International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988,Volume 5 (2013) pp. 696-703
© MIR Labs, www.mirlabs.net/ijcisim/index.html

Features Extraction of Arabic Calligraphy using extended Triangle Model for Digital Jawi Paleography Analysis

Mohd Sanusi Azmi[1], Khairuddin Omar[2], Mohammad Faidzul Nasrudin[2], Azah Kamilah Muda[1], Azizi Abdullah[2] and Khadijah Wan Mohd Ghazali[1]

[1]Faculty of Information Communication and Technology,
Universiti Teknikal Malaysia Melaka, 76100 Malaysia
sanusi@utem.edu.my
azah@utem.edu.my
khadijah@utem.edu.my

[2]Faculty of Information Science and Technology,
Universiti Kebangsaan Malaysia, 43650 Malaysia
ko@ftsm.ukm.my
mfn@ftsm.ukm.my
azizi@ftsm.ukm.my

***Abstract*: The style of writing or calligraphy applied in ancient manuscripts gives useful information to paleographers. The information helps paleographer to identify date, writer, number of writers, place of origin, and the originality of manuscripts. This information is known as features. The features from characters, tangent value, dominant background and also Grey-Level Co-occurrence Matrix (GLCM) have been used in this field of research. A novel technique was proposed for digital Jawi Paleography. Jawi is the original Malay writing based on Arabic characters. The technique proposed models triangles on images and extracts features from them. The features are used for classification. In this paper, new features for the Triangle Model are proposed. Also, the implementation of four zones is reported. The number of features has been extended from 12 to 45. For validation of proposed algorithm, 60,000:20,000 training and testing data from Hoda digit dataset has been prepared, selected randomly for 10 rounds of testing. For further verification, two Supervised Machine Learning (SML) and three Unsupervised Machine Learning (UML) algorithms were experimented. These experiments were conducted using a new Arabic calligraphy dataset that was set up from 1,225 Arabic letters taken from calligraphy books. From the data, SML experiments were conducted with the ratio of 807:408 for training and testing. Whereas for the UML, three classifiers were tested on 30 images of Arabic calligraphy dataset. Results from the tests prove that the Triangle Model technique can successfully be used in digital paleography of Jawi characters.**

***Keywords*:** Paleography, Calligraphy, Jawi, Arabic, Triangle Model, Features Extraction, Hoda dataset

I. Introduction

Calligraphies and illuminations in ancient manuscripts were used to express the aesthetic quality of the manuscripts, and the status of the writers and the kingdom they represent. From another perspective, analysis on the calligraphies and illuminations can also be used to collect important information of unknown manuscript i.e. the date it was written, place of origin, number of writers and its originality [1–4]. This research is named Paleography. The paleographers work by finding similarities between known and unknown manuscripts. From here, categorization is made based on specific range of time [4], [5]. The similarities and non-similarities are known as features. The features are extracted using either local or global approaches [6] and further categorized into local and global features as shown in Figure 1.

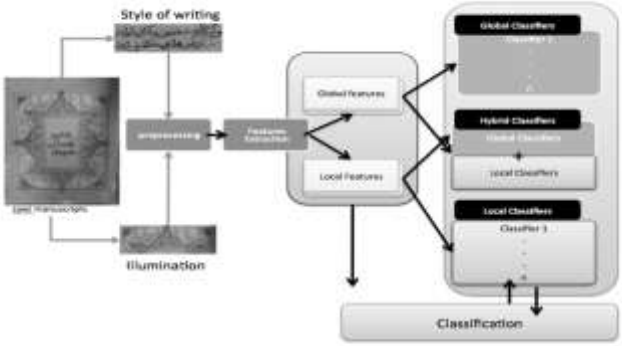


Figure 1. Framework for Digital Jawi Paleography [6]

The first system for digital paleography named System for Paleography Inspection (SPI) was developed by [7] in 1999 in the History Department, University of Pisa [1], [8]. The domain of the research was inspection of Roman ancient manuscripts. The system has never been used [1], but paved the first steps towards further paleography researches on Roman [8], [9], Hebrew [5] and Arabic/Jawi [3], [6], [10], [11].

[1], [7], [5], [9], [10] used local features while [8] used global features. In SPI, tangent distance and centroid values were extracted from isolated images of ancient documents. The extracted features were then categorized into Dendogram diagram [1]. Another research, done on Hebrew character by [5], was based on the dominant background zones by defining values between each region using Multi-Stage Thresholding. However, the research limited its scope to only one character out of all 22 Hebrew characters, extracted from 14 documents. Out of 280 images, 14 were used as test images, each for 14 classes. This averaged the number of testing for each class to one. Another research that becomes the basis for this research is [10] which proposed features based on values generated by triangles extracted from isolated images. In this research, the triangle-based features used will be expanded into four identified zones. This will be discussed in the following topics.

