Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 6.199

IJCSMC, Vol. 9, Issue. 2, February 2020, pg.21 – 37

Analysis of Procedures used to build an Optimal Fingerprint Recognition System

Dr. Amjad Hindi; Dr. Majed Omar Dwairi; Prof. Ziad Alqadi
Albalqa Applied University, Faculty of Engineering Technology, Jordan, Amman
amjadhindi@bau.edu.jo; majeddw@gmail.com; natalia_maw@yahoo.com

Abstract: Fingerprint recognition is one of the most widely use operation, due the big number of applications requiring this operation, so seeking a good method of fingerprint features extraction is a vital task. In this paper we will investigate three method of image features extraction: minutiae, RCSLP and c_clustering methods, we will analyze the performances of these methods, and we will for fixing the features when the fingerprint rotated. The obtained features will be used to feed a selected type with a selected architecture ANN in order to show which type of ANN gives the best results of fingerprint recognition.
Keywords: Minutiae, ridge ending, bifurcation, RCSLBP, c_mean clustering, ANN, FFANN, CFANN, EANN.

1- Introduction

Fingerprints can be used in all kinds of ways [1], [2], [3]:

- Provide biometric security (for example, to control access to secure areas or systems)
- Identify victims of unknown memory loss and death (e.g., major disaster victims, if their fingerprints are on file)
- Undertake basic checks (including applications for government employment, defense security clearance, hidden weapons permits, etc.) [4], [5].

Fingerprints are particularly important in criminal justice. Investigators and analysts can compare unknown Fingerprints collected from the crime scene to those known to victims, witnesses and potential suspects to help with criminal cases [6], [7], [8].

The process of fingerprint identification and recognition is a vital issue due to the wide range of applications required this process; this process can be implemented as shown in figure 1 applying the following steps [9]:

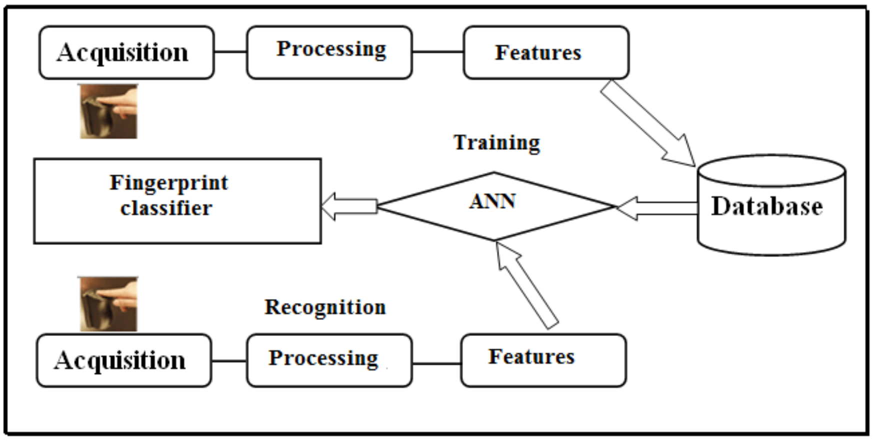


Figure 1: Process of fingerprint identification

- Fingerprint image acquisition.
- Image processing (Enhancement, thinning, binarization).
- Image features extraction.
- Saving the features in the features database.
- Crating artificial neural network (ANN) to be used as a recognition tool.
- Using the features database to train ANN.
- Once ANN was trained, we can use it to calculate fingerprint classifier, by running ANN with the selected fingerprint features.

In this research paper, we will analyze some methods used to extract fingerprint features, the extracted features will be used to train various types of ANN, and the obtained experimental results will be discussed in order to select the optimal method of features extraction and the optimal ANN which will be used as a recognizer.

2- Analysis of fingerprints features extraction methods

2-1 Minutiae method of features extraction

Human fingerprint image consists of several types of minutiae, the types, number and the coordinates of these minutiae are different from one person to another, thus any extracted information about minutiae can be used as a fingerprint identifier to recognize the fingerprint and the associated person, and figure 2 shows a fingerprint example and the detected minutiae [1..5].

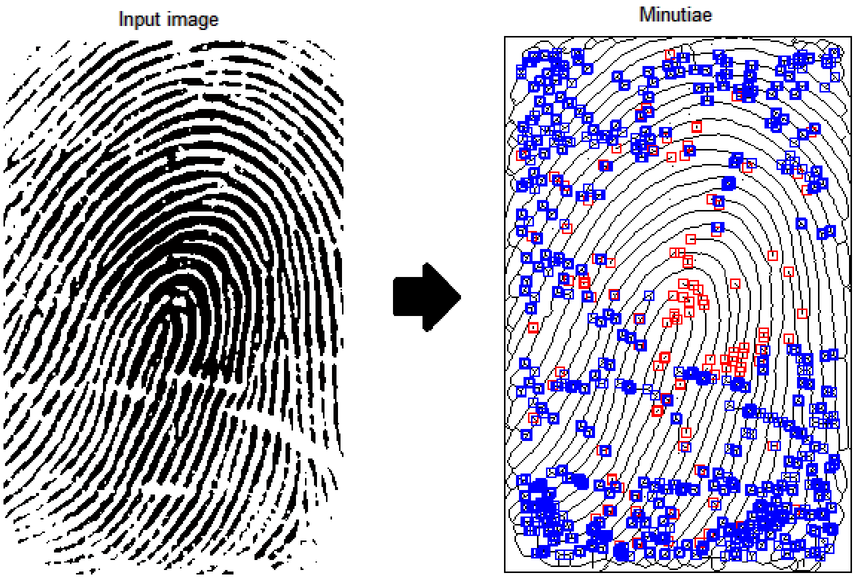


Figure 2: Fingerprint and the detected minutiae.

Figure 3 shows various types of fingerprint minutiae:

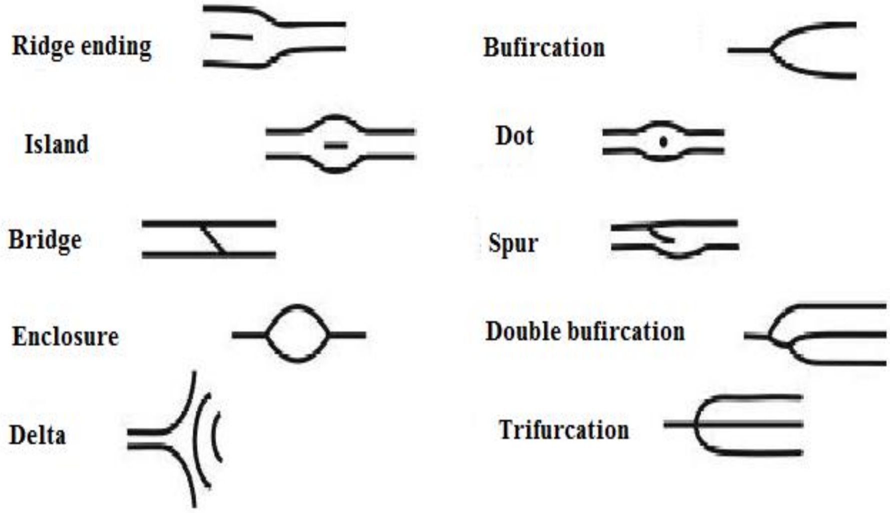


Figure 3: Various types of minutiae.

