. The purpose of the training process is to predict the shape of the pattern of each person in the database, especially those who did not obtain a clear pattern in the first model, as the loss function used here, which improves the values of weights during the training process, is the PU loss binomial this function shown in the section (2.9.1) and the equation number is (2.16) in chapter two, whose work is similar to the Cross Entropy Function. As the difference here is that in the Cross Entropy Function, both classes are labeled, but in binomial PU loss function not only one of the classes is labeled from the first model automatically, while the second class is the probability of the first class subtracted from one. The probability will be calculated at the pixel level, so this process is called dense prediction. And that the segmentation process will be of high quality because the work was at the pixel level in calculating the value of the error during the training process.

Chapter Four

Experimental Results

4.1 Introduction

In this chapter, all the experimental results will be discussed and explained for both two models.

Three stages of the first proposed model "**Fully Automated Finger Vein Binary Pattern Extraction based Unsupervised Learning Approach**" are conducted. The first stage is the pre-processing stage where the image is enhanced. The second stage is the finger vein image localization stage where each image has gone through different steps such as intensity adjustment, clustering, and post processing step to get the localized finger vein image. The last stage is the binary pattern extraction stage that is based on different steps such as GPE and GPO.

There are two kinds of result for the second proposed model "**Fully Automated Finger Vein Pattern Extraction based Deep Semantic Segmentation**" that depends on using the generated binary mask from the first proposed model.

The performance of the first proposed model based on the experiential results is evaluated using different evaluation criteria such in each stage. For instance, in the preprocessing stage the enhanced finger vein images are evaluated based on calculating different quantitative criteria such as Mean Square Error (MSE), Signal to Noise Ratio (SNR), and Peak Signal to Noise Ratio (PSNR). Also, the performance results for the second and third stage are evaluating and calculating. The performance results are evaluated using the confusion matrix to show the accuracy measure between the ground truth mask and the predicated one from deep semantic model.

The performance result of the deep semantic segmentation model for fully automated finger vein mask prediction evaluated on both training and

testing phase. The performance results are evaluated using the confusion matrix to show the accuracy measure between the ground truth mask and the predicated one from deep semantic model.

4.2 Implementation Environment

The physical studying of a thesis is divided in two parts the first part that represent the pre-processing is implemented using MATLAB R2019b programming language. The testing is also implemented under the environment of Windows-10 operating system, laptop computer processor: Core™ i5-7200U, CPU @2.50GHz (4CPUs), RAM 8.00 GHz. The second part of the physical studying that represents the training of our datasets is implemented using python 3.9 using the spider navigator under the environment of Linux Ubuntu operating system: Acer Nitro 5 gaming Laptop 9th Generation, Intel Core i7 – 9300H, NVIDIA GeForce GTX 1650, 15.6 Inch FHD IPS, 8GB, 8GB shared and NVME SSD, Backlit Keyboard.

4.3 Dataset

During this study, finger vein database SDUMLA-HMT was employed. Shandong University in Jinan, China collected SDUMLA-HMT in the summer of 2010. 106 participants, 61 of whom were men and 45 of whom were women, ranged in age from 17 to 31. Three types of images were taken for each participant. The first image is called Index, the second is called Middle, and the third is called Ring. These three types are for both the right and left hands, where 6 shots were taken for each type of finger. The three types that we mentioned, and therefore the total number of the database will be $106 \times 6 \times 6$, which equals 3,816 images, and these images were stored in Bmp format. Figure 4.1 show some samples of finger vein dataset, "*Finger vein data base*" is created in the system directory in our computer system to store the images [22].

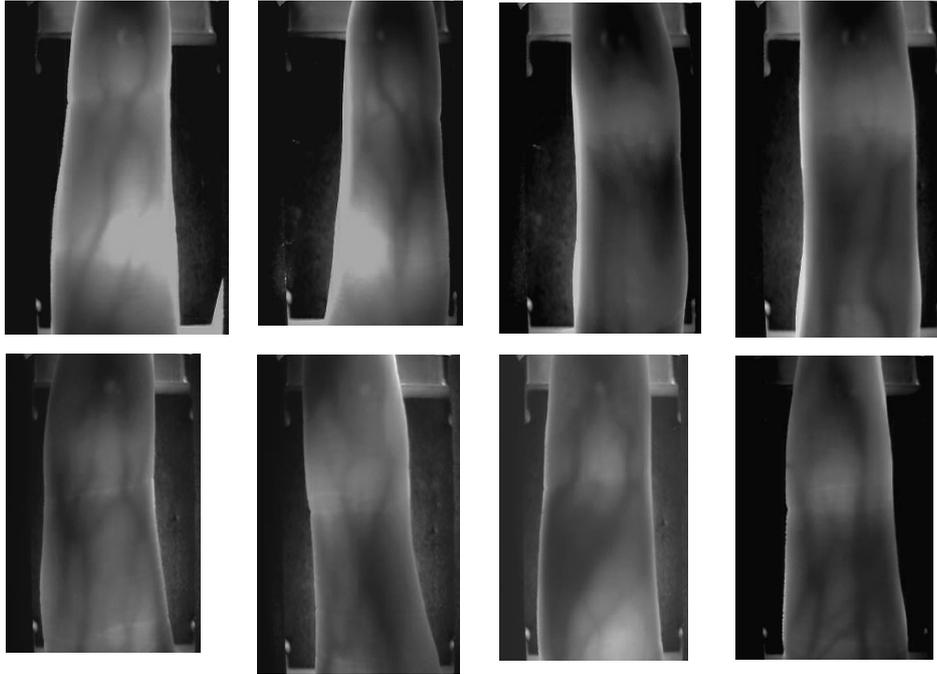


Figure (4.1): Samples of the Finger Vein Dataset [22].

4.4 Experiential Results of the Preprocessing Stage:

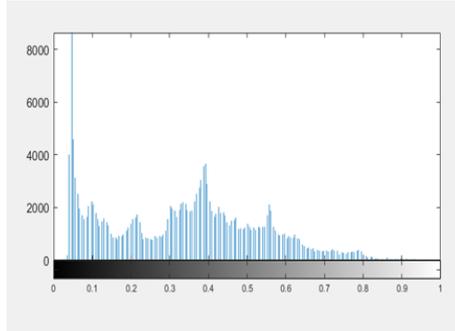
In this stage all the experimental results that have been drawn and achieved through the pre-processing stage from the first model are described, illustrated, and discussed.

4.4.1 Subjective Experimental Results of Preprocessing Stage

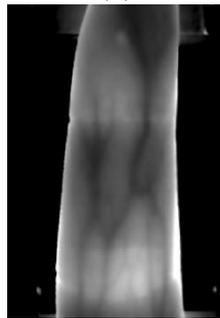
In the beginning, in order for the biometric identification system to be a good system, the data that we are working on, which is images, as shown in figure (4.1) must be of good quality, so because they are of low quality, we should make some improvements to these images and display Its results are to know the improved images, as the benefit of the preprocessing process is to prepare the data to extract the patterns we need from the finger area ROI. The process of improving the image went through several stages as mentioned in the previous chapter, where more than one technique was used to obtain an improved image. We will show in Figure (4.2) samples of the image results before and after the Enhancement process.



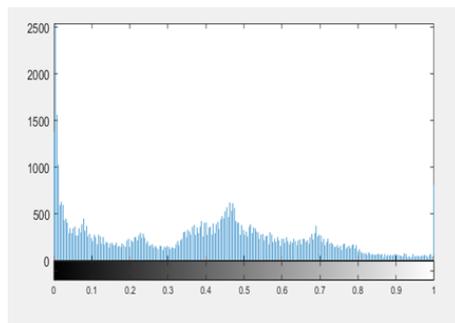
(a)



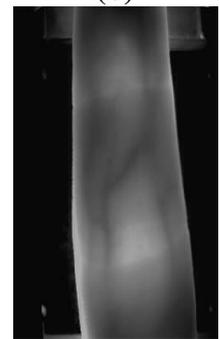
(e)



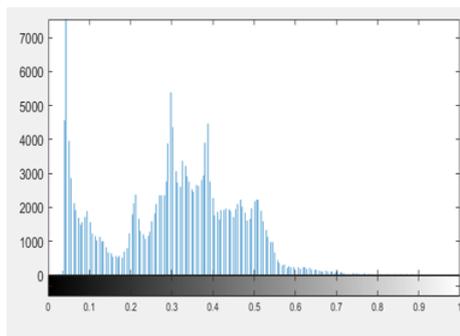
(b)



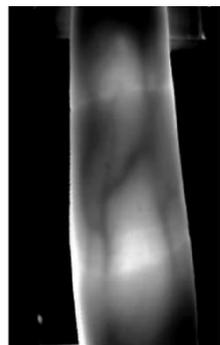
(f)



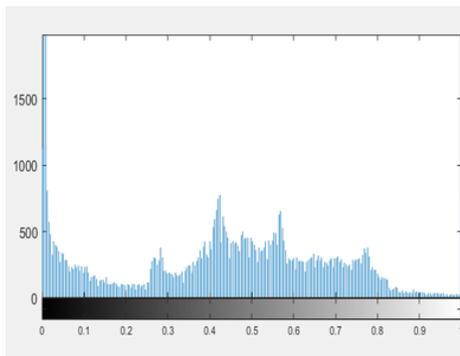
(c)



(g)



(d)



(h)

Figure (4.2): Experimental Result for Pre-processing Stage.

As image (a) and image (c) represent the original image and the corresponding ones in field (e) and field (g) represent the histogram of these original images. In the same context as the description of the figure, the images (b) and (d) represent the image after performing the enhancement process, and the corresponding in the field (f) and (h) where they represent the histogram of the improved images.

The enhanced finger vein images histograms that are shown in figure (4.2) showing the different enhanced level of the image intensity comparing with the original ones.

Where we notice through the Subjective results in Figure (4.2) that the shape of the veins has become darker than it was in the original image, and this is the point of conducting the enhancement process to obtain the important part and data from the image using Weiner filter and using the process of converting the image to the LAB color space system, where the lighting part was isolated from the color part, the work was related to the color part.

4.4.2 Quantitative Experimental Results of Preprocessing Stage

In term of measure or experiential results objectively, different quality metrics are calculated to extract the amount of information that has been gained after processed and enhanced each image.

To calculate the MSE, the mean and standard deviation of the original image are calculated, where we subtract the standard deviation from the mean we calculated. In the same way, we calculate the MSE of the enhanced image.

As for the SNR, this value is calculated by taking the logarithm of the base 10 for the signal-to-noise ratio multiplied by 10. In the same way, we calculate the SNR of the enhanced image.

As for the PSNR, this value is calculated by taking the logarithm of the base 10 for the image dimension ($M \times N$) divided on MSE multiplied by 10. In the same way, we calculate the PSNR of the enhanced image.

Table (4.1) shows some objective quality assessment for each image from the dataset. Random images are chosen from the dataset to show the result of quality metrics.

Table (4.1): Objective Quality Assessment of Processing Stage.

Image	Preprocessed Image		
	PSNR	MSE	SNR
1	56.17213	0.185417	5.383724
2	58.10306	0.059574	1.917117
3	55.24455	0.057664	2.038335
4	54.11828	0.059365	2.048946
5	53.26693	0.090922	2.875347
6	56.22729	0.115513	3.548883
7	54.70493	0.082199	2.963809
8	56.24177	0.09145	4.023359
9	58.00594	0.096553	3.975529
10	64.88765	0.015725	0.700419

In Figure (4.3), we notice that the MSE of the processed/Enhanced image (that is shown in the red curve) are improved and reduced among the original images comparing with the original MSE (that is shown the blue curve).

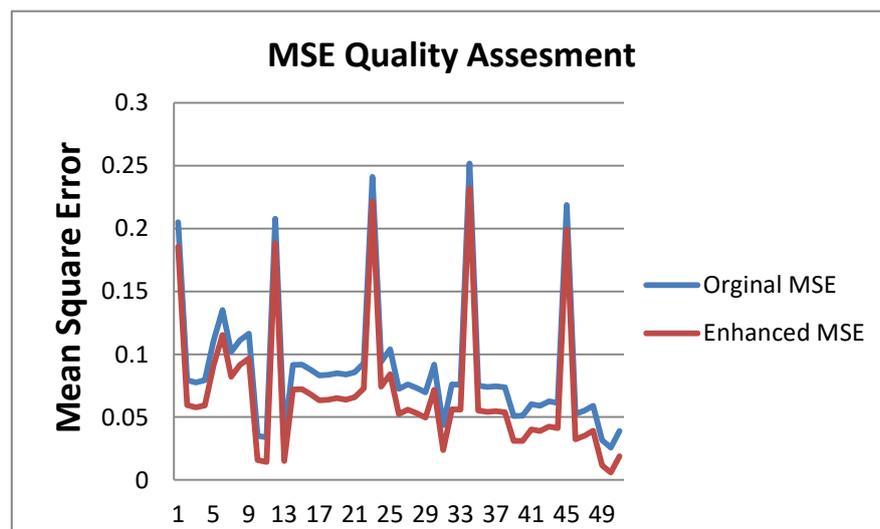


Figure (4.3): Mean square error (MSE) of the original finger vein images and for enhanced images.

In both Figures (4.4), and Figure (4.5), we notice that both PSNR and SNR for the processed and enhanced images (that is shown in the red curve) are improved and increased among the PSNR and SNR values of the original images (that is shown the blue curve) which gives us an indication about how much improving we gain by using the proposed system enhancement approach.