For the global features, a study was done on an implementation by [8] which was based on Haralick's Features, also known as Grey-Level Co-occurrence Matrix (GLCM). Although [8] claims the GLCM gives very significant result, it required the testing and training images to be in the same dimension due to the multiplication of matrices in GLCM.

In this paper, the Digital Jawi Paleography research is carried out based on the previous works by [3] and [10]. Amendments have been made and tests have been conducted by using standard dataset as well as generated datasets using SML and UML.

II. Pre-processing and Proposed Method

The proposed method is based on further enhancements on [10]. Stages followed in this proposed methods are:

a. Data collection
b. Segmentation of Jawi characters
c. Image categorization
d. Binarization
e. Features Extraction and proposed method

The Features Extraction and proposed method will be detailed here because it is a novel technique for Digital Jawi Paleography that is currently researched in Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, Bangi.

A. Data collection

The data collection stage was divided into two parts. In part one, the standard digit dataset HODA introduced by [12] was used. In the second part, Arabic calligraphy dataset was used.

The HODA digit dataset was chosen as the best dataset for the first part because it consists of Arabic digits and there were no standard dataset for the Roman [9] and Arabic paleography [10]. Besides, the number of classes is also near to the five types of calligraphy used in this research. The dataset consists of ten numeral digits from 0 to 9. HODA dataset has a total of 80,000 images, 60,000 for training and 20,000 for testing.

With the huge number of data for training and testing, HODA dataset helped in validating the correctness of the proposed algorithm using SML. The correctness of Arabic Calligraphy dataset that we developed could only be validated through matching process using distance methods such as Euclidian, Sorenson and Chebychev. Until now, 1225 Arabic Calligraphy letters have been used. The sources of dataset have been explained in [10]. The number of training images is 817 (≈50%) and 408 (≈50%) for the test. Table 1 below shows the sample of Arabic Calligraphy dataset.

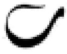
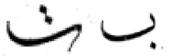
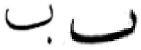
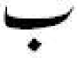
Arabic Calligraphy for character Ba, Ta and Tsa	Type of Arabic Calligraphy
	Diwani
	Riqah
	Thuluth
	Nasakh
	Farisi

Table 1. Sample of calligraphy

B. Segmentation of Jawi Character for the Training and Test images

The segmentation process of characters for Arabic Calligraphy dataset was done manually. This process was based on the works done in [5] and [13]. There have been 1225 images, clustered into five classes, based on the Arabic Calligraphy types normally seen in ancient Malay manuscripts [14]. The classes were i. Nasakh, ii. Thuluth, iii. Riqah, iv. Diwani and v. Farisi. The images can be obtained from the Pattern Recognition Research Group, Universiti Kebangsaan Malaysia.

For the HODA dataset, segmentations were not required because all images were in isolated form. Figure 2 below shows a small number of HODA dataset members after the process of inverting:

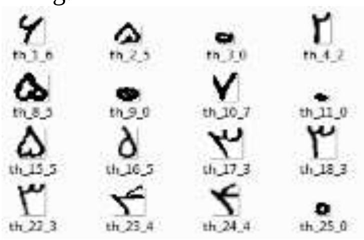


Figure 2. Hoda digit dataset

C. Binarization

The purpose of this process is to remove noise and prepare images for the proposed features extraction.

The training and test data of Arabic characters were automatically binarized using Otsu's method. Previously, the threshold had been fixed as either 127 or 180 as in [3]. In this paper, amendment has been made by applying Otsu's method as in [9]. The method dynamically chooses the discriminant threshold based on the foreground and background of image. So, the threshold value becomes more precise for images from various sources of Jawi ancient manuscripts. On the other hand, the HODA dataset does not need to go through the binarization process because it is already in binary form. The threshold value 127 was used for the Hoda dataset in order to undergo feature extraction.

D. Image Categorization

Based on [10], some of the Arabic and Jawi characters share similar shape but differ in the presence or absence of diacritics and location of diacritics. However, the difference made by the presence of diacritics in the Arabic calligraphy is not important in this research because this research focuses on the Arabic calligraphy type identification, not character recognition. Table 2 shows categorization of some characters. For example, a chosen character (ب) has been compared with other characters. All other characters that have similar shape (ت and ث), were grouped into the same group, group 'A'. Other characters that have nearly similar shape (ف, ق and ن) were grouped in another group related to the first group, group 'a'. Further, another character of different shape was chosen and compared with other remaining characters using the same process and grouped into group 'B' and 'b'. These groups were used in the testing phase using UML approach.

The reason for grouping the characters with similar shape is because some of the calligraphy books provide Arabic characters with no diacritics [10]. Thus, the training and test images in this paper follow the approach presented in the books [10].