In our research paper we will focus on the most popular types shown in figure 4:

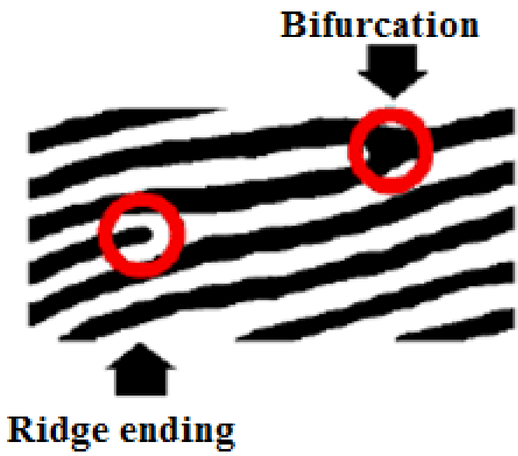


Figure 4: Ridge ending and bifurcation types of minutiae.

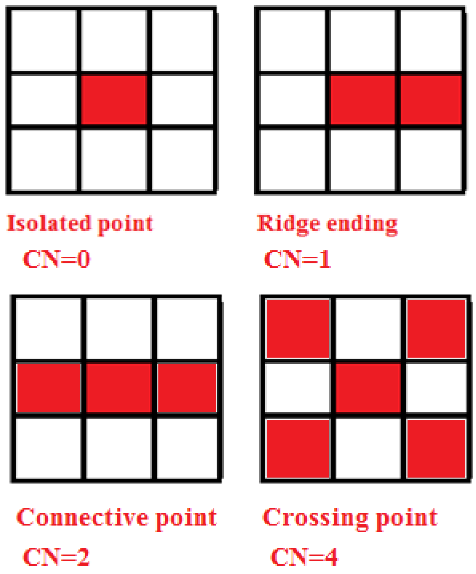
Each Minutia can be detected based on the pixel 8 neighbors (in the image binary version) by calculating the classification number (CN) as shown in figure 5 [6..9]:

P_4	P_3	P_2
P_5	P	P_1
P_6	P_7	P_8

$$CN = \frac{1}{2}\sum_{i=1}^{8} |P_i - P_{i+1}|$$

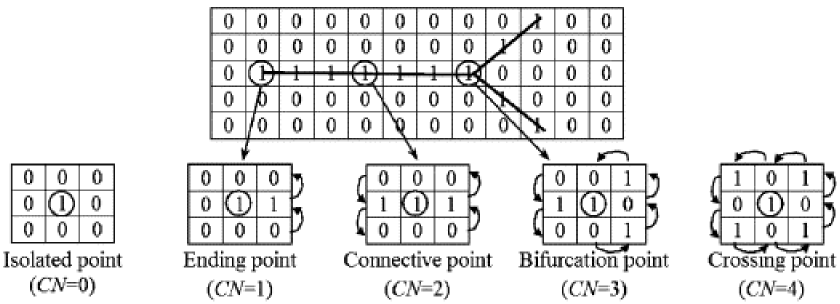
Figure 5: Pixel CN calculation

Based on the calculated value of CN for a selected pixel we can determine the minutiae type as shown in figure 6[34-52]:



A: Minutiae types

CN	*Property*
0	Isolated point
1	Ending point
2	Connective point
3	Bifurcation point
4	Crossing point



B: CN calculation

Figure 6: Minutiae types and CN calculation

Based on the detected minutiae types we can use each of the following information to create a fingerprint features (identifier) (see figure 7):

- Ridge ending coordinates.
- Bifurcation coordinates
- Distance between the ridges coordinates.
- Distance between bifurcations coordinate.
- Number of ridge ending points
- Number of bifurcation points.
- Number of each minutiae type.

To detect fingerprint minutiae we have to follow the following steps:

- Acquire fingerprint image.
- If the image is color then convert it to gray image.
- Get the binary version of the image by applying morphological thinning.

- Select a window of 3 by 3 pixels.
- For each pixel, using the window calculate CN.
- Define minutiae information depending on CN value and add 1 to the minutiae type.

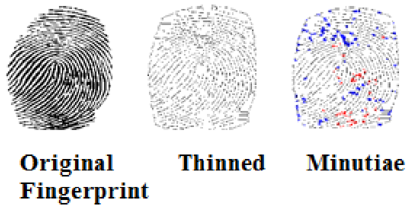


Figure 7: Minutiae extraction

2-2 Reduced local binary pattern (RLBP) method to extract features

Local binary pattern (LBP) method of image features extraction is also one of the popular methods recently used in image features extraction. All versions of this method are based on LBP method, which operates by calculating LBP operator for each pixel as shown in figure 8[10].

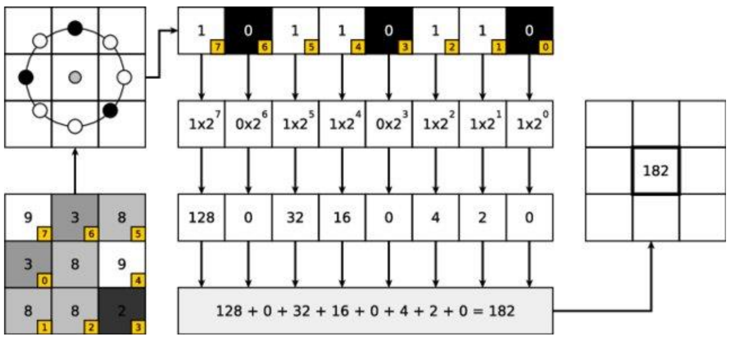


Figure 8: Pixel LBP operator calculation.