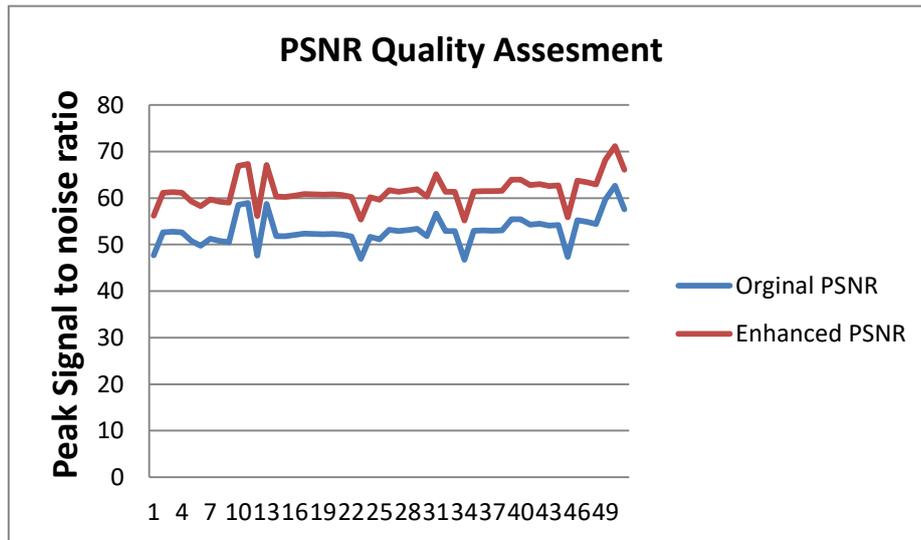


Figure (4.4): Peak-Signal-to-Noise-Ratio (PSNR) of the Original and Enhanced images.

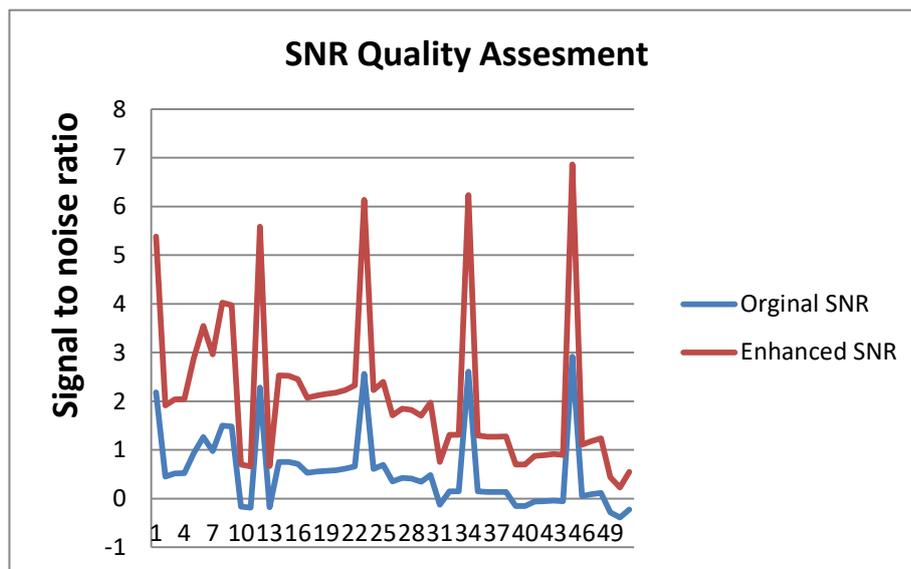


Figure (4.5): Signal-to-Noise-Ratio (SNR) of the Original and Enhanced image.

Where the x-coordinate represents the number of random images, the y-coordinate represents the values of the different parameters (MSE, SNR, and PSNR) in each form separately.

4.5 Experiential Results of Finger Vein Localization Stage:

In this part, we will talk about the experimental results of the second stage of the first model, which is stage (**Finger Vein Localization Stage**) as the first step (**Image Adjustment**) has been combined with the results of the previous stage (**Preprocessing Stage**). As for the second step (**Image Clustering**) and the third step (Post processing) of this stage, we will talk about its results in detail.

4.5.1 Subjective Experimental Results of Image Clustering.

This step (**Image Clustering**) is considered the backbone of the second stage (**Finger Vein Localization Stage**) of the first proposed model. The goal of this step is to obtain a binary mask to isolate the finger area from the background. The new K-mean algorithm was compared with other collection clustering algorithms, where approximately 100 images were selected from a database and a process was performed Label these 100 images to get a mask called (Ground Truth) The purpose of this process that we conducted is to evaluate the different algorithms that were used in this system, namely, the new K-mean algorithm, which we are working on in this thesis, FCM (Fuzzy C-mean) algorithm, the standard K-mean algorithm, and the Out's clustering algorithm, as each algorithm During its implementation, we will get different forms of binary masks. The comparison will be at the pixel level between the ground truth image and the binary mask images of the different algorithms. The following figure (4.6) shows the shape of the binary-masks resulting from the other algorithms.

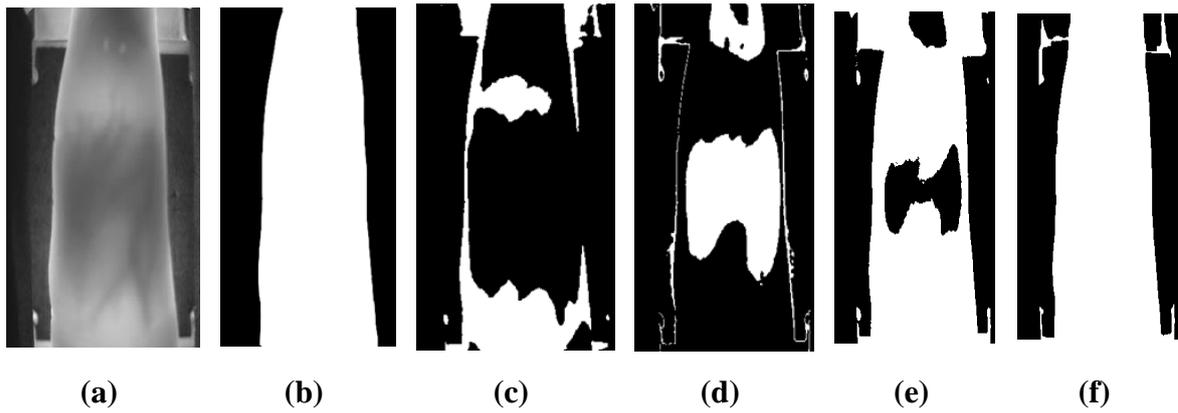
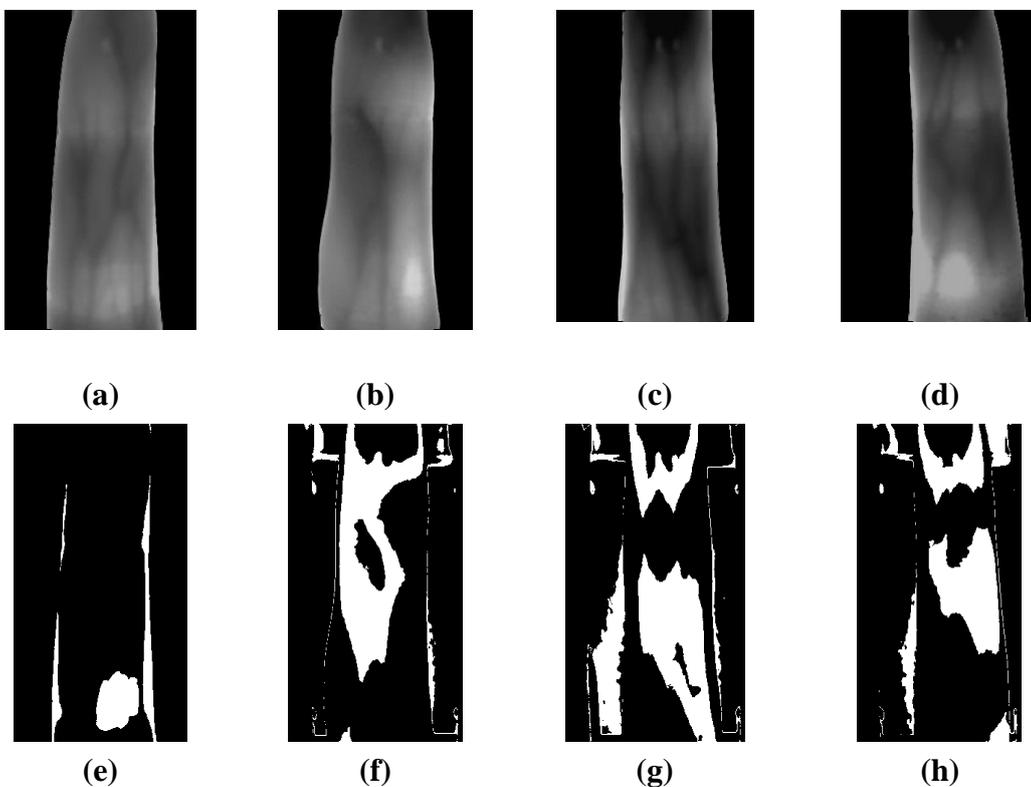


Figure (4.6): Different Binary Masks from Algorithms Used.

Where (a) represents the original image, (b) represents the ground truth and (c) represents the binary mask output from (FCM) algorithm, (d) represents the binary mask output from (K-mean) algorithm, and (e) represents the binary mask output from (Out's) algorithm, and (f) represents the binary mask output from (new K-mean) algorithm . Therefore, we notice from the figure (4.6) that the binary mask of new K-mean algorithm is the closest to ground truth.



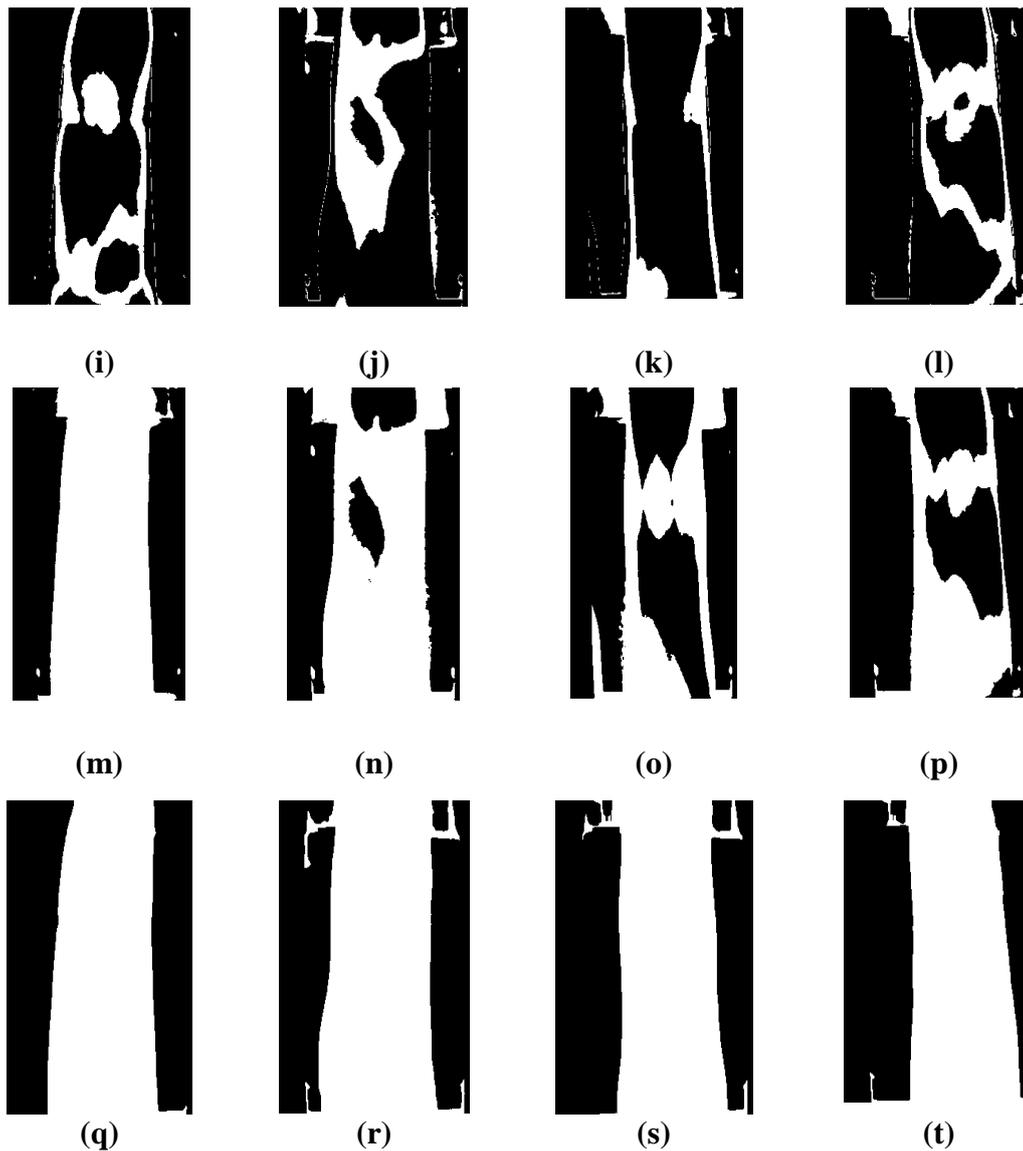


Figure (4.7): Different Binary Mask images (a)-(d) are original images, (e)-(h) are FCM Masks, (i)-(l) are standard K-mean Masks, (m)-(p) are Out's Masks, (q)-(t) are new K-mean Masks.

We notice that all the testing cases have been correctly detected and the finger image is extracted from the background correctly based new K-mean optimized model that is proposed in this approach. In contrast, Figures (4.7), show the same cases that have been tested in the model using the original k-means, FCM, and Out's algorithms, that are shown that all the cases have been failed to extract the finger object from its image correctly. New k-mean will get the finger area and isolate it from the background as there are some edges on top and bottom of the finger area. This will be solved by using some

post processing tools using Equation (2.12) in section (2.4.4) in chapter two. Figure (4.8) shows the shape of the binary masks after applying the post-processing tools.