Group	Character	Sample Images	
		Before	After
A	Ba		
	Ta		
	tsa		
a	Fa		
	Kaf		
	nun		
	nun		
B	sin		
	shin		
b	Sad		
	Dhad		

Table 2. Arabic calligraphy with similar and nearly similar shape

E. Feature Extraction and proposed method

The feature extraction was carried out based on [10]. Six features were selected from the triangle model proposed in [10]. However, the model has been improved by dividing each image into four zones as shown in figure 3 below.

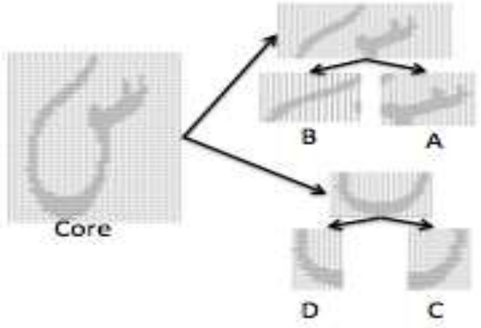


Figure 3. Segregation of isolated characters to four parts.

In figure 3, the image labeled as *Core* was divided into four zones labeled as "*A", "B", "C"* and "*D"* as suggested in [10] for future researches. The comparison between the results of [10] and those of this research is discussed in the Experimental Result topic. The following points present the steps taken for this stage.

i. Extracting three important points.

After the binarization process was completed, the images were ready for the process of extracting three important coordinates [10], [15]. The coordinates extracted are explained in table 3.

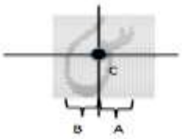
Point	Label	Connected points	Angle	Points Location
Point 1	A	b and c	A	
Point 2	C	a and b	C	
Point 3	B	a and c	B	

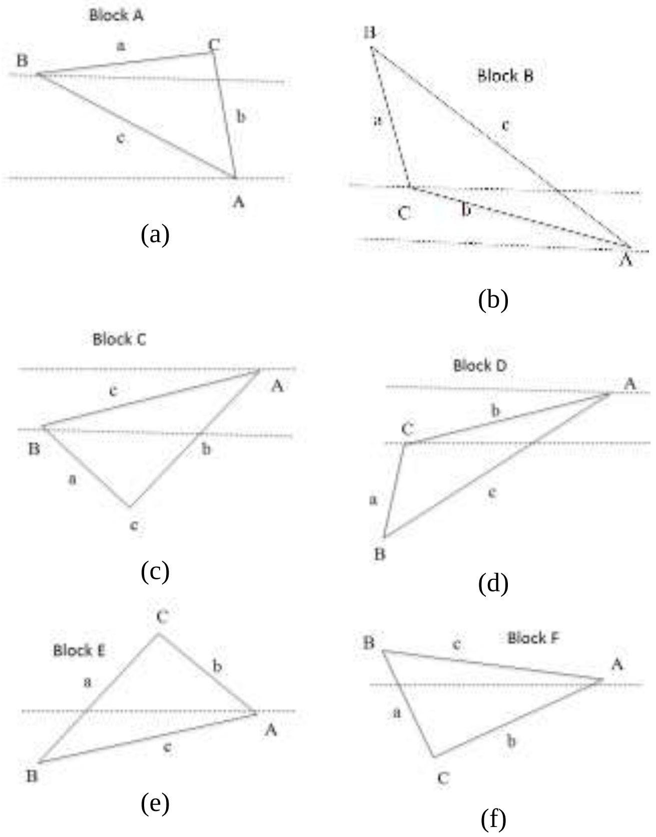
Table 3. Location of triangle points

Table 3 shows the location of points for the core triangle. Centroid is labeled as “C” in the image. Point “A” and “B” were taken from the centroid from the right and left of the centroid. These points were used to form a triangle.

In figure 6 shows the main triangle with four zones. The marks “x” in the image are the location of point 1, 2 and 3.

ii. Generating Triangles

The triangle model that was used in modeling the coordinates into triangles was the scalene triangle with exceptions for the triangles that were not in the scalene form. Figure 4 below shows the possible triangles or possible points of triangles for this research.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 4. Possible triangle blocks

iii. Extracting sub-points from four zones

After the Core features were extracted, the images were divided into four zones. The centroid from “Core” was used to divide the image into four zones. The features in zone “A” were extracted with the same criteria as the main triangle. Figure 5below shows the image of zone “A”. The centroid of each zone was the mean of the particular zone and contributed to point C for the particular zones. Point A was generated from right of the zone into the zone’s centroid whereas point B was from the left to the zone’s centroid. The location of points in zone A is as shown in figure 5 below.