LBP method produces a features array of 256 elements [13..15], each value of them points to the repetition of the array index (0 to 255), to reduce the number of features we can use a reduced center symmetric local binary pattern method (RCSLBP) which reduces the number of features to 4(repetition of zero to repetition of 3), this method acts as shown in figures 9 and 10[11], [12],[47]:

P7	**P4**	P8
P3	P	**P1**
P6	**P2**	P5
A0=(P1+P2-P3-P4>=P) (0 or 1)		
A1=(P5+P6-P7-P8>=P (0 or 1)		
RCSLBP=A0+A1*2		

Figure 9: RCSLBP calculation

100	**70**	**50**
80	**90**	**100**
200	**150**	**30**
A0=(100+150-80-70>=90) =1		
A1=(30+200-100-50>=90) =0		
RCSLBP=A0+A1*2 =1		

Figure 10: RCSLBP calculation example

To get fingerprint RCSLBP features we have to follow the following steps:

- Acquire the image of the fingerprint.
- If the image is color then convert it to gray image.
- Initialize features of 4 elements to zeros (repetition array).
- For each pixel in the gray image calculate RCSLBP operator, add 1 the repetition of this operator.

2-3 Using C_mean clustering method to create fingerprint features

C_mean clustering method is a method of grouping input data set into groups (cluster), and it is very useful to generate data set features[16], [17], [18].

Color or gray images can be used here as an input data set [43] .[44], but to increase the method efficiency by decreasing the features extraction time we can use the image histogram (reduced input data set with 256 values) [34-52] as an input data set, and here we can also fix the features even if we rotate the fingerprint image, because the histogram for all rotated version of the fingerprint image remain the same, figure 11 shows a fingerprint image and histogram [19], [20].

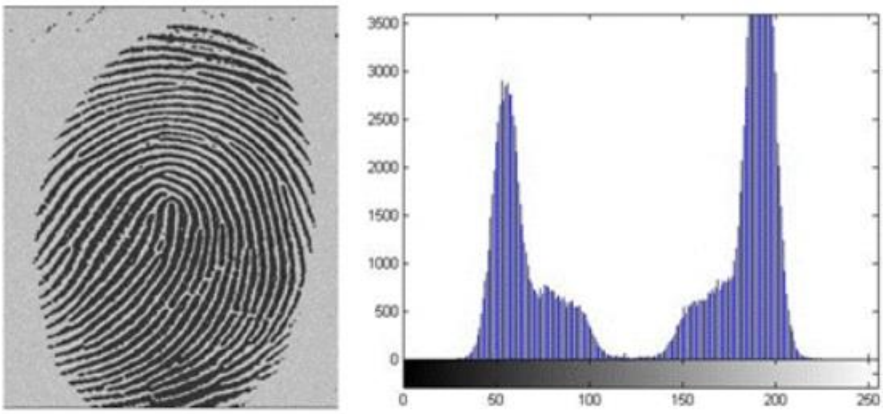


Figure 11: Fingerprint image and histogram

Here to enhance the fingerprint image we can use LBP image histogram as an input data set for clustering as shown in figure 12:

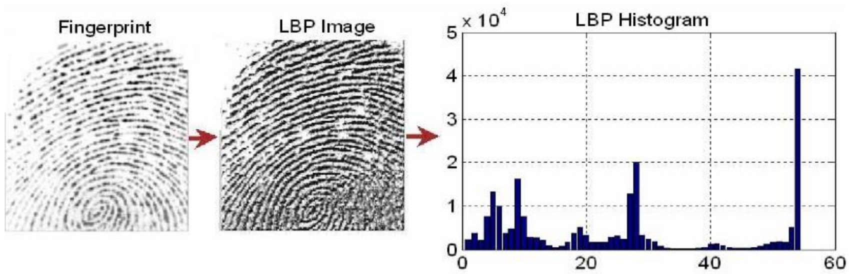


Figure 12: LBP histogram

C_mean clustering method can be implemented applying the following steps:

- Initialization

Here we have to prepare the input data set, for image clustering we have to calculate the image histogram to be used as an input data set, then the number of clusters must be defined, a centroid of each cluster must be given an initial value.

- While centroids are changing repeat the following passes:
    - Find cluster distance by taking the absolute value of subtraction the data item value from the centroid value.

- The data item then will belong to the cluster with minimum distant.
- Find each new centroid by averaging the data items that belong to the cluster.

Here we can use one of the following parameter as an identifier (features) for the selected data set:

- The cluster centroids.
- Number of points (data items) in each cluster.
- Within cluster sums (WCS), each of them will equal the summation of points belong to the cluster.

Worked example:

The input data set (X): 1, 3, 4, 11, 13, 16, 20, 25

Number of clusters 2

Centers: C1=1, C2=4

Tables 1 and 2 show the results of calculation

Table 1: Pass 1 and pass 2

X	Pass 1				Pass 2			
	Dist.1	Dist.2	Nearest cluster	New centers	Dist.1	Dist.2	Nearest cluster	New centers
1	**0**	**3**	**1**	**C1=1** **C2=13.14**	0	12.1400	1	**C1=2.66** **C2=17**
3	**2**	**1**	**2**		2	10.1400	1	
4	**3**	**0**	**2**		3	9.1400	1	
11	**10**	**6**	**2**		**10**	**2.1400**	**2**	
13	**12**	**9**	**2**		**12**	**0.1400**	**2**	
16	**15**	**12**	**2**		**15**	**2.8600**	**2**	
20	**19**	**15**	**2**		**19**	**6.8600**	**2**	
25	**24**	**21**	**2**		**24**	**1.8600**	**2**	

Table 3: Pass 3

X	Pass 3			
	Dist.1	Dist.2	Nearest cluster	New centers
1	**1.6600**	**16**	**1**	**C1=2.66** **C2=17**
3	**0.3400**	**14**	**1**	
4	**1.3400**	**13**	**1**	
11	**8.3400**	**6**	**2**	
13	**10.3400**	**4**	**2**	
16	**13.3400**	**1**	**2**	
20	**17.3400**	**3**	**2**	
25	**22.3400**	**8**	**2**	
Features				
Centers	**2.66; 17**			
Number of pints	**3; 5**			
WCS	**8; 85**			

2-4 Artificial neural network as a recognition tool

Artificial neural network (ANN) is a powerful computational model, which can be easily used to recognize any object using the object features as an input to feed ANN[21], [22].

ANN consists of a set of fully connected neurons, each neuron is connected to a set of input through a defined weights, this neuron performs two functions in order to generate an output (see figure 13) [23], [24],[25]:

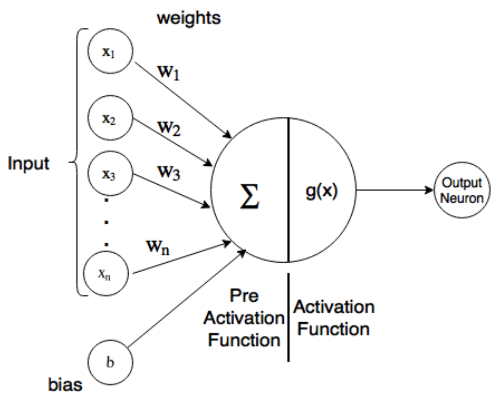


Figure 13: Neuron operations

- Summation of the products of inputs and associated with the weights.
- According to the selected activation function compute the output.