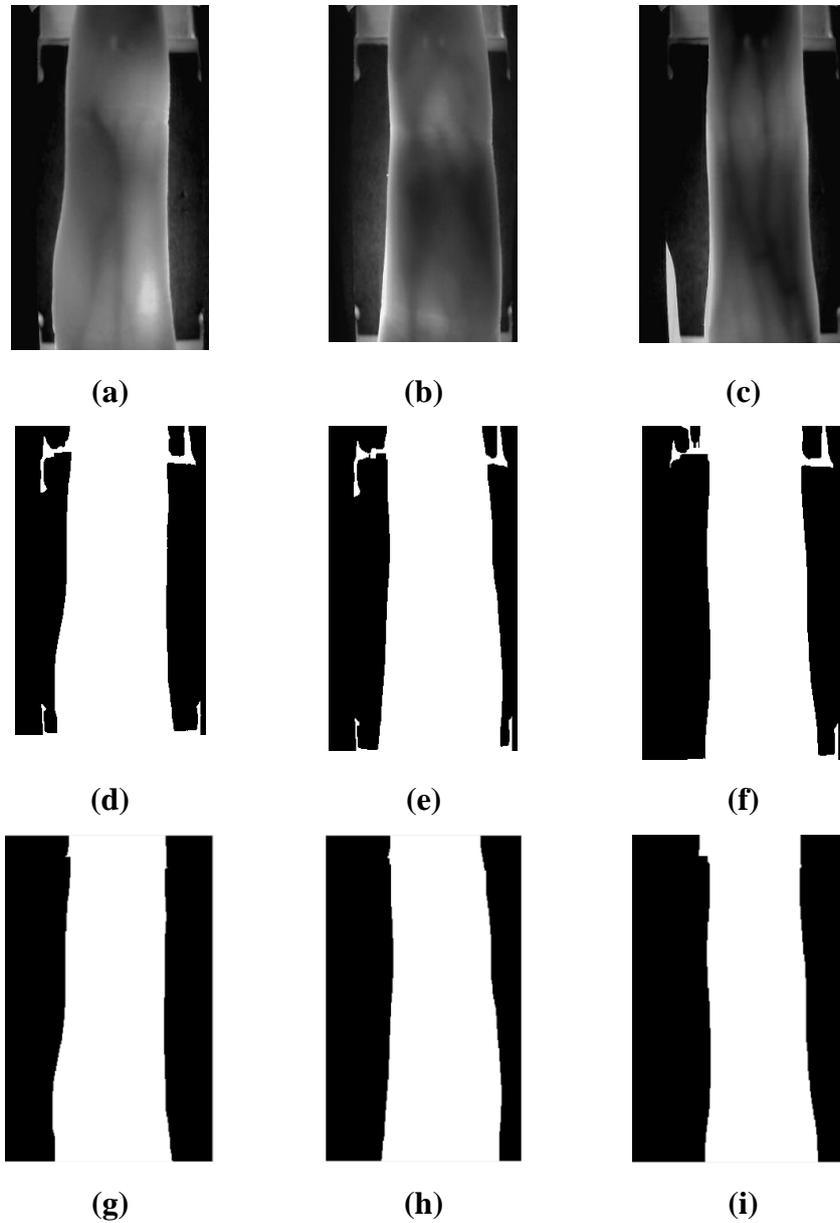


Figure (4.8): Experimental Result for Post-processing Step.

Where the images from (a) to (c) represent the original image and from (d) to (f) represent the new K-mean binary mask and from (g) to (i) represent the final binary mask after post processing tools applying.

4.5.2 Quantitative Experimental Results of Image Clustering.

After looking at the shape of the binary masks generated by the different algorithms used (FCM), (K-mean), (Out's), and (new K-mean optimized), we now come to evaluate the performance of these algorithms using the equations (2.21), (2.22), (2.23) and (2.24), section (2.10.2) in chapter two.

The method of evaluating the results numerically will be as follows: First, the binary masks resulting from the different algorithms are kept in separate folders, in addition to the Ground Truth in a separate folder as well. To ensure that each image matches its equivalent in the Ground Truth folder, the random images to be tested have been renamed to sequential numbers. To evaluate the performance of each algorithm, we read the images from the respective folders and make a comparison at the pixel level, meaning that the different parameters (TP), (TN), (FP), and (FN) that belongs to an (confusion matrix) for evaluating the model will pass different conditions:

The first condition, if the value of the pixel in the Ground Truth is (1) and in the mask of the algorithm to be evaluated is (1), then the value of (TP) will be incremented. The second condition, if the value of the pixel in the Ground Truth is (1) and in the mask of the algorithm to be evaluated is (0), then the value of (FP) will be incremented. The third condition, if the value of the pixel in the Ground Truth is (0) and in the mask of the algorithm to be evaluated is (1), then the value of (FN) will be incremented. The fourth condition, if the value of the pixel in the Ground Truth is (0) and in the mask of the algorithm to be evaluated is (0), then the value of (TN) will be incremented. Then, the Accuracy, Precision, Recall, and F1-measure are calculated. Table (4.2) shows that the new optimized K-mean algorithm achieved a better accuracy (98.25%) compared to the other unsupervised clustering algorithms. Moreover, comparing with the other metrics, new K-mean approach has achieved better Precision, Recall, and F1-measure by (91.5%), (99.5%), and (94.4%) respectively.

Table (4.2): Finger Vein Image Clustering Results Using 10 Samples Randomly Selected For Different Clustering Algorithm.

New optimized K-mean Algorithm				
Samples	Accuracy	Precision	Recall	F1-measure
1	97.8034	96.8951	99.3931	98.1282
2	99.4531	99.3513	99.6156	99.4833
3	99.3086	99.1178	99.6476	99.382
4	99.2656	99.1208	99.5689	99.3443
5	97.4635	95.8217	99.6841	97.7147
6	98.5677	97.6078	99.7847	98.6843
7	98.4492	97.5636	99.8084	98.6733
8	98.1393	97.0389	99.8146	98.4072
9	98.0065	96.8735	99.7484	98.2899
10	97.9466	96.1457	99.8141	97.9456
FCM Algorithm				
Samples	Samples	Samples	Samples	Samples
1	4.0378	0.0421	0.0303	0.0353
2	52.681	80.5259	20.1971	32.2944
3	44.0391	44.6054	11.8648	18.7439
4	37.1797	28.407	10.9867	15.8451
5	46.2917	65.4142	14.9479	24.335
6	49.5951	78.8994	17.0198	27.9996
7	45.9818	95.8727	6.2142	11.6718
8	58.7331	83.3277	19.8131	32.0141
9	51.3724	57.3116	3.7316	7.007
10	17.8385	9.6131	8.0766	8.7781
K-mean Algorithm				
Samples	Accuracy	Precision	Recall	F1-measure
1	60.0195	90.131	34.7943	50.2068
2	64.7917	99.1368	33.6725	50.2703
3	52.0065	79.4566	18.8404	30.4587
4	51.9714	79.7023	18.8409	30.4772
5	60.526	87.2441	32.1358	46.9703
6	62.8971	86.4281	36.8617	51.6813
7	57.987	89.8028	30.7815	45.8479
8	52.5013	68.3138	32.667	44.1987
9	48.5404	76.7991	14.904	24.9635
10	58.4167	83.3994	18.982	30.9253
Otsu Algorithm				
Samples	Accuracy	Precision	Recall	f-measure
1	84.9362	89.9866	83.2614	86.4935
2	97.5859	95.6338	99.9975	97.767
3	95.3867	96.0102	95.7077	95.8587
4	95.7253	96.2426	96.1013	96.1719
5	86.6276	89.4523	85.4999	87.4315
6	86.875	89.1654	86.0768	87.5938
7	89.1432	91.6428	89.3586	90.4863
8	72.5781	84.5695	64.0701	72.9062
9	60.1497	78.3174	42.336	54.9615
10	70.3815	80.9967	51.7418	63.1454

We can notice from Figure (4.9) that new optimized K-mean Algorithm has achieved better Accuracy, Precision, Recall and F1-measure as well.

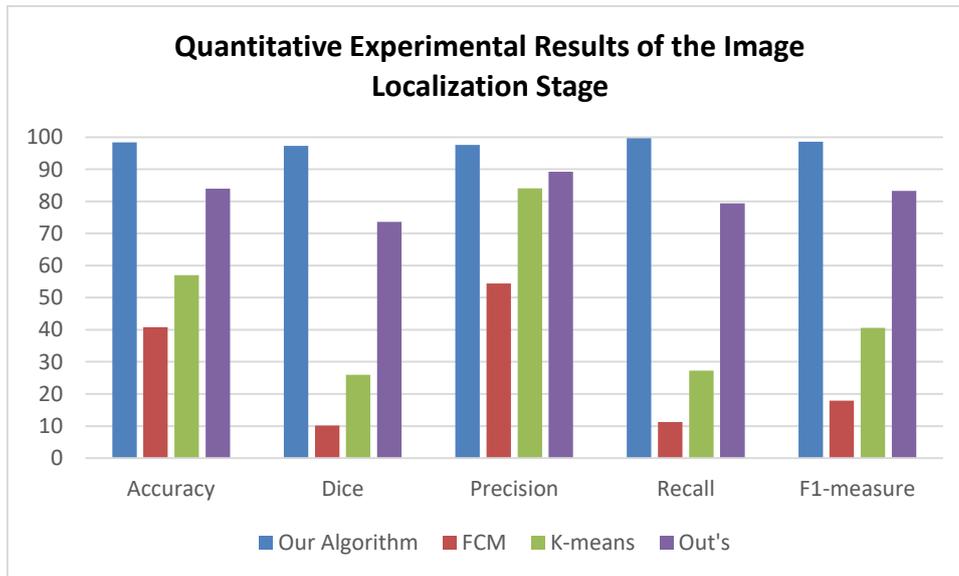


Figure (4.9): Quantitative Experimental Results of the Image Localization Stage Based on Using Different Unsupervised Clustering Algorithms.

It is worth noting that the shape of the image after the pre-processing process and the stage of localization of the image of the finger vein will be shown in the following figure (4.10):

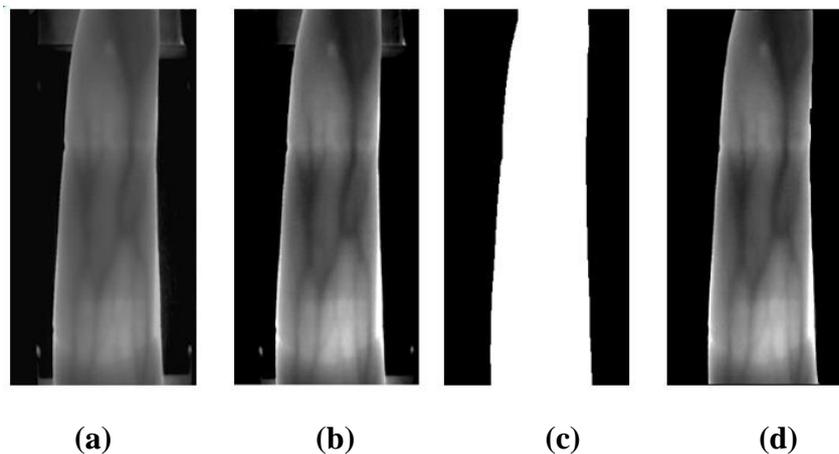


Figure (4.10): Different result for First and Second Stages in first Model. (a)Original image, (b) Enhanced image, (c) New K-mean Binary Mask, (d) Localized Finger Vein Image

4.5.3 Experiential Results of Finger Pattern Extraction Stage:

The third stage is the very important stage in the first model, because through this stage we have found the features needed, which is the extraction of the finger vein pattern, which is the unique feature that distinguishes people from each other. There are three steps in this stage which are the (miura_repeated_line_tracking) step, GPE step, GPO step.

The following sections will present the results of the three steps in this important stage, which is the stage of extracting the features that we want to compare on the basis of in the second model, knowing that the results that come out of this stage will be training data for the second model (**Fully Automated Finger Vein Pattern Extraction based Deep Semantic Segmentation**).

4.5.4 Subjective Miura_Repeated_Line_Tracking Result

The source [70] of this model was discussed in the third chapter, where the shape of the pattern of the finger vein appears inaccurately and has many separate lines that distort the shape of the pattern. Therefore, to address such a problem, the common techniques for such problems in the binary image of the finger vein pattern are using the morphological operation. Therefore, even when using this process, the shape of the pattern remains distorted. The following figure (4.11) shows a set of patterns produced from this model.

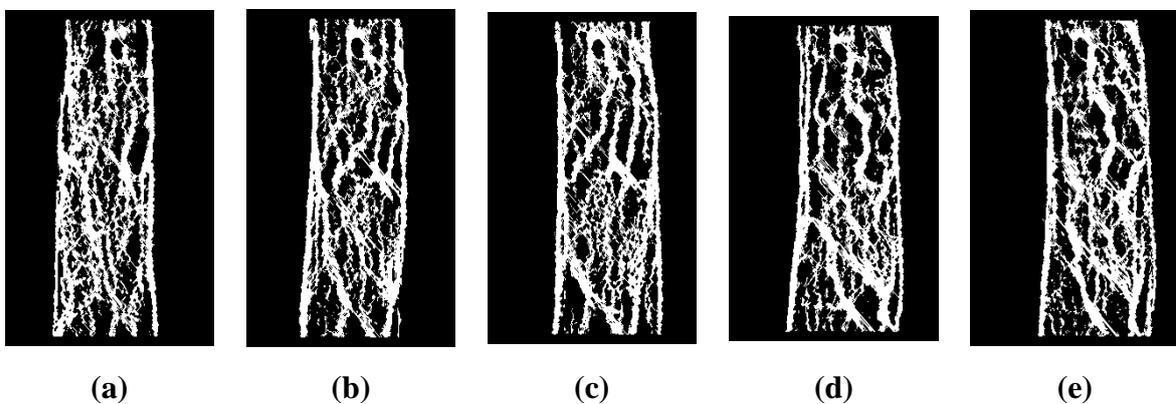


Figure (4.11): Different result For The First Step (Miura_Repeated_Line_Tracking Result) from Last Stage.

4.5.5 Subjective GPE and GPO Result

This step is considered the new idea to address the problem of morphological operation, as the goal of this technique is to find out the path of the vein pattern lines of the finger, as the algorithm or technology for this field was explained in the third chapter, where the results of tracing the path will be displayed and mapped in the form of a map to the locations of the veins, i.e. In the sense of places where pixels of dark color are collected, and whose hue is distinct from other pixels in the image used from the database, as shown in the following figure(4.12).