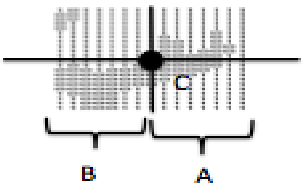
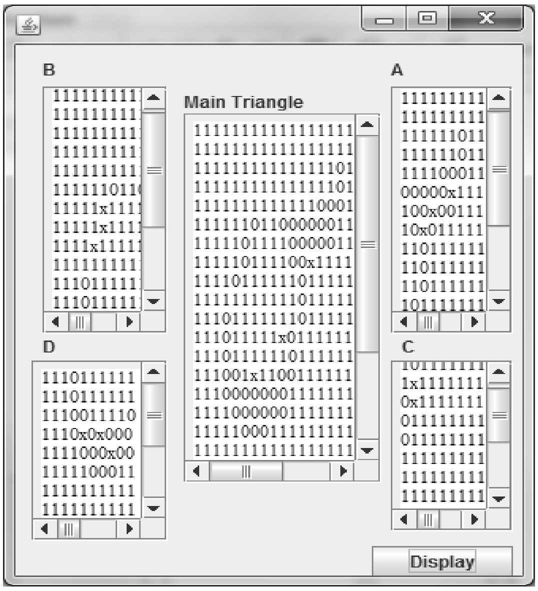


Figure 5. Three points selection for zone "A"

In figure 5 above, the point A is from zone’s centroid to the width of image. Point B is from centroid of the main triangle to the zone’s centroid. Figure 6 below shows the extracted image of four zones. Only the zones with three points will be processed for the features.



B

```
111111111
111111111
111111111
111111111
111111111
1111110110
11111x1111
11111x1111
1111x11111
111111111
111011111
111011111
```

Main Triangle

```
1111111111111111
1111111111111111
1111111111111101
1111111111111101
1111111111110001
1111110110000011
1111101111000011
111110111100x1111
1111011111011111
1111111111011111
1110111111011111
111011111x0111111
1110111111011111
111001x1100111111
11100000001111111
11110000001111111
11111000111111111
1111111111111111
```

A

```
11111111
11111111
11111011
11111011
11100011
00000x111
100x00111
10x011111
110111111
110111111
110111111
101111111
```

D

```
1110111111
1110111111
1110011110
1110x0x000
1111000x00
1111100011
1111111111
1111111111
```

C

```
101111111
1x1111111
0x1111111
011111111
011111111
111111111
111111111
111111111
```

Figure 6. Core triangle with 4 zones

iv. Features extraction and Proposed Method

Nine features were extracted from the triangles of the zones. Six of them were based on [10] as shown in table 4, whereas three more are the contribution of this research, as shown in table 5. Comparisons made to identify best features for classification are discussed later in this paper.

The details about the features in [10] were explained in table 4. The features that are checked with ✓ were used for zones “Core” and “A”.

No.	Feature Name	Features for classification	Description
1	a	X	Length from B(x, y) to C(x, y). Used for calculating ratio a/b, c/a and angles of triangle
2	b	X	Length from A(x, y) to C(x, y). Used for calculating ratio a/b, b/c and angles of triangle
3	c	X	Length from A(x, y) to B(x, y). Used for calculating ration b/c, c/a and angles of triangle.

4	c/a	✓	Ratio c to a
5	a/b	✓	Ratio a to b
6	b/c	✓	Ratio b to c
7	A	✓	Angle of A
8	B	✓	Angle of B
9	C	✓	Angle of C

Table 4. Features from triangles as in [10]

An important contribution of this research is the proposal of additional features calculated from the values of the features described by [10] in table 4 above. First, the length of each side of triangle was calculated using the Pythagorean Theorem. Figure 7 below shows how the side a, b and c were calculated.

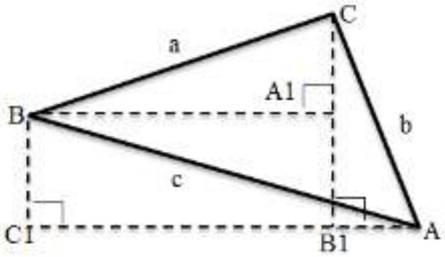


Figure 7. Calculating triangle's side

$$a^2+b^2=c^2 \quad (1)$$

$$\therefore$$

$$a=\sqrt{(A1(y)-C(y))^2+(A1(x)+B1(x))^2} \quad (2)$$

$$b=\sqrt{(B1(y)-B(y))^2+(A(x)+B1(x))^2} \quad (3)$$

$$a=\sqrt{(C1(y)-C(y))^2+(A(x)+C1(x))^2} \quad (4)$$

Then, angles of A, B and C were calculated by using the formula below.

$$A=\arccos\frac{b^2+c^2-a^2}{2ab} \quad (5)$$

$$B=\arccos\frac{a^2+c^2-b^2}{2ac} \quad (6)$$

$$C=\arccos\frac{a^2+b^2-c^2}{2ab} \quad (7)$$

The angles of A, B and C were scaled by dividing them with 180. This process was taken to reduce the range of values for each feature. Hence, the gradients of each point can be calculated, and these become the new features proposed by this research. Table 5 below describes the gradients of each point as the new feature.