ANN neurons are organized into layer (input layer, output layer, and one or more optional hidden layers) as shown in figure 14[26]. [27].

The outputs are to be calculated from the input layer toward the output layer, the error between the calculated output and the target output will be calculated, if the error is not acceptable then the weight will be updated starting from the output layer toward the input layer, this process will be continued while the error is not acceptable, each round of calculations is called a training cycle (epoch), figure 15 shows how to do ANN calculations.

To use ANN as a recognition tool, we have to pay our attention on the optimality of this tool, which characterize by small amount of time required for training and recognition, and closed to 100% recognition ratio (recognition without error) [28], [29], [30].

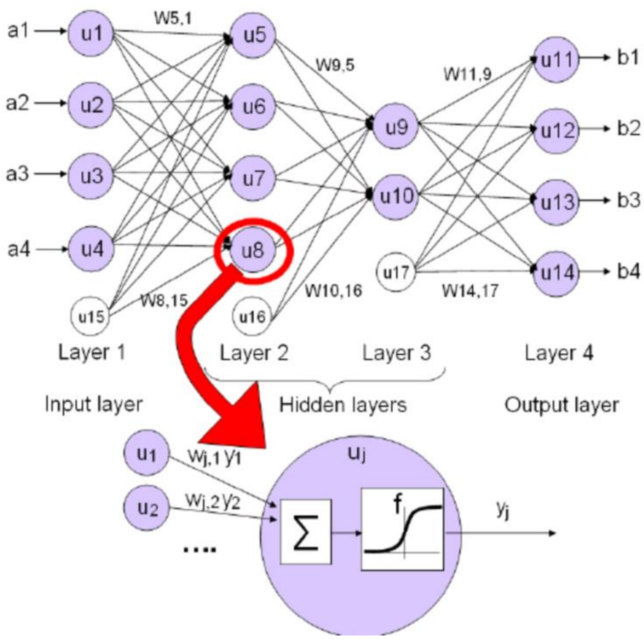


Figure 14: Multilayer ANN

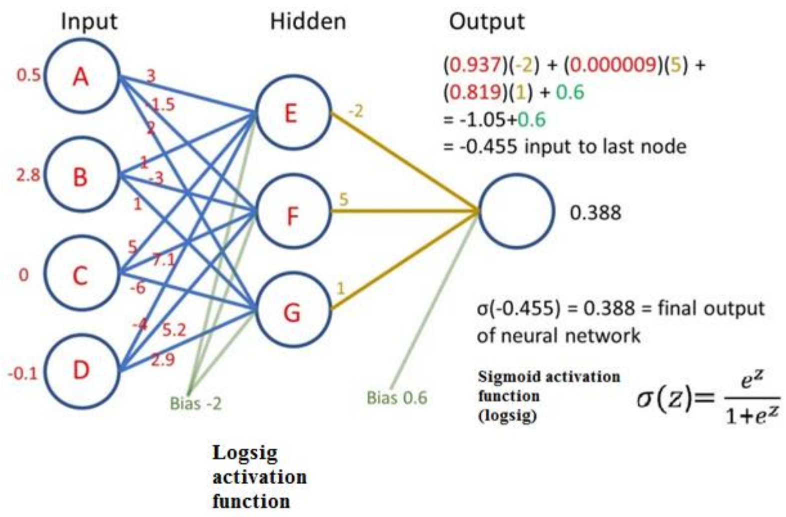


Figure 15: ANN calculations

To reach the ANN optimality we have to consider the following:

- Good selection of the fingerprints features.
- Selecting appropriate type of ANN.
- Selection ANN architecture: number of layers, number of neurons in each layer, and selecting the activation function for each layer.
- Initializing the weights to zeros.

Different types of ANN can be used as a recognition tool, and here in this paper we will focus on feed forward ANN (FFANN), cascade feed forward ANN (CFANN) and Elman ANN (EANN).

In FFANN the weights are connected as shown in figure 16-a from the previous layer to the next layer, in CFANN the weight are also connected from the layer to the next layer(see figure 16-b), while in EANN the outputs calculated in the hidden layer are used to feed the previous layer as shown in figure 17[31], [32].