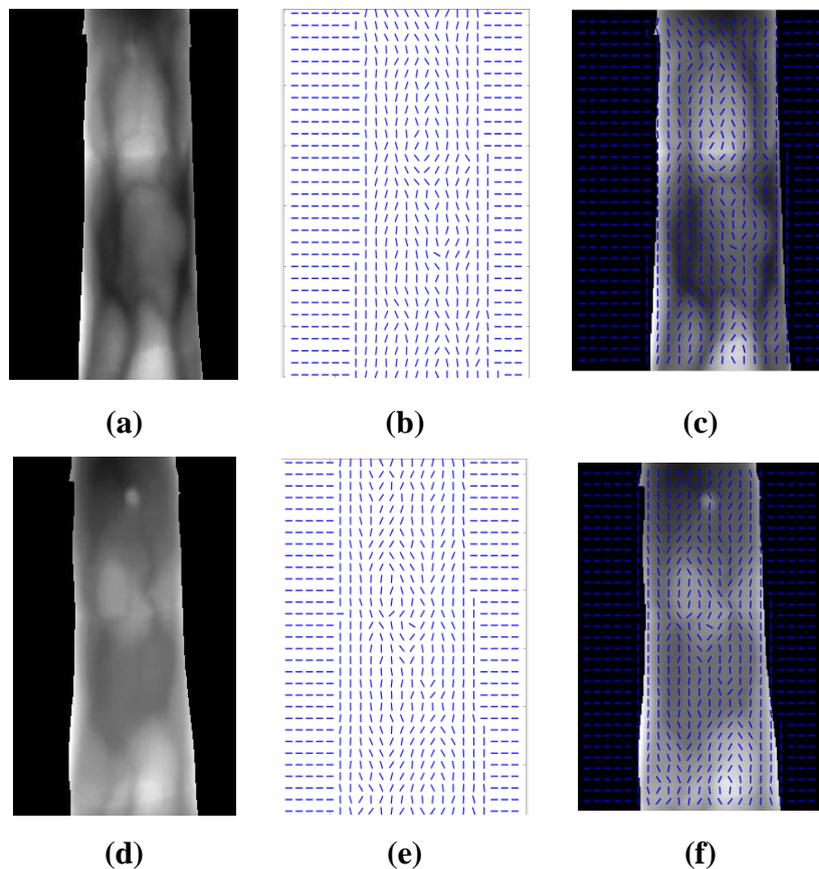


Figure (4.12): Finger Vein Direction Map.

After this technique, we come to the stage of developing the shape of the pattern after determining the path, as the image was divided into blocks, and this path was determined at the level of the block, meaning the places where the pixels gather in the same place and path. At this stage, we were able

to know the angle of curvature of these lines so; we connect the places of the separate lines and get an accurate shape of the finger vein pattern. Figure (4.13) depicts the final optimization results, which are based on the GPO For each block. In the development process GPO, we calculate the frequency of each direction, since if the frequency is of high value, we will take into account this block and make a connection to the separate line. But if the frequency is of low value, then we do not do the connection process and therefore we have a good shape for the finger vein pattern .We employed the same structural line and degree. Once the small object pixels in the digital finger vein line image have been eliminated; they are removed using the binary image closure approach.

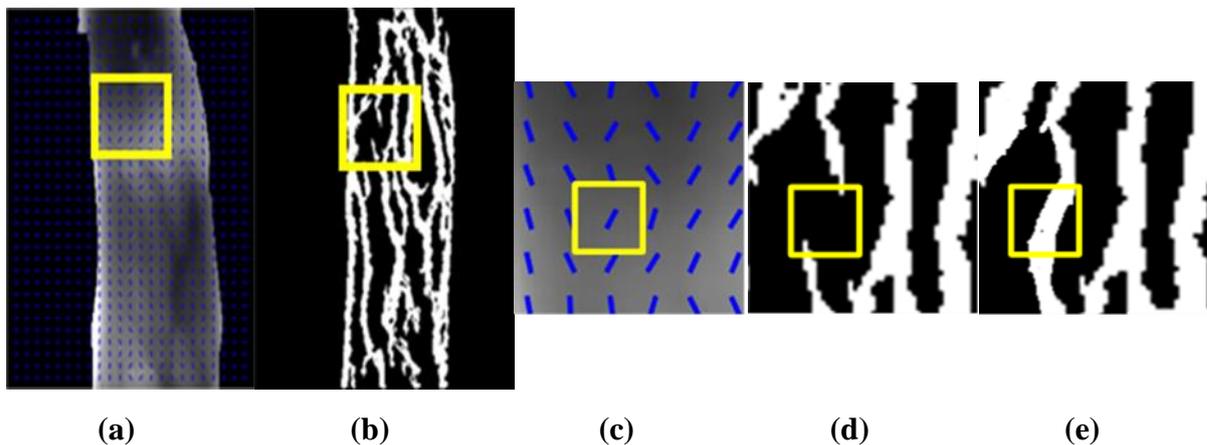


Figure (4.13): Line connection using GPO technique.

Where (a) is Global pattern Extraction Step image (b) is Finger vein pattern from previous miura_repeated_line_tracking (c) is One Block of the Orientation Estimation (d) is the block corresponding to the block we set the path of the finger pattern lines (e) is Separate line processed and gets the exact shape of the pattern we want.

The following figure (4.14) shows the results after completing the second step (GPE) and (GPO) and the clear effect on the shape of the pattern, which has improved significantly.

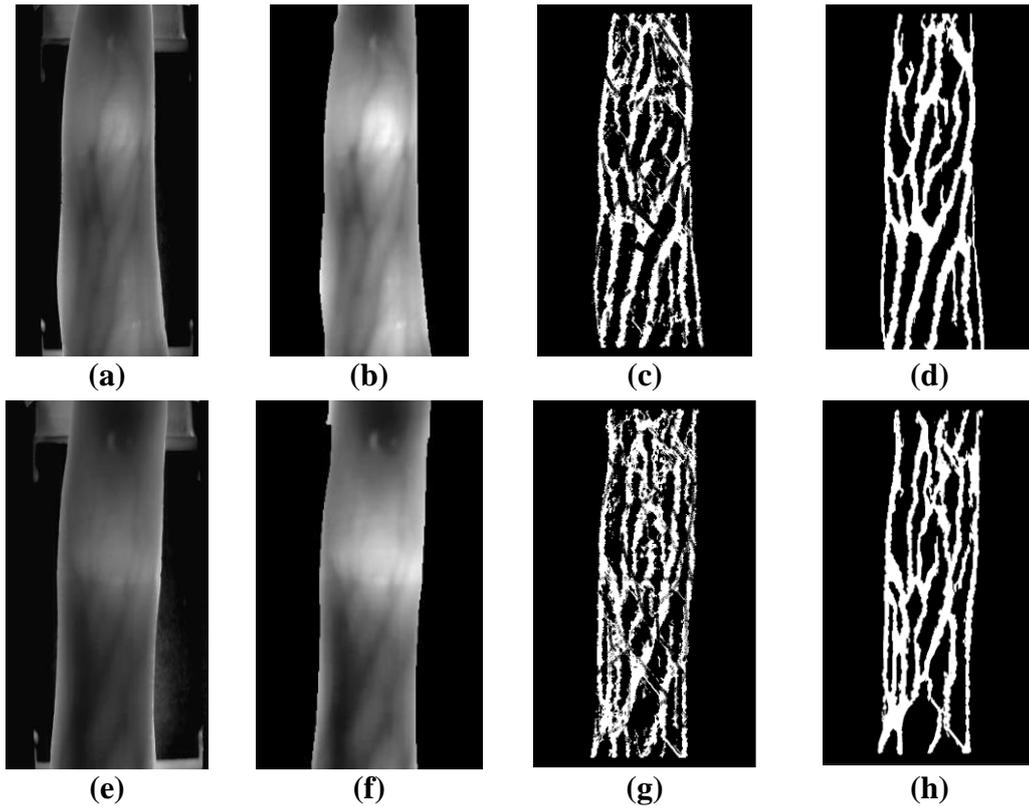


Figure (4.14): Finger Vein Line Extraction Results based on GPE and GPO.

Where (a) and (e) Original Image , (b) and (f) Localized Finger vein Image , (c) and (g) Miura_Repeated_Line_Tracking Result,(d) and (h) Final Pattern after applying GPE and GPO Mapping Technique. The final result for all processes in the first model is illustrated in figure (4.15).

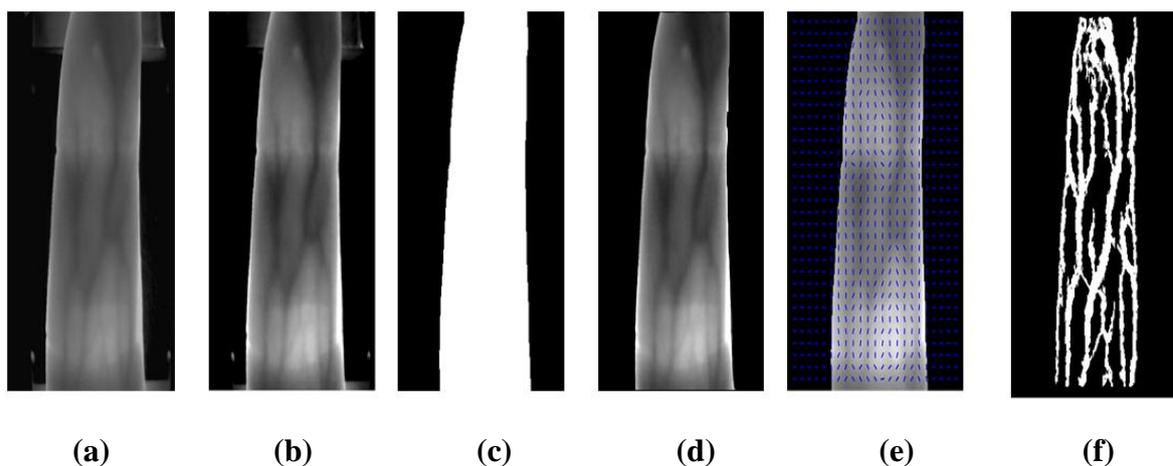


Figure (4.15): Final Result for all operation in the first model.

Where (a) Original Image , (b) Enhanced Finger vein Image , (c) Binary Mask , (d) Localized Finger vein Image, (e) GPE and GPO Mapping Technique, (f) Finger Vein Pattern Extraction. The first model was able to extract most of the finger vein lines. These states are automatically isolated as a training data set that is later used in the second model. However, some other cases in which the first model was unable to extract decent finger vein lines were considered as a test data set to be adapted to the second model.

4.6 Experiential Results of the Supervised Learning Model

In this section the experimental results of the deep semantic segmentation is discussed and illustrated .In term of evaluate the performance result of the second proposed model; the proposed model has been evaluated based on two different tasks. The first task is based on evaluate the fully automated deep semantic segmentation to generate (predict) a perfect binary pattern for each individual finger vein from the testing dataset. Second, the predicted binary pattern has been used to identify the human finger vein using different fingers such as the left, middle index, and right ring finger vein. For this reasons, different standard evaluation criteria are used to evaluate the performance of the proposed model such as dice coefficients and the intersection over union IoU that have been discussed in (2.25) and (2.26) respectively in section (2.10.2) in chapter two.

4.6.1 Training and Testing Scheme.

The first step in this model is to prepare the training dataset which are a binary mask pattern for each finger that has been generated in the first model. The whole dataset in second proposed model has been divided into different portions such as 80% of the whole data is training and 20% of the data is testing set. Table (4.3) shows the training, validation and testing datasets of second proposed model. As we mentioned earlier, the reason for using such a model is because some cases have failed to find a pattern shape for some database data, so it was isolated as test data, which constitutes a small

percentage, as the percentage 20% was replaced with some data from the training part.

Table (4.3): Training and Testing Data Partitions of Our Proposed System.

Data partition		Number of images
Training 80%	Training 60%	2,339 images
	Validation 20%	589 images
Testing 20%	593 images	
Total	3,521 images	

4.6.2 Fully Automated Binary Mask Prediction Results

Since the deep semantic segmentation have been proposed and used in this model, there are two class of results have been conducted here (training and testing results). All the results are explained, discussed, and illustrated in this section.

1. Training Performance Results.

Fully Automated Finger Vein Pattern Extraction based Deep Semantic Segmentation has been evaluated based on used different training parameters. We adjust the performance results based on using different epochs such as 5,10,15,20, and 25.

The experimental results of the training/validation dataset based on using 5 epochs are shown in Table (4.4), this table shows that highest IOU score that is achieved on the training was (85%) while (84%) on the validation dataset. Also, we can notice that the minimizes loss value of the performance on the training dataset was (7.998%) while (8.387%) on the validation dataset. Figure (4.16) illustrates both training/validation loss and IOU curves for the five epochs.

Table (4.4): Training Performance Results based 5 Epochs Only.