No.	Feature Name	Features for classification	Description
1	GraBA	✓	Gradient of B and A
2	GraBC	✓	Gradient of B and C
3	GraCA	✓	Gradient of C and A

Table 5. Extended Features proposed from triangle

Based on table 5, the gradients are computed based on coordinates A, B and C. The formulae below show how the gradients were calculated.

$$GraBC=\frac{B(y)-C(y)}{B(x)-C(x)} \quad (8)$$

$$GraBA=\frac{B(y)-A(y)}{B(x)-A(x)} \quad (9)$$

$$GraCA=\frac{B(y)-A(y)}{B(x)-A(x)} \quad (10)$$

III. Experimental Result

There are two types of testing that have been conducted. The types of testing are (UML) and (SML).

At the first stage, the proposed features and algorithm were validated using HODA digit dataset. There were 60,000 images for training and 20000 for training. All the images were randomly selected for ten sets of training and testing. The mean and standard deviation values presented shows the accuracy of the proposed algorithm.

Two types of classifiers were used in the SML. The classifiers were Multi-layer Perceptron (MLP) and Random Forest (RF). The result of each classification process is as shown in table 6 and 7 below.

No. of run	Feature Core with Zone A (%)	Proposed feature with c/a, a/b, b/c	Proposed Features with [HIS]
1	60.17	70.635	72.905
2	60.715	70.555	72.405
3	60.63	70.365	73.075
4	60.73	69.635	72.89
5	61.495	71.495	73.17
6	58.72	70.305	72.69
7	60.305	68.185	71.175
8	60.29	71.205	71.82
9	60.65	70.33	72.995
10	60.75	69.81	71.61
mean	60.41167	70.30111	72.56944
Standard deviation	0.708737	0.916634	0.698324

Table 6. Multi-layer perceptron (learning rate 0.4)

Table 6 above shows the tests conducted using 60,000 train and 20,000 test sets. The approach proposed in [10] gives accuracy of 60.41167%. With the proposed features, the accuracy is increased to 70.30111 without angles of A, B and C. When all features in [10] and this research were used, the accuracy result was increased to 72.56944%.

No. of run	**Feature Core with Zone A (%)**	**Proposed features with c/a, a/b, b/c**	**Proposed Features with** [10]
1	69.37	77.095	76.935
2	68.39	77.075	77.525
3	69.385	76.98	77.88
4	68.76	77.275	77.485
5	69.215	77.535	77.395
6	69.005	77.04	77.685
7	68.875	77.605	77.31
8	69.105	76.89	77.59
9	69.295	76.945	77.17
10	69.535	77.105	77.69
Mean	69.04444	77.16	77.44167
Standard Deviation	0.345375	0.243121	0.276426

Table 7. Random Forest

The Random Forest classifier can give a significant result to classification, according to [16]. Thus, the testing was also conducted using this classifier. The results in table 7 shows Random Forest gives better result compared to Multi-layer Perceptron in table 6.

Table 8 shows the result for the proposed features with the implementation of 4 zones as mentioned in [10].

No. of run	**Features without angle of A, B and C**	**Features with angle A, B and C**
1	88.185	89.72
2	89.13	90.175
3	88.76	90.46
4	88.635	90.135
5	88.88	89.295
6	88.125	89.845
7	88.285	90.01
8	88.75	90.295
9	88.285	90.875
10	87.735	90.325
Mean	88.55944	90.09
Standard Deviation	0.421882	0.433263

Table 8. Proposed method using MLP

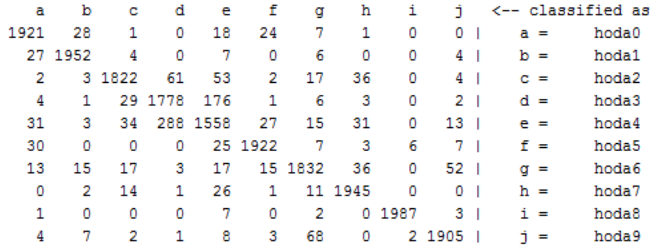
By using Core Triangle with the four zones as in figure 5, the accuracy is increased significantly. The accuracy from 70.30% in table 6 is increased to 88.55944% in table 8 using MLP classifier. When using features in [10] and the proposed features, the average result produced was further increased to 90.09%.