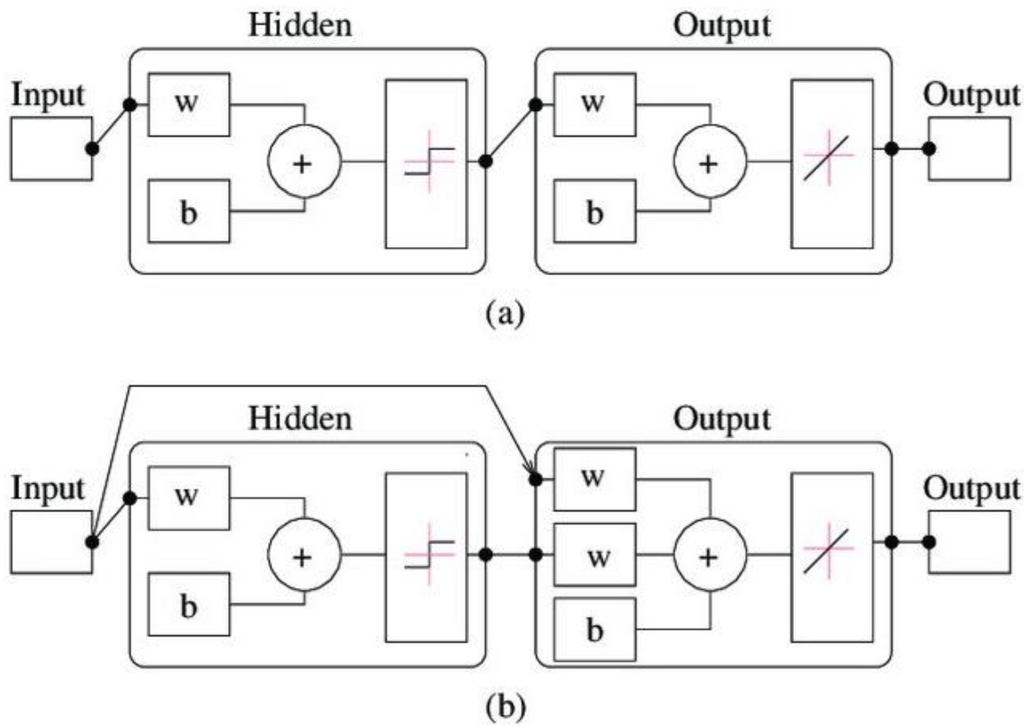


Figure 16: a: FFANN b: CFANN

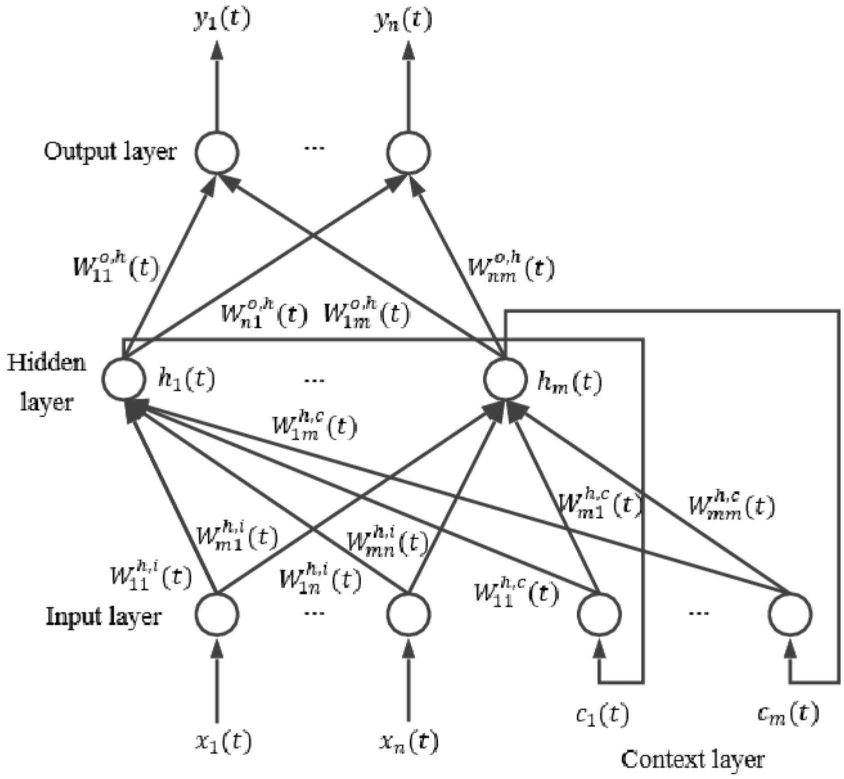


Figure 17: EANN

3- Implementation and experimental results

3-1 Minutiae extraction

Different fingerprints were selected, a matlab code was written and implemented to obtain fingerprint identifier based on CN calculation, table 4 show the obtained results for this phase of implementation:

Table 4: Minutiae calculations and extractions

				Features (Identifier)				
Fingerprint	Resolution	Size	Number of different minutiae	**Number of ridge ending**	**Number of bifurcation**	**DisX**	**DisY**	Extraction time
1	600x385x1	231000	6	476	1503	445.1	555.3	0.379000
2	600x342x1	205200	7	1122	3423	288.5	176.1	0.359000
3	456x331x1	150936	6	**77**	**253**	**435.9082**	**369.2655**	0.196000
4	490x392x3	576240	7	143	3244	221.5	253.4	0.297000
5	490x490x3	720300	6	49	3936	584.8	480.5	0.360000
6	170x127x3	64770	7	115	841	59.0762	40.8534	0.044000
7	225x224x3	151200	7	156	752	115.3776	190.2130	0.079000
8	250x201x3	150750	5	25	393	119.0042	176.2044	0.070000
9	265x190x3	151050	7	129	2154	175	159.1	0.094000
10	255x198x3	151470	6	30	377	147.4110	166.1716	0.072000
11	177x285x3	151335	5	96	478	128.1601	132.4009	0.084000
12	186x272x3	151776	7	359	510	115.2606	94.1116	0.073000
13	1133x784 x1	888272	7	581	3015	660.7	880.6	1.085000
14	237x213x3	151443	8	187	786	159.9062	176.7993	0.075000
Average								**0.2334**

From table 4 we can see the following facts:

- The fingerprint features are formed from the number of ridge ending points, the number of bifurcation points, Euclidean distanceof the ridge ending points between x and y coordinates, Euclidean distanceof the bifurcation points between x and y coordinates, Euclidean distance[20] can be calculated applying the following formula[33]:

$$d(\mathbf{p},\mathbf{q}) = d(\mathbf{q},\mathbf{p}) = \sqrt{(q_1-p_1)^2+(q_2-p_2)^2+\cdots+(q_n-p_n)^2}$$
$$= \sqrt{\sum_{i=1}^{n}(q_i-p_i)^2}.$$

  Where p and q are the points coordinates.
- The extracted features for a selected fingerprint are unique, thus we can use them as an identifier to recognize the fingerprint.
- The extracted time is significantly small.
- The main disadvantage of minutiae method of features extraction is that the features are sensitive to any rotation of the fingerprint.

Table 5 shows the features for a reshaped fingerprint shown in figure 18.

Figure 18: Reshaped fingerprint 3

Table 5: Reshaped fingerprint features

Original fingerprint features	Reshaped fingerprint features
77.0000	1456
253.0000	7297
435.9082	237.9
369.2655	101.1

3-2 RCSLBP method

The same previously used different fingerprints were treated, a matlab code was written and implemented to obtain fingerprint identifier based on CSLBP operator calculation, table 6 show the obtained results for this phase of implementation:

Table 6: CSLBP method results

Fingerprint							time
1	600x385x1	231000	163234	14026	22148	29626	0.314000
2	600x342x1	205200	150884	13878	21086	17472	0.297000
3	456x331x1	150936	**113304**	**6769**	**12115**	**17178**	0.219000
4	490x392x3	576240	179392	2761	3883	4284	0.319000
5	490x490x3	720300	174219	19695	20686	23544	0.343000
6	170x127x3	64770	11356	3443	3703	2498	0.036000
7	225x224x3	151200	37765	2743	3440	5558	0.099000

8	250x201x3	150750	34618	2803	4607	7324	0.089000
9	265x190x3	151050	35082	3635	5204	5523	0.095000
10	255x198x3	151470	38892	2131	3031	5534	0.082000
11	177x285x3	151335	31835	6083	4996	6611	0.091000
12	186x272x3	151776	34344	4163	4806	6367	0.086000
13	1133x784 x1	888272	720683	32076	53399	78284	1.272000
14	237x213x3	151443	37502	2951	4812	4320	0.085000
Average							**0.2448**

From table 6 we can see the following facts:

- The fingerprint features are formed from the number of CSLBP operator values repetitions.
- The extracted features for a selected fingerprint are unique, thus we can use them as an identifier to recognize the fingerprint.
- The extracted time is significantly small.
- The main disadvantage of CSLBP method of features extraction is that the features are sensitive to any rotation of the fingerprint.