Epochs	Training		Validation	
	Loss	IOU	Loss	IOU
Epoch 1	0.1522	0.7426	0.105	0.8155
Epoch 2	0.09754	0.8263	0.09388	0.8326
Epoch 3	0.08743	0.8424	0.08859	0.8409
Epoch 4	0.08327	0.8491	0.09169	0.8347
Epoch 5	0.07998	0.8546	0.08387	0.848

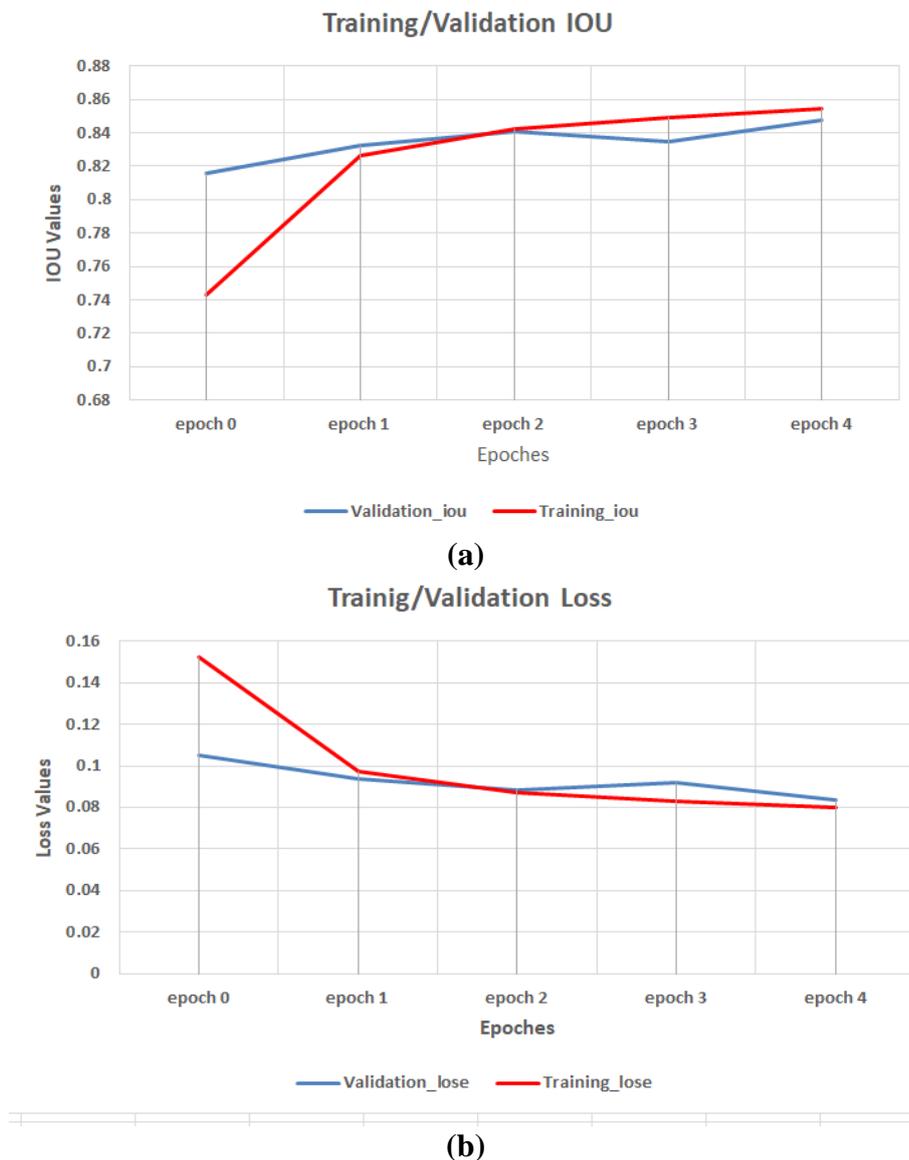
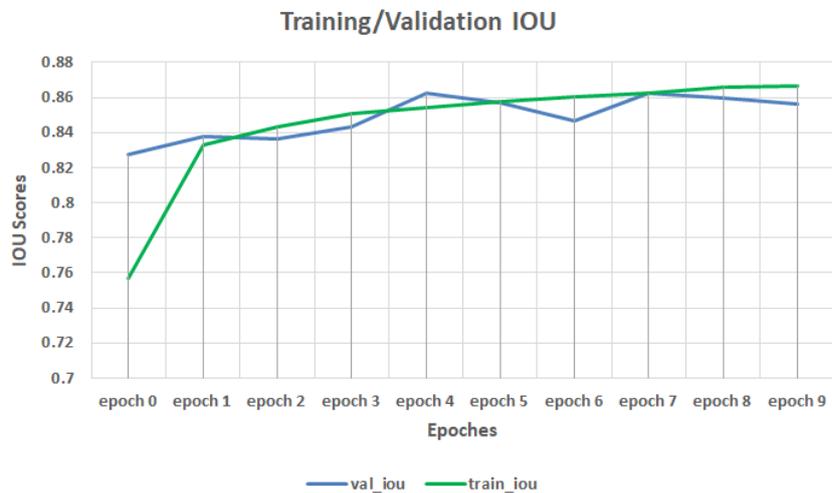
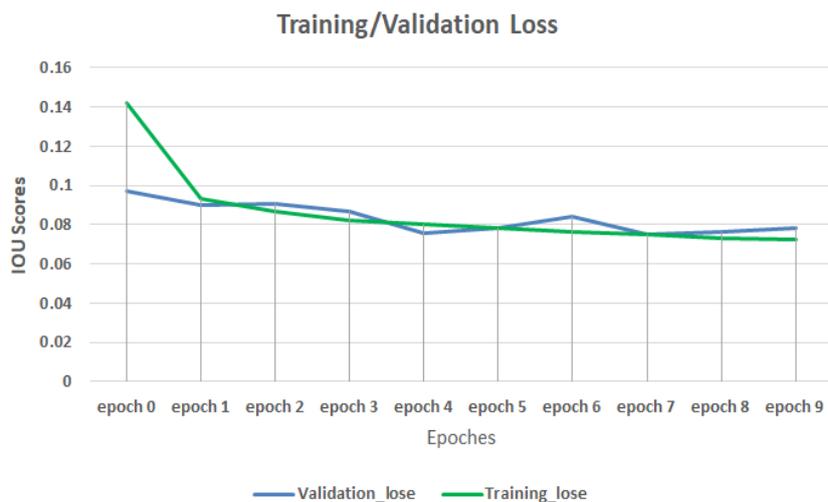


Figure (4.16): Training and Validation Performance Results Based on Using 5 Epochs only (a) Represents the Intersection over Union Curve (IoU), (b) Loss Function Curve.

The experimental results of the training/validation dataset based on using 10 epochs are shown in Table (4.5), this table shows that highest IOU score that is achieved on the training was (86%) while (85.5%) on the validation dataset. Also, we can notice that the minimizes loss value of the performance on the training dataset was (7.267%) while (7.839%) on the validation dataset. Figure (4.17) illustrates both training/validation loss and IOU curves for the 10 epochs.

Table (4.5): Training Performance Results using 10 Epochs Only.

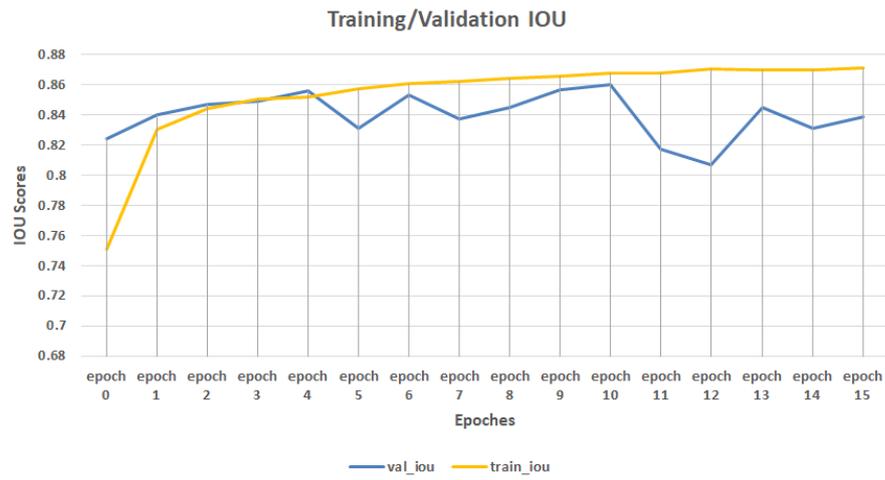
Epochs	Training		Validation	
	Loss	IOU	Loss	IOU
Epoch 1	0.1422	0.7569	0.09699	0.8276
Epoch 2	0.09331	0.8327	0.08988	0.838
Epoch 3	0.08688	0.8429	0.09096	0.8361
Epoch 4	0.08195	0.851	0.08675	0.8431
Epoch 5	0.08014	0.854	0.07539	0.8621
Epoch 6	0.07807	0.8575	0.07849	0.8567
Epoch 7	0.07638	0.8603	0.08424	0.8469
Epoch 8	0.0752	0.8622	0.07501	0.8624
Epoch 9	0.07292	0.8661	0.07659	0.8597
Epoch 10	0.07267	0.8666	0.07839	0.8563

**(a)****(b)****Figure (4.17): Training and Validation Performance Results Based on Using 10 Epochs only (a) Represents the Intersection over Union Curve (IoU), (b) Loss Function Curve.**

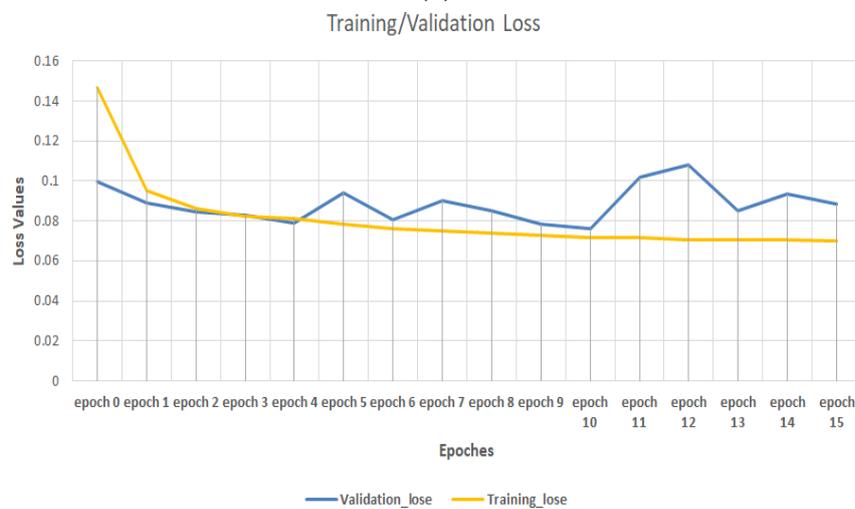
The experimental results of the training/validation dataset based on using 15 epochs are shown in Table (4.6), this table shows that highest IOU score that is achieved on the training was (87%) while (83%) on the validation dataset. Also, we can notice that the minimizes loss value of the performance on the training dataset was (7.998%) while (8.387%) on the validation dataset. Figure (4.18) illustrates both training/validation loss and IOU curves for the 15 epochs.

Table (4.6): Training Performance Results using 15 Epochs Only.

Epochs	Training		Validation	
	Loss	IOU	Loss	IOU
Epoch 1	0.1466	0.7511	0.09976	0.824
Epoch 2	0.09516	0.8302	0.08893	0.8402
Epoch 3	0.08639	0.844	0.08451	0.8469
Epoch 4	0.08248	0.8504	0.08305	0.8493
Epoch 5	0.08149	0.8519	0.079	0.8559
Epoch 6	0.07837	0.8572	0.09392	0.8313
Epoch 7	0.07627	0.8607	0.08053	0.8535
Epoch 8	0.07531	0.8622	0.08996	0.8373
Epoch 9	0.07416	0.8641	0.08517	0.8453
Epoch 10	0.0731	0.8658	0.07828	0.8569
Epoch 11	0.07182	0.8679	0.0764	0.8603
Epoch 12	0.07167	0.8681	0.1018	0.8173
Epoch 13	0.07038	0.8703	0.1081	0.8067
Epoch 14	0.07066	0.8697	0.0852	0.845
Epoch 15	0.0706	0.8698	0.09335	0.8312



(a)



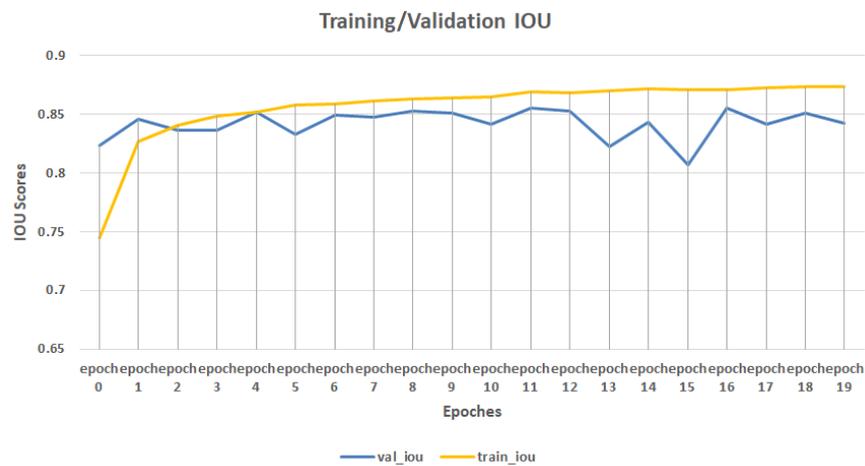
(b)

Figure (4.18): Training and Validation Performance Results Based on Using 15 Epochs only (a) Represents the Intersection over Union Curve (IoU), (b) Loss Function Curve.

The experimental results of the training/validation dataset based on using 20 epochs are shown in Table (4.7), this table shows that highest IOU score that is achieved on the training was (87%) while (84%) on the validation dataset. Also, we can notice that the minimizes loss value of the performance on the training dataset was (6.825%) while (8.656%) on the validation dataset. Figure (4.19) illustrates both training/validation loss and IOU curves for the 20 epochs.

Table (4.7): Training Performance Results using 20 Epochs Only.