No. of run	**Features without angle of A, B and C**	**Features with angle A, B and C**
1	92.45	92.67
2	92.55	92.935
3	92.46	92.715
4	92.775	92.76
5	92.705	92.615
6	92.55	92.765
7	92.82	93.11
8	92.65	92.65
9	92.43	92.665
10	92.555	93.03
Mean	92.59889	92.765
Standard Deviation	0.137547	0.172692

Table 9. Proposed method with RF

With the Random Forest classifier, the results became better. The classifier gave an average of 92.59889% for the features without angle of A, B and C. Using features in [10] and proposed features, the results further increased to 92.765%, an increase of 0.166111%.

Figure 8 below shows the confusion matrix for the best result in table 9.



```
   a    b    c    d    e    f    g    h    i    j   <-- classified as
1921   28    1    0   18   24    7    1    0    0 |    a =      hoda0
  27 1952    4    0    7    0    6    0    0    4 |    b =      hoda1
   2    3 1822   61   53    2   17   36    0    4 |    c =      hoda2
   4    1   29 1778  176    1    6    3    0    2 |    d =      hoda3
  31    3   34  288 1558   27   15   31    0   13 |    e =      hoda4
  30    0    0    0   25 1922    7    3    6    7 |    f =      hoda5
  13   15   17    3   17   15 1832   36    0   52 |    g =      hoda6
   0    2   14    1   26    1   11 1945    0    0 |    h =      hoda7
   1    0    0    0    7    0    2    0 1987    3 |    i =      hoda8
   4    7    2    1    8    3   68    0    2 1905 |    j =      hoda9
```

Figure 8. Confusion matrix

Based on the result and the confusion matrix in figure 8, significant results have been achieved in the classification using HODA digit dataset. With the result, we believe that the proposed algorithm has been doing well in classification although not in 100% accuracy.

As our next step, we developed our own Arabic calligraphy dataset as described in [10]. In this early stage, the categorization to calligraphy types “Diwani”, “Riqah”, “Nasakh”, “Thuluth” and “Farisi” has not yet been made to individual characters, as this is part of our future work. The result for the SML is as shown in table 10 below.

	MLP (0.4)	**RF**
Features without angle of A, B anc	50.2451%	65.6853
Features with ang A, B and C	53.4314%	64.4608

Table 10. Arabic calligraphy using SML

Table 10 above shows the best result for the MLP as 53.4314% and RF as 65.6853%. We believe that this result can possibly achieve dramatic improvement if categorization is made on individual characters.

Due to this problem, we further used UML to recognize calligraphy based on characters in groups as proposed in table 2. The UML test evaluated the type of Arabic calligraphy based on distance for each character. To produce the best matching results, three UMLs have been chosen. The UMLs are Euclidian, Sorenson and Chebychev.

Table 11 below shows name of images used in testing. 30 images have been used in this stage.

No	**Diwani**	**Farisi**	**Nasakh**
1	ainrhin_D__I .png	KAF_MM__ iru.png	KAFV1_ANG__N.png
2	KAFV2_D__	ainrhin_MM_	ainrhin_ANG__N.png

	e.png	Biru.png	
3	KAF_MM__ iru.png	batasa_farisi_ de.png	ainrhin_ANG__N.png
4	alif_D__Ded g	hajimkha_far _Dede.png	alif_ANG__N.png
5	batasa_MM_ Biru.png	kef_MM__F_ u.png	batasa_ANG__N.png
6	daldzal_D__I e.png	mim_MM__I ru.png	daldzal_MM__N_Biru. png
7	fa_D__Dede.	saddadv2_far _Dede.png	fa_ANG__N.png
8	hajimkha_D_ de.png	sinsyinv3_MI F_Biru.png	hakhajim_ANG__N.pn g
9	kef_MM__D uPG1.png	ya2_MM__F u.png	kefV2_ANG__N.png
10	saddad_D__I .png	thadza_MM_ Biru.png	ya_N__Dede.png

Table 11. Test samples of Arabic Calligraphy for UML

The results for the UML test for type “Diwani”, “Farisi” and “Nasakh” from table 11 above is as shown in table 12, 13c and 14 below.