Table 7 shows the features for a reshaped fingerprint shown in figure 18.

Table 7: Reshaped fingerprint 3 features (CSLBP method)

Original fingerprint features	Reshaped fingerprint features
113304	106344
6769	20415
12115	16416
17178	6191

3-3 C_mean clustering method

The same previously used different fingerprints were treated, a matlab code was written and implemented to obtain fingerprint identifier based on C_mean clustering method calculation, table 8 show the obtained results for this phase of implementation:

Table 8: C_ mean clustering method results

Fingerprint							time
1	600x385x1	231000	252	184	98	9	1.063000
2	600x342x1	205200	253	181	94	5	0.949000
3	456x331x1	150936	**252**	**186**	**98**	**9**	0.711000
4	490x392x3	576240	254	205	150	83	2.658000
5	490x490x3	720300	250	181	97	10	3.302000
6	170x127x3	64770	251	178	97	12	0.303000
7	225x224x3	151200	253	243	170	85	0.699000
8	250x201x3	150750	251	187	103	8	0.711000
9	265x190x3	151050	251	187	109	23	0.708000
10	255x198x3	151470	253	188	105	13	0.708000
11	177x285x3	151335	250	185	102	9	0.720000
12	186x272x3	151776	251	179	93	18	0.699000
13	1133x784 x1	888272	252	191	102	16	4.045000
14	237x213x3	151443	250	229	127	9	0.706000
Average							**1.2844**

From table 8 we can see the following facts:

- The fingerprint features are formed from the centroid or within clusters sums values.
- The extracted features for a selected fingerprint are unique, thus we can use them as an identifier to recognize the fingerprint.
- The extracted time is bigger.

- This method solves the disadvantage of the previous method by keeping the features without any changes even if we rotate the fingerprint image. Table 9 shows the features for a reshaped fingerprint shown in figure 18.

Table 9: Reshaped fingerprint 3 features (C_mean clustering method)

Original fingerprint features	Reshaped fingerprint features
252	**252**
186	**186**
98	**98**
9	**9**

3-4 Fingerprint recognition

Input databases with fingerprints extracted features were created, each database is 2D matrix with 4 rows and number of columns equals the number of processed fingerprint, these databases were used to feed ANN.

To use ANN as a recognizer, we follow the following phases:

1) Training phase:
   This phase can be implemented applying the following steps:
- Get the saved features database to be used as an input data set for ANN.
- Create ANN: here we have to select ANN type, number of layers, and number of neurons in each layer, define the activation function for each layer.
- Initialize ANN weights to zero.
- Set the error to zero.
- Select a suitable number of training cycles.
- Train ANN.
- If the obtained error equal zero, then save ANN to be used in the next phase, else modify ANN architecture and repeat training.

2) Running ANN as a recognizer: this phase can be implemented applying the following steps:
- Load ANN.
- Get the fingerprint features.
- Run ANN with the features to get a classifier or identifier which will point to the recognized fingerprint or person.

The extracted earlier fingerprints by the three methods were used with three similar CFANN, with input layer of 4 neurons, output layer of one neuron; table 10 shows the results of executing these ANNs, while table 11 shows ANN parameters:

Table 10: Using CFANN

Fingerprint	Minutia	RCSLBP	C_mean
	Recognition time(s)(Including features extraction time)	Recognition time(s)(Including features extraction time)	Recognition time(s)(Including features extraction time)
1	0.4670	0.3430	1.1590
2	0.4040	0.2860	1.1630
3	0.2840	0.2890	1.1580
4	0.3700	0.3030	1.1540
5	0.4580	0.2980	1.1540
6	0.1440	0.2890	1.1560
7	0.1800	0.2860	1.1570
8	0.1690	0.2890	1.1600
9	0.1930	0.2870	1.1570
10	0.1720	0.2880	1.1590
11	0.1790	0.2850	1.1600

12	0.1740	0.2880	1.1570
13	1.1930	0.2950	1.1560
14	0.1770	0.2860	1.1590
Average	**0.3260**	**0.2937**	**1.1578**

Table 11: CFANN parameters

Parameter	C-mean clustering features	Minutia extraction features	RCSLBP features
ANN size(byte)	28661	28661	28661
Layers	2	2	2
Input layer neurons	4	4	4
Output layer neurons	1	1	1
Training cycles	70	40	307
Error	0	0	0
Recognition ratio	100%	100%	100%
Input layer activation function	logsig	logsig	logsig
Output layer activation function	Linear	Linear	Linear

The three types of features gave us a 100% recognition ratio but with different amount of recognition time.

Tables12, 13 shows the results of using various types off ANN based on C_mean clustering features and from these tables we can see that the optimal ANN can be achieved by using CFANN.

Table 12: Using various type of ANN with fixed architectures

Parameter	CFANN	FFANN	EANN
ANN size(byte)	28661	26653	29023
Layers	2	2	2
Input layer neurons	4	4	4
Output layer neurons	1	1	1
Training cycles	70	49	10000
Error	0	0	57
Recognition ratio	100%	100%	43%
Input layer activation function	logsig	logsig	logsig
Output layer activation function	Linear	Linear	Linear

Table 13: Using various type of ANN with different architectures

Parameter	CFANN	FFANN	EANN
ANN size(byte)	28661	33963	58603
Layers	2	3	4
Input layer neurons	4	4	5
Output layer neurons	1	1	1
Hidden layer neurons	0	8	16, 8
Training cycles	70	30	18000
Error	0	0	17
Recognition ratio	100%	100%	83%

Input layer activation function	logsig	logsig	Tansig
Output layer activation function	Linear	Linear	Linear
Hidden layer activation function	-	logsig	Tansig, Tansig

Conclusion

Three methods of fingerprints features extraction were investigated and analyzed; the obtained experimental results showed that these methods are suitable for creating a unique fingerprint identifier to be used for human recognition or identification, but it was shown that C_mean method of clustering has a good advantage by fixing the features and keeping them without changes for the same fingerprint with any position. For recognition purposes it was shown that the beast type of ANN to be used as a recognition tool is CFANN, which provides better performance and accuracy comparing with other types of ANN.

References

[1]. Jude Hemanth & Valentina Emilia Balas, ed. (2018). *Biologically Rationalized Computing Techniques For Image Processing Applications*. Springer. p. 116. ISBN 9783319613161.