Epochs	Training		Validation	
	Loss	IOU	Loss	IOU
Epoch 1	0.1506	0.745	0.09972	0.8236
Epoch 2	0.09735	0.8265	0.08592	0.8455
Epoch 3	0.08812	0.8411	0.09109	0.8364
Epoch 4	0.0835	0.8485	0.09109	0.8362
Epoch 5	0.08134	0.8521	0.08151	0.8517
Epoch 6	0.07772	0.8582	0.09268	0.8333
Epoch 7	0.0773	0.8588	0.0826	0.8497
Epoch 8	0.0755	0.8618	0.08398	0.8478
Epoch 9	0.07454	0.8634	0.08096	0.8526
Epoch 10	0.074	0.8643	0.08209	0.8507
Epoch 11	0.07351	0.865	0.08731	0.8416
Epoch 12	0.07116	0.8691	0.07941	0.8551
Epoch 13	0.07152	0.8683	0.08091	0.8524
Epoch 14	0.0707	0.8698	0.09857	0.8227
Epoch 15	0.0696	0.8716	0.08617	0.8433
Epoch 16	0.07007	0.8707	0.1076	0.8074
Epoch 17	0.07001	0.8707	0.07892	0.8554
Epoch 18	0.06887	0.8727	0.08703	0.8418
Epoch 19	0.06816	0.8738	0.08145	0.851
Epoch 20	0.06825	0.8736	0.08656	0.8422



(a)

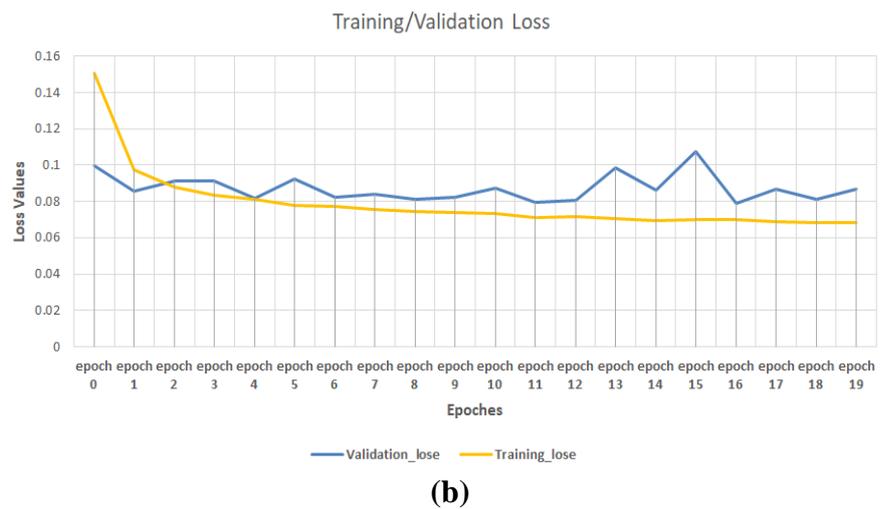


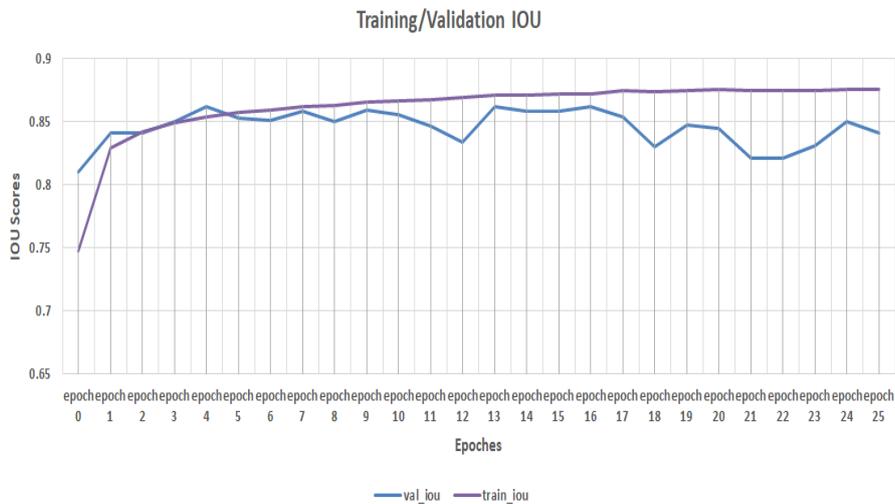
Figure (4.19): Training and Validation Performance Results Based on Using 20 Epochs only (a) Represents the Intersection over Union Curve (IoU), (b) Loss Function Curve.

The experimental results of the training/validation dataset based on using 25 epochs are shown in Table (4.8), this table shows that highest IOU score that is achieved on the training was (88%) while (84%) on the validation dataset. Also, we can notice that the minimizes loss value of the performance on the training dataset was (6.675%) while (8.692%) on the validation dataset. Figure (4.20) illustrates both training/validation loss and IOU curves for the 25 epochs.

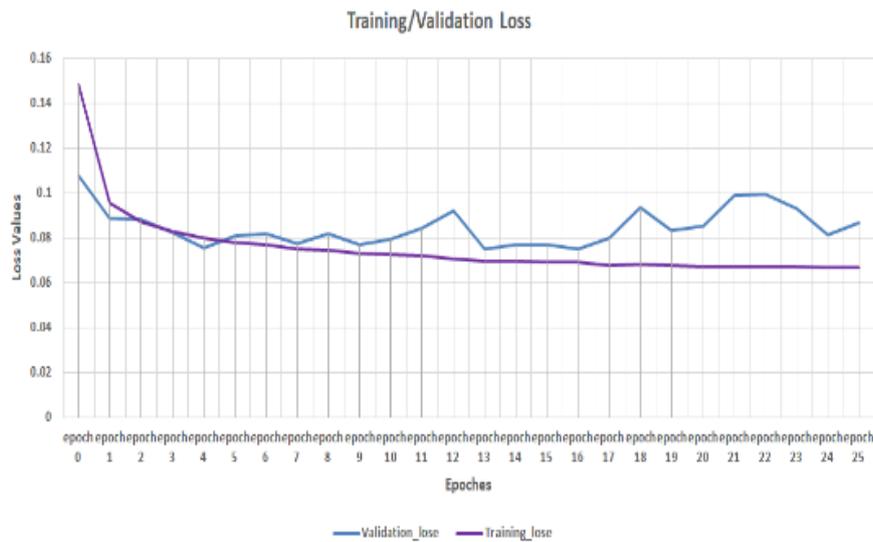
Table (4.8): Training Performance Results using 25 Epochs Only.

Epochs	Training		Validation	
	Loss	IOU	Loss	IOU
Epoch 1	0.1486	0.7473	0.1081	0.8106
Epoch 2	0.09542	0.8295	0.08865	0.8413
Epoch 3	0.0875	0.842	0.08828	0.8415
Epoch 4	0.08304	0.8492	0.08267	0.8505
Epoch 5	0.08004	0.8542	0.07561	0.8619
Epoch 6	0.07828	0.8572	0.08092	0.8532
Epoch 7	0.07701	0.8592	0.08202	0.8512
Epoch 8	0.07524	0.8622	0.07756	0.8581
Epoch 9	0.07457	0.8633	0.08215	0.8504
Epoch 10	0.07311	0.8657	0.07686	0.8594
Epoch 11	0.07255	0.8667	0.07934	0.8554
Epoch 12	0.07224	0.8671	0.08431	0.8466
Epoch 13	0.07074	0.8697	0.09225	0.8334
Epoch 14	0.06996	0.871	0.07494	0.8624

Epoch 15	0.06995	0.8709	0.07731	0.8583
Epoch 16	0.06934	0.8719	0.0773	0.8582
Epoch 17	0.06916	0.8722	0.07492	0.8625
Epoch 18	0.06772	0.8747	0.08008	0.8535
Epoch 19	0.06803	0.874	0.09384	0.8306
Epoch 20	0.06757	0.8748	0.08326	0.8479
Epoch 21	0.06728	0.8753	0.08517	0.8446
Epoch 22	0.06746	0.8749	0.09905	0.8214
Epoch 23	0.06735	0.8751	0.09973	0.8209
Epoch 24	0.06742	0.8749	0.0932	0.831
Epoch 25	0.06677	0.876	0.08166	0.8505



(a)



(b)

Figure (4.20): Training and Validation Performance Results Based on Using 25 Epochs only (a) Represents the Intersection over Union Curve (IoU), (b) Loss Function Curve.

For all epochs figure information , the training curve represents the images during the training phase, while the valid curve represents the testing of the data during the training phase, where we also note that the training curve for IoU starts with small values and ends with large values, which means that the training is done correctly and the same case for Loss function where the curve It starts with large values, and these values gradually decrease, which means a decrease in the value of the error, and this is a good indication that the data training is done correctly.

2. Testing Performance Results

Fully Automated Finger Vein Pattern Extraction based Deep Semantic Segmentation has been evaluated based on used (593) images on the testing dataset. The samples of the testing dataset have not been included on the training data set which means that the proposed model has not seen the testing images during the training phase. The values shown in the table (4.9) were calculated by comparing the image patterns of the test data with what the model predicted, where the IoU criterion equation (2.25) ,section (2.10.2) in chapter two was used, which is used to match the pixels between the Ground Truth and the mask predicted by the model, where the part (TP) represents the intersection area between the two images and is meant by the two images the Ground Truth and the mask that the model predicted during the testing phase, while the part (FP) represents the area that the model predicted outside the bounds of the Ground Truth, and (FN) represents the number of pixels inside the Ground Truth that the model failed to predict. Table shows that the higher accuracy (94%) has been achieved by second proposed model.

Table (4.9): Finger vein Testing Performance Result.

Criteria	Average values
Accuracy	94.08%
Precession	91.33%
Recall	93.74%
F1-measure	92.51%

Figure (4.21) show the similarity between the ground truth mask of the tested finger vein images and the predicted mask that the deep semantic segmentation based PU-CNN model predicted it after the training operation.

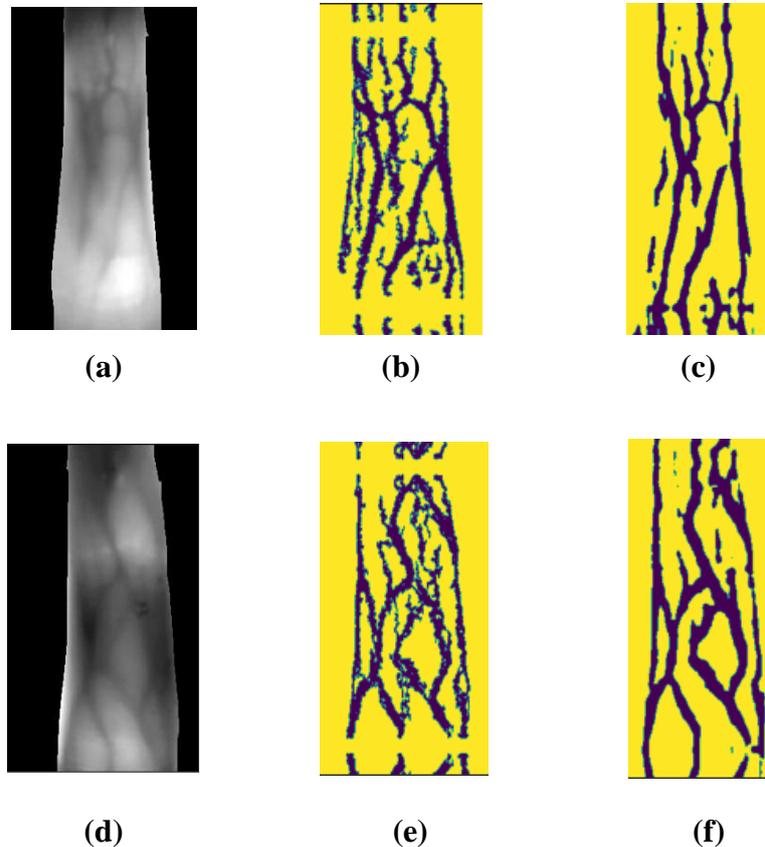


Figure (4.21): Fully Automated Finger Vein Pattern Extraction based Deep Semantic Segmentation using PU Loss Function Result (a, d) Localized Finger Vein Image (b, e) Finger Vein Pattern (Ground Truth) (c, f) Predicted Pattern mask.

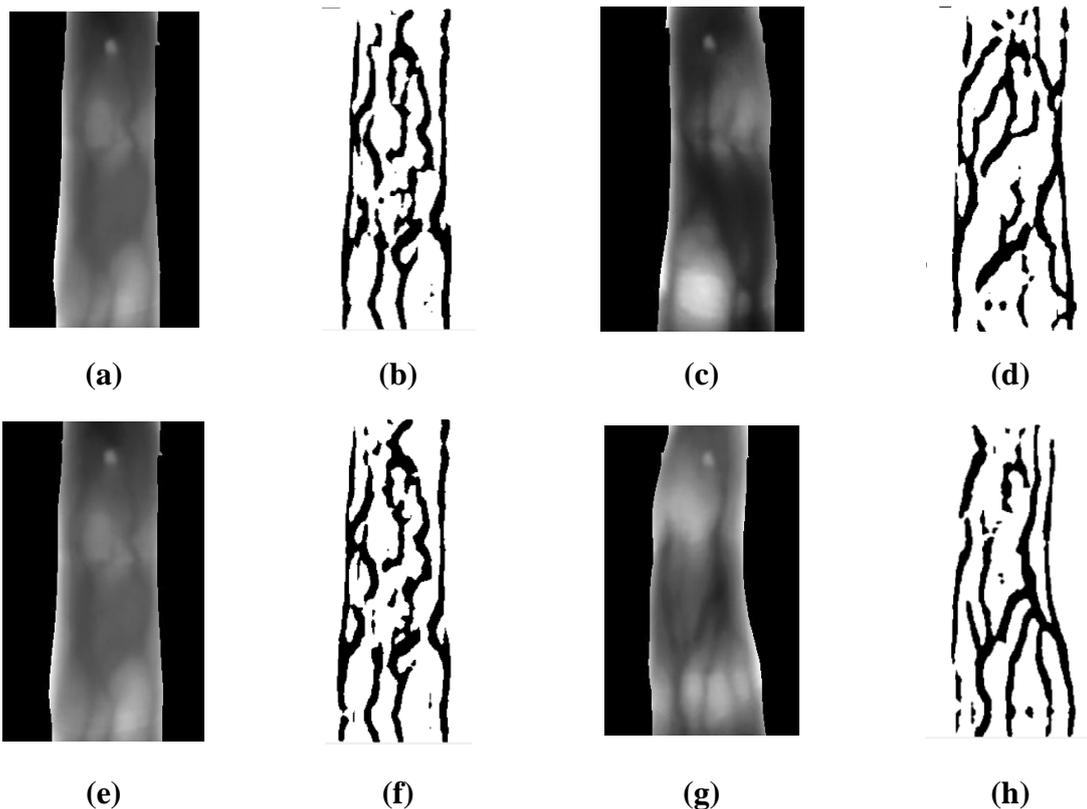
4.6.3 Experiential Results of the Finger Vein Identification

The other part of the experimental results has been drawn based on the human identification using the predicted finger vein binary masks. The testing results are based on using two samples from the same person using different finger from both left and right hand fingers. The index, middle and ring from both hand fingers.