Euclidian		**Sorenson**		**Chebychev**	
Rank	**Distance**	**Rank**	**Distance**	**Rank**	**Distance**
1	1.623	1	0.18	1	0.536
1	1.12	1	0.075	1	0.5
2	1.263	2	0.134	9	5
1	1.737	1	0.135	1	0.724
1	1.474	2	0.163	7	0.667
1	1.688	1	0.212	2	1
1	1.282	1	0.111	2	0.583
1	1.944	1	0.335	0	1
1	2.694	1	0.363	0	0.8
1	4.035	1	0.28	4	3.241

Table 12. UML for Diwani

Euclidian		**Sorenson**		**Chebychev**	
Rank	**Distance**	**Rank**	**Distance**	**Rank**	**Distance**
3	3.213	2	0.34	1	0.333
0	2.387	0	0.774	0	0.5
1	0.529	1	0.037	1	0.295
1	1.165	1	0.124	2	0.5
6	2.022	5	0.287	9	0.515
1	7.452	0	10.006	3	0.5
3	3.044	10	0.505	3	0.287
1	1.815	1	0.257	1	0.257
6	2.18	6	0.148	0	1.194
3	3.761	1	0.334	4	1.136

Table 13. UML for Farisi

Euclidian		**Sorenson**		**Chebychev**	
Rank	**Distance**	**Rank**	**Distance**	**Rank**	**Distance**
1	1.533	1	0.127	1	0.553
5	2.47	8	0.182	3	0.875667
3	0.876	3	0.876	8	0.182438
1	3.883	2	0.226	4	2
3	2.311	5	0.21	1	0.898772
1	2.596	1	0.148	1	1
7	1.565	8	0.155	6	0.5
2	0.877	1	0.071	3	0.417
2	0.469	2	0.043	2	0.181
1	0.955	1	0.131	2	0.333

Table 14. UML for Nasakh

The summary of the UML tests is shown in table 15 below. Euclidian Distance scored better than Sorenson and Chebychev for both top 1% and top 10%, with accuracy of 56.67%, and 96.67% respectively.

UML	**Top 1 (%)**	**Top 10 (%)**
Euclidian	56.67	96.67
Sorenson	53.33	93.33
Chebychev	30	86.67

Table 15. Summarization of UML for Arabic calligraphy

IV. Conclusion

The implementation of four zones using the proposed features in this research gives significant result improvements to Hoda digit dataset. Results of the ten rounds of testing conducted proved that the algorithm gives balanced results. However, there is a room for improvement for the proposed algorithm. For the classifications of Arabic calligraphy, the dataset need to be thoroughly improved. Calligraphy experts need to be consulted for the grouping and identification of each Arabic character. Also, we found that the SML classifiers are not suitable to be used with smaller number of data for training and testing.

V. Future Improvement

For future improvements, the accuracy of triangle model can be increased by increasing the number of zones. Besides, tests for the invariance of the proposed algorithm need to be conducted. On the Arabic calligraphy dataset, a proper method to group characters that have similar shape across different calligraphy types need to be identified. Finally, to increase the recognition using UML, the Mean Average Precision (MAP) can possibly be applied in the future.

Acknowledgment

This research is sponsored by the Government of Malaysia using the excellent facilities in the Pattern Recognition Group, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia. The authors would like to thank Tuan Haji Hamdan Abdul Rahman for his consultation on Jawi and the Malay language and Ustaz Mohd Asrak Othman from National Association of Calligraphy Malaysia for consultation on Arabic calligraphy and validation of the dataset.

References

[1] A. Ciula, “Digital palaeography : using the digital representation of medieval script to support palaeographic analysis,” vol. 1, no. Spring, pp. 1–31, 2005.

[2] A. T. Gallop, *Beautifying Jawi: Between Calligraphy and Paleography*. Tanjung Malim: Universiti Pendidikan Sultan Idris, 2005.

[3] M. S. Azmi, K. Omar, M. Faidzul, N. Khadijah, and W. Mohd, “Arabic Calligraphy Identification for Digital Jawi Paleography using Triangle Blocks,” in *2011*

International Conference on Electrical Engineering and Informatics, 2011, no. July, pp. 1714–1718.

[4] M. S. Azmi, K. Omar, M. Faidzul nasrudin, A. Kamilah, and A. Abdullah, “Digital Paleography : Using the Digital Representation of Jawi Manuscripts to Support Paleographic Analysis,” in *2011 International Conference on Pattern Analysis and Intelligent Robotics*, 2011, no. June, pp. 71–77.

[5] I. B. Yosef, K. Kedem, I. Dinstein, M. Beit-arie, and E. Engel, “Classification of Hebrew Calligraphic Handwriting Styles : Preliminary Results,” in *Proceedings of the First International Workshop on Document Image Analysis for Libraries (DIAL’04)*, 2004, no. ii, p. 299.

[6] M. S. Azmi, K. Omar, M. Faidzul, N. Azah, K. Muda, and A. Abdullah, “Digital Paleography : Using the Digital Representation of Jawi Manuscripts to Support Paleographic Analysis,” in *International Conference on Pattern Analysis and Intelligent Robotics*, 2011, no. June.

[7] G. Z. Aiolli, F., M. Simi, D. Sona, A. Sperduti, A. Starita, “SPI: A System for Palaeographic Inspection,” vol. 4, pp. 34–38, 1999.