[2]. Ronald F. Becker & Aric W. Dutelle (2018). *Criminal Investigation*. Jones & Bartlett Learning. p. 133. ISBN 9781284082852.

[3]. Fingerprinting of UK school kids causes outcry Archived August 10, 2017, at the Wayback Machine, *The Register*, and July 22, 2002 (in English).

[4]. Fingerprint Source Book: manual of development techniques, published 26 March 2013 Archived February 11, 2017, at the Wayback Machine retrieved on February 9, 2017; see also Max M. Houck (Ed.): Forensic Fingerprints, London 2016, p. 21, 50 er.

[5]. Ziad A.A. Alqadi, Musbah Aqel, Ibrahiem M. M. El Emary, Fingerprint Matching Algorithm Based on Ridge Path Map, European Journal of Scientific Research ISSN 1450-216X Vol.15 No.3 (2006), pp. 344-351.

[6]. Chengsheng Yuan, Xinting Li, Q. M. Jonathan Wu, Jin Li and Xingming Sun, Fingerprint Livens Detection from Different Fingerprint Materials Using Convolutional Neural Network and Principal Component Analysis, CMC, vol.53, no.4, pp.357-372, 2017.

[7]. Shankar Bhausaheb Nikam, Texture and Wavelet-Based Spoof Fingerprint Detection for Fingerprint Biometric Systems, 2008 First International Conference on Emerging Trends in Engineering and Technology, DOI**:** 10.1109/ICETET.2008.134.

[8]. Gragnaniello, D.; Poggi, G.; Sansone, C.; Verdoliva, L. (2015): Verdoliva, Local contrast phase descriptor for fingerprint liveness detection. Pattern Recognition, vol. 48, no. 4, pp. 1050-1058.

[9]. Wang, B.; GU, X.; Ma, L.; San, Y. (2016): Temperature Error Correction Based on BP Neural Network in Meteorological Wireless Sensor Network. International Conference on Cloud Computing and Security, pp. 117-132.

[10]. Esa Prakasa, Texture Feature Extraction by Using Local Binary Pattern, NKOM, Vol. 9, No. 2, November 2015: 45-48.

[11]. Ziad A. Alqadi, Majed O. Al-Dwairi, Amjad A. Abu Jazar and Rushdi Abu Zneit, Optimized True- RGB color Image Processing, World Applied Sciences Journal 8 (10): 1175-1182, ISSN 1818-4952, 2010.

[12]. A. A. Moustafa, Z. A. Alqadi, Color Image Reconstruction Using A New R'G'I Model, journal of Computer Science, Vol.5, No. 4, pp. 250-254, 2009.

[13]. Jamil Al Azzeh, Hussein Alhatamleh, Ziad A. Alqadi, Mohammad Khalil Abuzalata, Creating a Color Map to be used to Convert a Gray Image to Color Image; International Journal of Computer Applications , November 2016,Volume 153,Issue 2.

[14]. AlQaisi Aws and AlTarawneh Mokhled and Alqadi Ziad A. and Sharadqah Ahmad A,Analysis of Color Image Features Extraction using Texture Methods, TELKOMNIKA, volume 17, number 3, pages1220—1225, year 2019.

[15]. Al-Azzeh J. , Zahran B. , Alqadi Ziad, Ayyoub B. and Abu-Zaher, M., A novel zero-error method to create a secret tag for an image, Journal of Theoretical and Applied Information Technology, volume 96,number13,pages 4081-4091, year 2018.

[16]. Moustafa, A.A., Alqadi, Ziad A., A practical approach of selecting the edge detector parameters to achieve a good edge map of the gray image, Journal of Computer Science, volume 5,number 5,pages 355-362,year 2009.
[17]. Naseem Asad ; Ismail Shayeb ; Qazem Jaber ; Belal Ayyoub ; Ziad Alqadi ; Ahmad Sharadqh, Creating a Stable and Fixed Features Array for Digital Color Image, IJCSMC, Vol. 8, Issue. 8, August 2019, pg.50 – 62.
[18]. Ahmad Sharadqh Jamil Al-Azzeh , Rashad Rasras , Ziad Alqadi , Belal Ayyoub, Adaptation of matlab K-means clustering function to create Color Image Features, International Journal of Research in Advanced Engineering and Technology, volume 5, issue 2, pages 10-18, year 2019.
[19]. Akram A Moustafa, Ziad A Alqadi, Eyad A Shahroury, Performance evaluation of artificial neural networks for spatial data analysis, Contemporary engineering sciences, v 4, issue 4, pp 149-163, 2011.
[20]. Khaled M Matrouk, Alasha'ary, Abdullah I. Al-Hasanat, Ziad A. Al-Qadi, Hasan M. Al-Shalabi, Investigation and Analysis of ANN Parameters, European Journal of Scientific Research, v 121, issue 2, pp 217-225, 2014.
[21]. Jamil Al-Azzeh, Ziad Alqadi, Mohammed Abuzalata, Performance Analysis of Artificial Neural Networks used for Color Image Recognition and Retrieving, IJCSMC, Vol. 8, Issue. 2, pp 20 – 33, 2019.
[22]. Khaled M Matrouk, Alasha'ary, Abdullah I. Al-Hasanat, Ziad A. Al-Qadi, Hasan M. Al-Shalabi, Experimental Investigation of Training Algorithms used in Back propagation Artificial Neural Networks to Apply Curve Fitting, European Journal of Scientific Research, v 121, issue 4, pp 328-335, 2014.
[23]. Prof. Mohammed Abu Zalata Dr. Ghazi. M. Qaryouti , Dr.Saleh Khawatreh, Prof. Ziad A.A. Alqadi, Optimal Color Image Recognition System (OCIRS), International Journal of Advanced Computer Science and Technology., v 7, issue 1, pp 91-99, 2017.
[24]. Eng. Sameh S. Salma Prof. Ziad A. AlQad, Eng. Ahmad S. AlOthma , Eng. Mahmoud O. Alleddaw ,Eng. Mohammad Ali Al-Hiar , Eng. Osama T. Ghaza, Investigation of ANN Used to Solve MLR Problem, IJCSMC, v 8, issue 5, 2019.
[25]. Jamil Al-azzeh Belal Ayyoub, Ahmad Sharadqh, Ziad Alqadi, Simulink based RNN models to solve LPM, International Journal of Research in Advanced Engineering and Technology, v 5, issue 1, pp 49-55, 2019.
[26]. Mohammed Abuzalata Jamil Al-Azzeh, Ziad Alqadi, Performance Analysis of Artificial Neural Networks used for Color Image Recognition and Retrieving, IJCSMC, v 8, issue 2, pp 20-33, 2019.
[27]. Sumit Goyal and Gyandera Kumar Goyal, Cascade and Feed forward Back propagation Artificial Neural Network Models For Prediction of Sensory Quality of Instant Coffee Flavored Sterilized Drink, Canadian Journal on Artificial Intelligence, Machine Learning and Pattern Recognition Vol. 2, No. 6, 2011.
[28]. Scott E. Fahlman and Christian Lebiere, the Cascade Correlation Learning Architecture. D. S. Touretzky (ed.), Advances in Neural Information Processing Systems 2, pages 524–532, 1990.
[29]. Thomas R. Schultz, Francois Rivest, Laszl ´ o Egri, ´ Jean-Philippe Thivierge, and Fred´ eric Dandurand, ´ Could Knowledge-based Neural Learning Be Useful in Developmental Robotics?. The Case of KBCC. International Journal of Humanoid Robotics Vol. 4, No. 2, pages 245–279, 2007.
[30]. Chong, S., Rui, S., Jie, L.: Temperature drift modeling of MEMS gyroscope based on Genetic-Elman neural network. In: Mechanical Systems and Signal Processing, 2016, vol. 72, p. 897-905.
[31]. Yongchun, L.: Application of Elman neural network in short-term load forecasting. In: Artificial Intelligence and Computational Intelligence (AICI), 2010 International Conference on, IEEE, 2010. p. 141-144.
[32]. Storn, R., Price, K.: Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. In: Journal of global optimization, 1997, vol. 11, no 4, p. 341-359.
[33]. Ay, Nihat; Amari, Shun-ichi (2015). "A Novel Approach to Canonical Divergences within Information Geometry" (PDF). *Entropy*. **17** (12): 8111–8129. Bibcode: 2015 Entrp...17.8111A. Doi: 10.3390/e17127866.
[34]. Ziad A. AlQadi, A Highly Secure and Accurate Method for RGB Image Encryption, IJCSMC, v. 9, issue 1, pp. 12-21, 2020.