Figure (4.21) illustrates some predicted masks for the same persons and different predicted masks for the two different persons. It is also shows the fully automated finger vein pattern prediction for two finger vein images from

the same person as is shown in (a) and (e). Anyone can notice that the predicted masks (binary pattern) are almost identical as is shown in (b) and (f) which leading us to have a decision that those two finger veins belong to the same person. The same case here is applied for the original finger vein images that are shown in Figure (4.22) (i) and (l) respectively.

Also, anyone can notice that the predicted mask (binary pattern) that the deep semantic segmentation has predicted is not identical which leading us to have a decision that those two finger veins are not belonging to the same person as we note in finger vein images (c),(g). The same case here is applied for the original finger vein images that are shown in Figure (4.21) (k) and (o) respectively.



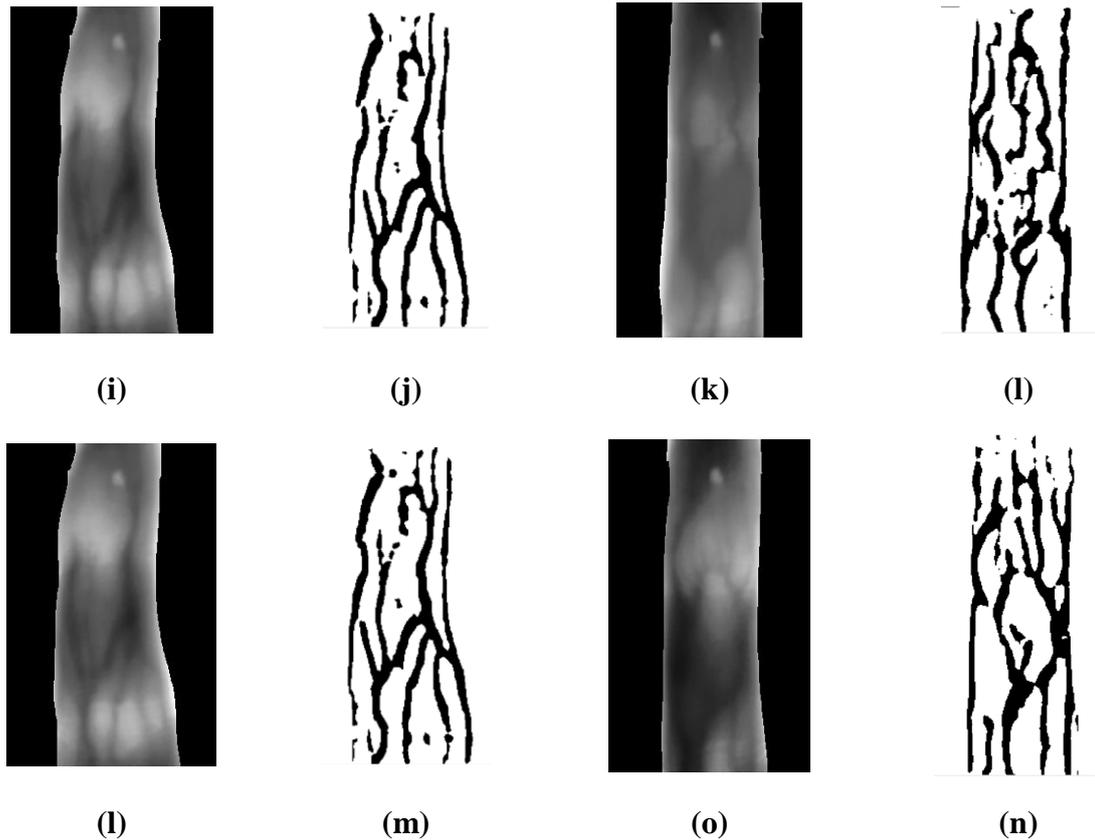


Figure (4.22): Fully Automated Finger Vein Human Identification Results using the Predicted Masks That Been Generated from the Deep Semantic Segmentation (a),(e) and (i),(l) are the Localized Finger Vein Image For The Same Person , (b),(f) and (j),(m) are The Predicted Mask For The Same Persons and (c), (g), (k),(o) are Localized Images For the Different Persons and (d),(h),(l),(n) are The Predicted Mask For The Different Persons.

In terms of identify and verify the human finger vein image, each two finger vein images are fed to out fully automated finger vein binary mask prediction approach and produced two binary masks. The two binary masks are evaluated by calculating the similarity using evaluation criteria to see if they are belonging to the same person or not. Table (4.10) illustrates the performance results of the human identification stage which clearly state that two finger veins that are belong to the same person has achieved (95%) matching score while two different binary masks have (73%) similarity scores.

Table (4.10): Finger Vein Identification/Verification Based Similarity Performance for the same person.

Criteria	Same person	different person
Accuracy	95.16553	73.842425
Precession	95.993765	83.6310983
Recall	95.959113	83.98858
F1-measure	95.97592	83.83438

Chapter Five

Conclusion and Future Work

5.1 Conclusion

From the results which were carried out in chapter four, the following points are inferred:

1. This thesis proposes a new optimization model for k-means clustering based unsupervised learning algorithm.
2. This thesis proposes another optimization technique to track the path of hidden veins inside the finger using global pattern extraction GPE and global pattern optimization GPO.
3. The first unsupervised learning model does not depend on distance approach like k-means; instead, it depends on intensity difference that gives an accurate result for identifying and localizing finger vein images.
4. The first unsupervised learning model has been evaluated and compared with different standard approaches in the unsupervised learning such as k-means, FCM, Out's thresholding.
5. The evaluation stage shows that the first approach has achieved very significant results (better accuracy).
6. The performance results show that the designed model is found to be good for finger vein image isolation and detection.
7. First proposed model shows very significant performance results in terms of generating and predicting very clear binary masks.
8. The purpose of the first model is to generate clear pattern for each finger in the dataset.
9. The second model came to treat some cases that did not generate a pattern for the finger vein image.

10. The second model not need to label data for training because the data are automatically labeled using Positive Unlabeled loss function technique.
11. This thesis presents new finger vein identification and verification based on find the pure similarity of the actual finger vein pattern by proposing two different unsupervised and supervised models.
12. Identify or verify human based on their actual finger vein pattern has not been purely proposed yet.
13. The second approach is designed based on using a new loss function. The new loss function is basically based on label one class only which is the finger vein lines that have been extracted from the first mode for this reason it is called positive-unlabeled loss.
14. The proposed system shows a higher accuracy (98.6%) that was achieved by our optimized algorithm, while the second highest score is (76.5), which was achieved by Out's thresholding algorithm.
15. Although, we did not have any comparison between our deep learning model and the state-of-the-art since our model that has been designed to predict, extract, and identify the human based on their binary pattern while the other deep learning models have been designed based on using the whole finger vein images to identify the subject.

5.2 Future work

For future work, different suggestions can be proposed to improve this work, such:

1. Our future works is based on using a new sensor to extract a new finger vein datasets. The new sensor is able to extract a thermal finger vein image.
2. We are planning to design an enforcement deep learning approach to extract the interest regions in the finger vein from the whole image and will designed a guided deep enforcement learning approach to predict the probability of the human identification.
3. Collecting data at the local level by making a device with simple tools and applying it at the level of government institutions instead of fingerprints that still suffer from privacy violations.
4. Proposed other methods for feature level fusion and for matching score fusion.
5. Proposed another method for matching and classification based on another technique such as genetic algorithm or Partial Swarm Optimization.
6. Work to improve the image further to obtain more features in the stages of pattern extraction.

References

- [1] Mahmood, M. R., & Abdulazeez, A. M, "A Comparative study of a new hand recognition model based on line of features and other techniques", In international Conference of reliable information and communication technology (pp. 420-432). Springer, Cham, 2017.
- [2] Yang, J., & Wang, J. , "Finger-vein image restoration considering skin layer structure", In International Conference on Hand-Based Biometrics (pp. 1-5),IEEE, 2011.
- [3] Yang, J., & Zhang, B," Scattering removal for finger-vein image enhancement", In International Conference on Hand-Based Biometrics (pp. 1-5). IEEE,2011.
- [4] Sabhanayagam, T., Venkatesan, V. P., & SenthamaraiKannan,"A comprehensive survey on various biometric systems". International Journal of Applied Engineering Research, 13(5), 2276-2297,2018.
- [5] Bouchrika," A survey of using biometrics for smart visual surveillance: Gait recognition", In Surveillance in Action (pp. 3-23),Springer Cham, 2018.
- [6] Zhang, X., & Wang, W," Finger vein recognition method based on GLCM-HOG and SVM", In 3rd International Conference on Information Systems and Computer Aided Education (ICISCAE) (pp. 698-701), IEEE,2020.
- [7] Anand, J., Flora, T. A., & Philip, A. S," Finger-vein based biometric security system", International Journal of Research in Engineering and Technology eISSN, 2(12), 197-200,2013.
- [8] Kant, C , & Nath, R," Reducing process-time for fingerprint identification system", International Journals of Biometric and Bioinformatics, 3(1), 1-9,2009.

- [9] Saxena, N., Saxena, V., Dubey, N., & Mishra, "Hand geometry: A new method for biometric recognition", *International Journal of Soft Computing and Engineering (IJSCE)*, 2(6), 2231-2307,2013.
- [10] Therar, H. M., Mohammed, E. A., & Ali," Biometric signature based public key security system" *International Conference on Advanced Science and Engineering (ICOASE)*, (pp. 1-6), IEEE,2020.
- [11] Borah, T. R., Sarma, K. K., & Talukdar," Retina recognition system using adaptive neuro fuzzy inference system" *International Conference on Computer, Communication and Control (IC4)* (pp. 1-6), IEEE,2015.
- [12] Patil, P., & Patil," Offline signature recognition using global features", *International Journal of Emerging Technology and Advanced Engineering*, 3(1), 408-411,2013.
- [13] Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M. P., & Iyengar," A survey on deep learning: Algorithms, techniques, and applications", *ACM Computing Surveys (CSUR)*, 51(5), 1-36,2018.
- [14] Baker, E. J., Alazawi, S. A., Ahmed, N. T., Ismail, M. A., Hassan, R., Abd Halim, S., & Sutikno," User identification system for inked fingerprint pattern based on central moments". *Indonesian Journal of Electrical Engineering and Computer Science*, 24(2), 1149-1160,2021.
- [15] Ali, M. M., Mahale, V. H., Yannawar, P., & Gaikwad, "Overview of fingerprint recognition system". *international conference on electrical, electronics, and optimization techniques (ICEEOT)* (pp. 1334-1338),2016.
- [16] Akintoye, K. A., Rahim, M. S. M., & Abdullah," Challenges of finger vein recognition system: a theoretical perspective". *International Journal of Emerging Technology and Advanced Engineering*, 8(2), 196-204,2018.

- [17] K. Syazana-Itqan, A. R. Syafeeza, N. M. Saad, N. A. Hamid, and W. H. Bin Mohd Saad, "A Review of Finger-Vein Biometrics Identification Approaches", *Indian J. Sci. Technol.*, vol. 9, no. 32, 2016.
- [18] W. Yang, X. Huang, F. Zhou, and Q. Liao, "Comparative competitive coding for personal identification by using finger vein and finger dorsal texture fusion", *Inf. Sci. (Ny)*, vol. 268, pp. 20–32, 2014.
- [19] B. T. Ton and R. N. J. Veldhuis, "A high quality finger vascular pattern dataset collected using a custom designed capturing device," *Proc. - 2013 Int. Conf. Biometrics, ICB 2013*, 2013.
- [20] Y. Lu, S. J. Xie, S. Yoon, Z. Wang, and D. S. Park, "An available database for the research of finger vein recognition," *Proc. 2013 6th Int. Congr. Image Signal Process. CISP 2013*, vol. 1, no. Cisp, pp. 410–415, 2013.
- [21] Zhou, Y., & Kumar, " Human identification using palm-vein images", *IEEE transactions on information forensics and security*, 6(4), 1259-1274,2011.
- [22] Y. Yin, L. Liu, and X. Sun, "SDUMLA-HMT: A multimodal biometric database," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 7098 LNCS, pp. 260–268, 2011.
- [23] J. Zhang and J. Yang, "Finger-vein image enhancement based on combination of gray-level grouping and circular gabor filter", *International Conference on Information Engineering and Computer Science*, pp. 1–4,2009.
- [24] C.-B. Yu, D.-M. Zhang, H.-B. Li, and F.-F. Zhang, "Finger-vein image enhancement based on muti-threshold fuzzy algorithm", *International Congress on Image and Signal Processing*, pp. 1–3, 2009.

- [25] H. Winnemöller, "Xdog: advanced image stylization with extended difference-of-gaussians", in *Proceedings of the ACM SIGGRAPH/eurographics symposium on non-photorealistic animation and rendering*, pp. 147–156, 2011.
- [26] K. A. Akintoye, M. R. M. Shafry, and A. H. Abdullah, "A novel approach for finger vein pattern enhancement using Gabor and Canny edge detector", *Int. J. Comput. Appl.*, vol. 157, no. 2, 2017.
- [27] L. Yang, G. Yang, Y. Yin, and X. Xi, "Finger vein recognition with anatomy structure analysis", *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 8, pp. 1892–1905, 2017.
- [28] Yang, J., & Wang, J, " Finger-vein image restoration considering skin layer structure".International Conference on Hand-Based Biometrics , (pp. 1-5) IEEE,2011.
- [29] Liu, Z., & Song, S," An embedded real-time finger-vein recognition system for mobile devices", *IEEE Transactions on consumer Electronics*, 58(2), 522-527,2012.
- [30] Tiwari, S., Chourasia, J. N., & Chourasia, " A review of advancements in biometric systems", *International Journal of Innovative Research in Advanced Engineering*, 2(1), 187-204,2015.
- [31] R. Dwivedi, S. Dey, M. Anand, and S. Apurv, "A fingerprint based crypto - biometric system for secure communication", *J. Ambient Intell. Humaniz. Comput.*, no. 0123456789, 2019.
- [32] B. Huang, Y. Dai, R. Li, D. Tang, and W. Li, "Finger-vein authentication based on wide line detector and pattern normalization", *Proc. - Int. Conf. Pattern Recognit.*, vol. 1, pp. 1269–1272, 2010.
- [33] Cattani, C., & Pierro," On the fractal geometry of DNA by the binary image analysis", *Bulletin of Mathematical Biology*, 75(9), 1544-1570,2013.