[8] I. Moalla, a. M. Alimi, F. Lebourgeois, and H. Emptoz, “Image Analysis for Palaeography Inspection,” *Second International Conference on Document Image Analysis for Libraries (DIAL’06)*, pp. 303–311, 2006.

[9] A. A. Brink, J. Smit, M. L. Bulacu, and L. R. B. Schomaker, “Writer identification using directional ink-trace width measurements,” *Pattern Recognition*, vol. 45, no. 1, pp. 162–171, 2012.

[10] M. S. Azmi, K. Omar, M. F. Nasrudin, A. K. Muda, and A. Abdullah, “Arabic calligraphy classification using triangle model for Digital Jawi Paleography analysis,” in *2011 11th International Conference on Hybrid Intelligent Systems (HIS)*, 2011, pp. 704–708.

[11] K. Omar, M. S. Azmi, S. N. Syeikh Abdullah, A. Abdullah, and M. F. Nasrudin, “Framework of Jawi Digital Paleography: A Preliminar Work,” in *2nd International Conference on Mathematical Sciences*, 2010, p. 5.

[12] H. Khosravi and E. Kabir, “Introducing a very large dataset of handwritten Farsi digits and a study on their varieties,” *Pattern Recognition Letters*, vol. 28, pp. 1133–1141, 2007.

[13] M. F. Nasrudin, “Pengecaman Tulisan Tangan Jawi Luar Talian Menggunakan Jelmaan Surih,” Universiti Kebangsaan Malaysia.

[14] M. S. Azmi, K. Omar, and A. Abdullah, “Perekayasaan Histogram Orientasi Kecerunan Mengesan Erotan dan Pencongan manuskript Merong Mahawangsa,” *Jurnal Teknologi Maklumat & Multimedia*, vol. 2, pp. 63–79, 2005.

[15] M. S. Azmi, K. Omar, M. Faidzul, N. Khadijah, and W. Mohd, “Arabic Calligraphy Identification for Digital Jawi Paleography using Triangle Blocks,” *Science And Technology*, no. July, 2011.

[16] R. Caruana, “An Empirical Evaluation of Supervised Learning in High Dimensions,” *Communication*, 2008.

Author Biographies

Mohd Sanusi Azmi received BSc. And Msc from Universiti Kebangsaan Malaysia (UKM) in 2000 and 2003. He joined Department of Software Enginering, Universiti Teknikal Malaysia Melaka (UTeM) in 2003. Now, he is currently a senior lecturer at UTeM and also PhD a candidate at Faculty of Information Science and Technology, UKM. His research interests are image processing, feature extraction, supervised and unsupervised learning.

Khairuddin Omar received his B. and Master in Computer Science from Universiti Kebangsaan Malaysia at 1986 and 1989, respectively. He received his Doctor of Philosophy in 2000 From Universiti Putra Malaysia. Currently, he is a Professor in Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia. His research interests include Artificial Intelligence (Pattern Recognition in decision making with uncertainty – Bayesian Reasoning, Neural Networks, Fuzzy Logic, Fuzzy Neural Networks etc.; & Image with applications to Jawi/Arabic Manuscripts, biometric authentication).

Mohammad Faidzul Nasrudin is a senior lecturer at the Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia (UKM) and an active researcher at the Center for Artificial Intelligence Technology (CAIT), UKM. He obtained degree in Computer Information System at the Western Michigan University in Kalamazoo. He received a master’s from UKM and a doctorate in Artificial Intelligence by joining the Malaysia - Imperial College Doctoral Programme. His current research focuses on document analysis and recognition and metaheuristic optimization.

Azah Kamilah Muda received the BSc and MSc n Universiti Teknologi Malaysia (UTM), Malaysia in 7 and 1999 respectively. From 1997 to 2002 she rked as a lecturer at UTM. Since 2002 she has been h Universiti Teknikal Malaysia Melaka as a lecturer in T, UTeM. She obtained her PhD in bio-inspired tern recognition from UTM in 2009. Her research erests are Pattern Recognition, Image Processing, Soft Biologically Inspired Computing, System Identification and Artificial Intelligence.

zi Abdullah studied Computer Science at Universiti bangsaan Malaysia (UKM) and completes the Master Software Engineering at Universiti Malaya, Malaysia. received the PhD degree from Utrecht University, the herlands in 2010. He is currently a senior lecturer in the ool of Computer Science, Faculty of Technology and ormation Science, UKM. His current research focuses the algorithmic aspects and machine learning, ecially supervised learning, computer vision and robotics.

Khadijah W. M. Ghazali received MSc. in Computer Science from Universiti Teknologi Malaysia, Malaysia in August 2007 and Bachelor of Information Technology Universiti Kebangsaan Malaysia in April 2002. She is working in F. Her research interests include mobile learning and IPv6 networks.