[35]. Belal Zahran Rashad J. Rasras, Ziad Alqadi, Mutaz Rasmi Abu Sara, Developing new Multilevel security algorithm for data encryption-decryption (MLS_ED), International Journal of Advanced Trends in Computer Science and Engineering, v. 8, issue 6, pp. 3228-3235, 2019.

[36]. J Al-Azzeh M Abuzalata, Z Alqadi, Modified Inverse LSB Method for Highly Secure Message Hiding, International Journal of Computer Science and Mobile Computing, v. 8, issue 2. pp. 93-103, 2019.

[37]. Bilal Zahran Belal Ayyoub, Jihad Nader, Ziad Al-Qadi, Suggested Method to Create Color Image Features Victor, Journal of Engineering and Applied Sciences, v. 14, issue 1, pp. 2203-2207, 2019.

[38]. A Waheeb, Ziad AlQadi, Gray image reconstruction, Eur. J. Sci. Res, v. 17, pp. 167-173, 2009.

[39]. Ziad Alqadi, Bilal Zahran, Qazem Jaber, Belal Ayyoub, Jamil Al-Azzeh, Enhancing the Capacity of LSB Method by Introducing LSB2Z Method, IJCSMC, v. 8. Issue 3, pp. 76 – 90, 2019.

[40]. Ziad A Alqadi, Akram A Moustafa, Majed Alduari, True Color Image Enhancement Using Morphological Operations, International Review on Computers & Software, v. 4, issue 5, pp. 557-563, 2009.

[41]. Musbah J Aqel, Ziad ALQadi, Ammar Ahmed Abdullah, RGB Color Image Encryption-Decryption Using Image Segmentation and Matrix Multiplication, International Journal of Engineering and Technology, v. 7, issue 3.13, pp. 104-107, 2018.

[42]. Ziad AlQadi, Hussein M Elsayyed, Window Averaging Method to Create a Feature Victor for RGB Color Image, International Journal of Computer Science and Mobile Computing, v. 6, issue 2, pp. 60-66, 2017.

[43]. R Abu Zneit, Ziad AlQadi, M Abu Zalata, A Methodology to Create a Fingerprint for RGB Color Image, International Journal of Computer Science and Mobile Computing, v. 16, issue 1, pp. 205-212, 2017.

[44]. Rashad J Rasras, Mohammed Abuzalata, Ziad Alqadi, Jamil Al-Azzeh, Qazem Jaber, Comparative Analysis of Color Image Encryption-Decryption Methods Based on Matrix Manipulation, International Journal of Computer Science and Mobile Computing, v. 8, issue 3, pp. 14-26, 2019.

[45]. Jamil Al-azzeh Ahmad Sharadqh, Belal Ayyoub, Ziad Alqadi, Experimental investigation of method used to remove salt and pepper noise from digital color image, International Journal of Research in Advanced Engineering and Technology, v. 5, issue 1, pp. 23-31, 2019.

[46]. RA Zneit, Ziad Alqadi, Dr Mohammad Abu Zalata, Procedural analysis of RGB color image objects, International Journal of Computer Science and Mobile Computing ,v. 6, issue 1, pp. 197-204, 2017.

[47]. Ziad Alqadi, A Novel Methodology for Repairing a Torn Image, International Journal on Communications Antenna and Propagation, v. 4, issue 1, pp. 366-372, 2011.

[48]. Akram A Moustafa, Ziad A Alqadi, Reconstructed color image segmentation, Proceedings of the World Congress on Engineering and Computer Science 2009 Vol II WCECS 2009, October 20-22, 2009, San Francisco, USA.

[49]. M. Qaryouti, Dr. Saleh Khawatreh, Prof. Ziad AA Alqadi, Prof. Mohammed Abu Zalata, Optimal Color Image Recognition System (OCIRS), International Journal of Advanced Computer Science and Technology, v. 7, issue 1,pp. 91-99, 2017.

[50]. Mutaz Rasmi Abu Sara Rashad J. Rasras, Ziad A. AlQadi, A Methodology Based on Steganography and Cryptography to Protect Highly Secure Messages, Engineering, Technology & Applied Science Research, v. 9, issue 1, pp. 3681-3684, 2019.

[51]. Jihad Nadir, Ashraf Abu Ein, Ziad Alqadi, A Technique to Encrypt-decrypt Stereo Wave File, International Journal of Computer and Information Technology, v. 5, issue 5, pp. 465-470, 2016.

[52]. Haitham Alasha'ary, Abdullah Al-Hasanat, Khaled Matrouk, Ziad Al-Qadi, Hasan Al-Shalabi, A Novel Digital Filter for Enhancing Dark Gray Images, European Journal of Scientific Research, Vol.122 No.1, 2014, pp.99-106.