- [34] T. S. Engg, "Color Traits Transfer to Grayscale Images 3 . Color trait transfer algorithm using Kekre ' s LUV Color Space of Source Image RGB space of Source Image", pp. 82–85, 2008.
- [35] Owotogbe, J. S., Ibiyemi, T. S., & Adu," Edge detection techniques on digital images-a review", *Int J Innov Sci Res Technol*, 4, 329-332,2019.
- [36] Saifullah, S., Drezewski, R., Khaliduzzaman, A., Tolentino, L. K., & Ilyos, "K-Means Segmentation Based-on Lab Color Space for Embryo Detection in Incubated Egg", *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI)*, 8(2), 175-185,2022.
- [37] Sundararajan," Digital image processing: a signal processing and algorithmic approach" Springer,2017.
- [38] G. Singh and A. Mittal, "Various Image Enhancement Techniques- A Critical Review", *Int. J. Innov. Sci. Res.*, vol. 10, no. 2, pp. 267–274, 2014.
- [39] L. Zhang, X. Wang, X. Dong, L. Sun, W. Cai, and X. Ning, "Finger Vein Image Enhancement Based on Guided Tri-Gaussian Filters", *ASP Trans. Pattern Recognit. Intell. Syst.*, vol. 1, no. 1, pp. 17–23, 2021.
- [40] B. R. Jana, H. Thotakura, A. Baliyan, M. Sankararao, R. G. Deshmukh, and S. R. Karanam, "Pixel density based trimmed median filter for removal of noise from surface image", *Appl. Nanosci.*, no. 0123456789, 2021.
- [41] U. Erkan, D. N. H. Thanh, L. M. Hieu, and S. Enginoglu, "An iterative mean filter for image denoising", *IEEE Access*, vol. 7, pp. 167847–167859, 2019.
- [42] W. Gao, L. Yang, X. Zhang, and H. Liu, "An improved Sobel edge detection" , *IEEE Int. Conf. Comput. Sci. Inf. Technol. ICCSIT 2010*, vol. 5, pp. 67–71, 2010.
- [43] J. S. Goldstein, I. S. Reed, and L. L. Scharf, "A multistage representation of the wienerfilter based on orthogonal projections " , *IEEE Trans. Inf. Theory*, vol. 44, no. 7, pp. 2943–2959, 2013.

- [44] Abedi, M., Mosazadeh, K., Dehghani, H., & MadanchiZare," Geological noise removal in geophysical magnetic survey to detect unexploded ordnance based on image filtering", *Int. J. Informatics Inf. Syst* , 23 - 11 9(5), 2015.
- [45] Li, T., Wei, L., & Hsu, W," A multi-pronged evaluation for image normalization techniques" *IEEE 18th International Symposium on Biomedical Imaging (ISBI)* (pp. 1292-1296), 2021.
- [46] H. Henderi, T. Wahyuningsih, and E. Rahwanto, "Comparison of Min-Max normalization and Z-Score Normalization in the K-nearest neighbor (kNN) Algorithm to Test the Accuracy of Types of Breast Cancer ", *Int. J. Informatics Inf. Syst.*, vol. 4, no. 1, pp. 13–20, 2021.
- [47] M. Zhou, K. Jin, S. Wang, J. Ye, and D. Qian, " Color Retinal Image Enhancement Based on Luminosity and Contrast Adjustment " , *IEEE Trans. Biomed. Eng.*, vol. 65, no. 3, pp. 521–527, 2018.
- [48] Oo, S. Z., & Khaing," Brain tumor detection and segmentation using watershed segmentation and morphological operation", *International Journal of Research in Engineering and Technology*, 3(3), 367-374, 2014.
- [49] Badillo, S., Banfai, B., Birzele, F., Davydov, I. I., Hutchinson, L., Kam-Thong, T., ... & Zhang," An introduction to machine learning. *Clinical pharmacology & therapeutics* ", 107(4), 871-885,2020.
- [50] Mahesh," Machine learning algorithms-a review ", *International Journal of Science and Research (IJSR)*.[\[Internet\]](#), 9, 381-386,2020.
- [51] Alloghani, M., Al-Jumeily, D., Mustafina, J., Hussain, A., & Aljaaf, A. J." A systematic review on supervised and unsupervised machine learning algorithms for data science *Supervised and unsupervised learning for data science* ", 3-21, *Springer* ,2020.

- [52] A. Bansal, M. Sharma, and S. Goel, "Improved K-mean Clustering Algorithm for Prediction Analysis using Classification Technique in Data Mining", *Int. J. Comput. Appl.*, vol. 157, no. 6, pp. 35–40, 2017.
- [53] Nayak, J., Naik, B., & Behera, "Fuzzy C-means (FCM) clustering algorithm: a decade review from 2000 to 2014 ", *Computational intelligence in data mining-volume 2*, 133-149,2015.
- [54] Dargan, S., Kumar, M., Ayyagari, M. R., & Kumar, "A survey of deep learning and its applications: a new paradigm to machine learning. *Archives of Computational Methods in Engineering* " , 27(4), 1071-1092,2020.
- [55] Zhang, W., Li, H., Li, Y., Liu, H., Chen, Y., & Ding," Application of deep learning algorithms in geotechnical engineering: a short critical review " , *Artificial Intelligence Review*, 54(8), 5633-5673 , 2021.
- [56] Asgari Taghanaki, S., Abhishek, K., Cohen, J. P., Cohen-Adad, J., & Hamarneh, " Deep semantic segmentation of natural and medical images: a review. *Artificial Intelligence Review* " , 54(1), 137-178 ,2021.
- [57] Zhang, J., Zhan, Z. H., Lin, Y., Chen, N., Gong, Y. J., Zhong, J. H., ... & Shi, "Evolutionary computation meets machine learning: A survey", *IEEE Computational Intelligence Magazine*, 6(4), 68-75,2011.
- [58] Groschner, C. K., Choi, C., & Scott, "Machine learning pipeline for segmentation and defect identification from high-resolution transmission electron microscopy data " , *Microscopy and Microanalysis*, 27(3), 549-556,2021.
- [59] Bepler, T., Morin, A., Rapp, M., Brasch, J., Shapiro, L., Noble, A. J., & Berger," Positive-unlabeled convolutional neural networks for particle picking in cryo-electron micrographs " , *Nature methods*, 16(11), 1153-1160,2019.

- [60] Yeganeh, H., & Wang," Objective quality assessment of tone-mapped images ", IEEE Transactions on Image processing, 22(2), 657-667, 2012.
- [61] L. S. Chow and R. Paramesran, " Biomedical Signal Processing and Control Review of medical image quality assessment " , *Biomed. Signal Process. Control*, vol. 27, pp. 145–154, 2016.
- [62] Kumcu, A., Bombeke, K., Platiša, L., Jovanov, L., Van Looy, J., & Philips," Performance of four subjective video quality assessment protocols and impact of different rating preprocessing and analysis methods " , IEEE Journal of Selected Topics in Signal Processing, 11(1), 48-63 , 2016.
- [63] Min, X., Zhai, G., Zhou, J., Farias, M. C., & Bovik," Study of subjective and objective quality assessment of audio-visual signals " , IEEE Transactions on Image Processing, 29, 6054-6068 , 2020.
- [64] Chen, Y., Zhang, M., Bai, M., & Chen," Improving the signal-to-noise ratio of seismological datasets by unsupervised machine learning " , Seismological Research Letters, 90(4), 1552-1564 ,2019.
- [65] Cui, Z., Gan, Z., Tang, G., Liu, F., & Zhu," Image signature based mean square error for image quality assessment " , Chinese Journal of Electronics, 24(4), 755-760 , 2015.
- [66] I. Amaya, R. Correa, and R. Correa, " Reconstructing design parameters of a rectangular resonator via peak signal-to-noise ratio and global optimization algorithms algorithms " , *Inverse Probl. Sci. Eng.*, vol. 5977, no. January, pp. 1–23, 2017.
- [67] Yang, H., Ciftci, U., & Yin, " Facial expression recognition by de-expression residue learning " , In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2168-2177) , 2018.

- [68] A. C. Sundermann, K. E. Hartmann, S. H. Jones, E. S. Torstenson, and D. R. Velez, "Annals of Epidemiology Validation of maternal recall of early pregnancy medication exposure using prospective diary data", *Ann. Epidemiol.*, 2016.
- [69] Anwyl-Irvine, A., Dalmaijer, E. S., Hodges, N., & Evershed, " Realistic precision and accuracy of online experiment platforms, web browsers, and devices ", *Behavior research methods*, 53(4), 1407-1425 , 2021.
- [70] Chicco, D., & Jurman, " The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation " , *BMC genomics*, 21(1), 1-13 , 2020.
- [71] Kang, D., Park, S., & Paik, " SdBAN: Salient object detection using bilateral attention network with dice coefficient loss " , *IEEE Access*, 8, 104357-104370 , 2020.
- [72] Ahmed, F., Tarlow, D., & Batra, " Optimizing expected intersection-over-union with candidate-constrained CRFs " , In *Proceedings of the IEEE international conference on computer vision* (pp. 1850-1858) , 2015.
- [73] N. Miura, A. Nagasaka, and T. Miyatake, " Feature extraction of finger vein patterns based on iterative line tracking and its application to personal identification " , *Syst. Comput. Japan*, vol. 35, no. 7, pp. 61–71, 2004.

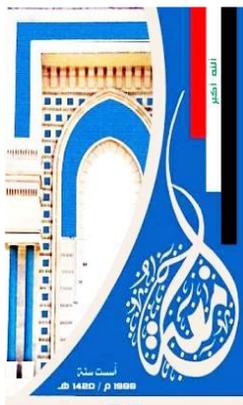
الخلاصة

اليوم ، يكتسب تحديد وريد الإصبع شعبية كحل محتمل لإطار عمل تحديد القياسات الحيوية. تتضمن معظم مجموعات البيانات على الكثير من الضوضاء والأشياء غير المرغوب فيها في الخلفية ، مما يجعل من الصعب تحديد واستخراج وريد الإصبع. عتبة الصورة يبدو أنها طريقة مناسبة لمرحلة بناء القناع الثنائي التي يتم استخدامها في بعض الصور الدقيقة للكشف عن وريد الإصبع واستخراجها. في هذه الأطروحة ، تم اقتراح نموذجين لاستخراج نمط أوردة الأصابع ، نموذج غير خاضع للإشراف ونموذج خاضع للإشراف. يتضمن النموذج الأول غير الخاضع للإشراف منهجية تستند إلى مرحلتين تطويريتين.

تستخدم المرحلة الأولى تطويرًا جديدًا لخوارزمية (K-mean) ، والتي تعتمد على الاختلاف بين شدة البكسل للنقاط. يتم تحديد النقاط المركزية وفقًا لعملية تطبيع بكسلات الصورة وجعلها تشتمل على خمسة مستويات من الشدة ، حيث يتم تجميع وحدات البكسل المتشابهة في مجموعة واحدة تسمى الكتلة ، حيث أن الغرض من هذه العملية هو عزل منطقة الإصبع عن الخلفية التي تحتوي على تفاصيل غير مهمة.

تتضمن مرحلة التطوير الثانية تقسيم الصورة إلى كتل أو بلوكات بأبعاد متساوية وتحديد مسارات نمط عروق الأصابع باستخدام تقنيات GPE و GPO للحصول على شكل دقيق وواضح لنمط الوعاء الدموي ، حيث أن الغرض من العملية هو إنشاء نمط الوريد لكل صورة في قاعدة البيانات لغرض تدريبها في النموذج الثاني. النموذج الثاني الخاضع للإشراف هو المكان الذي يتم فيه تقييم الأقنعة المتوقعة واختيارها بناءً على تشابه أقنعة النمط الثنائي المتوقعة. في النهاية ، يتم تحديد الهوية البشرية والتحقق من صحتها بناءً على أقنعة نمط ثنائية نقية متوقعة باستخدام إطار عمل جديد في CNN يسمى U-NET. يحقق النموذج الأول غير الخاضع للإشراف دقة تصل إلى ٩٨,٦٪ وهي أعلى بكثير من مناهج التعلم القياسية الأخرى غير الخاضعة للإشراف مثل خوارزميات Fuzzy C-Means (FCM) وخوارزميات K-mean .

تظهر نتائج النموذج الثاني الخاضع للإشراف أن هذا النموذج قادر على إنشاء نمط وريد إصبع نظيف مهم جدًا وقادر على التعرف بنجاح على صور عروق الأصابع من نفس الفرد من خلال تحقيق متوسط دقة بنسبة ٩٥ ٪ من درجات التشابه بين اصبعين لنفس الشخص.



جمهورية العراق

وزارة التعليم العالي والبحث العلمي

جامعة ديالى / كلية العلوم / قسم علوم الحاسوب

التجزئة الدلالية العميقة المعتمدة على الشبكة العصبية الالتفافية لاستخراج النمط الثنائي الخاص بوريد الاصبع من الصورة المنخفضة الجودة .

الاطروحة مقدمة الى قسم علوم الحاسوب / كلية العلوم/ جامعة ديالى وهي جزء من
متطلبات نيل درجة الماجستير في علوم الحاسوب

الباحث

علي صلاح حميد

بإشراف

أ.م.د عادل عبد الوهاب العزاوي

استاذ مساعد