Improve Pattern Recognition Performance Based on Fractal Geometry Selection

Thamir R. Saeed
FPGA System Design and Pattern Recognition Group (FSDPRG),
Department of Electrical Engineering, University of Technology
thaalwaily@gmail.com

Abstract

In n-tuple and Hidden Markov Model(HMM) the recognition has been based on the feature selection. The feature selection in n-tuple depends on the number of tuples and its location. While, in HMM the feature has been related to the states. Where, the suitable features selection lead to optimal recognition. In this paper, a novel approach has presented for n-tuple and Hidden Markov model feature selection by using the Sierpiński fractal technique. The memory size and the recalling time taken to get individual classifier response has been reduced by 29.35% while the recognition is advancing the conventional n-tuple by 12.5% and 11.6% with and without frequency of occurrence respectively. In addition, the improvements noted in the HMMF proposed algorithm is 2.19% in recognition side, while it is 60% in complexity reduction. This approach is found to be robust in the presence of noise, where, the n-tupleF has advanced in recognition by 38.27% the conventional n-tuple algorithms, while HMMF has overperformed the n-tupleF by 14.44%. Simulation results show the maximum recognition is 92.3% for n-tupleF for character recognition, and HMMF is 99.98% for face recognition.

Keywords :- pattern recognition, single layer nets, n-tuple nodes, Fractal Geometry, HMM.

الخلاصه

التمييز في طرق ان–صفوف وموديل ماركوف المخفي يبنى على اختيار السمات. والسمات في طريقة ان–صفوف تعتمد على عدد الصفوف وموقعها. بينما في موديل ماركوف المخفي السمات قد ارتبطت بالحالات. حيث ان اختيار السمات المناسبة يؤدي الى افضل تمييز. في هذه الورقة تم تقديم نهجا جديد ل ان–صفوف وموديل ماركوف المخفي لأختيار السمات باستخدام تقنية سيربنسكي الكسورية. حجم الذاكره والوقت المشار اليه الماخوذ لأداء كل مصنف قل بمقدار 29.35% بينما التمييز تقدم عن ان–صفوف التقليدي ب 12.5% مع تكرار الحدث و 11.6% بدون تكرار الحدث. بالاضافة الى التحسن بموديل ماركوف المخفي المقترح كان بمقدار 2.19% في التمييز بينما قل التعقيد بمقدار 60%. هذا التقرب وجد انه قوي بوجود الضوضاء حيث ان التمييز ل ان–صفوف تقدم عن ان–صفوف التقليدي بمقدار 38.27% بينما تقدم موديل ماركوف المخفي بمقدار 14.44%عن التقليدي. نتائج المحاكاة اضهرت اعلى تمييز كان 92.3% ل ان –صفوف لتمييز الحروف و99.98% لمودي ماركوف المخفي لتمييز الوجوه.

الكلمات المفتاحية: تمييز الانماط ، شبكة الطبقة الواحدة، عقد ان–صفوف، الهندسة الكسورية. موديل ماركوف المخفي.

1. Introduction

Pattern recognition (PR) as a field is extremely diverse and has been applied in many areas such as control, image processing, instrumentation, economic forecasting and speech processing (Jim, 1996). Where, the object classifications into class (identifiable categories) after extracting its features of the data, is the goal of the pattern recognition. These data may be numerical, pictorial, textual, linguistic or any combination of these categories (Jim, 1990). The flexibility and reliability in feature extraction represent the most important factors for pattern recognition. While, the PR algorithms must convey the similarities among patterns in the same class, and, must express the differences among patterns in different classes.

Pattern recognition algorithms are often implemented by using two techniques. The solid-state RAM circuits are represented the first one, which is cheap, availability and simple in implementation(Mazin, 2012;Teresa, 1999; Harald, 1993). Also, the RAM-based neural network has some appealing features like accuracy, flexibility, one-shot learning and fast recalls time. However, this technique has been yet not robust for industry applications and needs large memory size(Stonham, 1987 ; Yong, 2015). While the computer vision is represents the second one, where, it plays a vital role in daily life. These ways have found a wide range of applications from information security and processing to video surveillance and Optical Character Recognition (OCR). The latter technique has been found weaknesses in online applications(Hajian, 1983).

Furthermore, the choice of the appropriate algorithm is depended on the intensity of the data. Where, the n-tuple classifier is a RAM-based neural network, which, can be considered of a range of Look-Up Tables (LUTs) that store the weights of architecture(Jim, 1990; George, 1992). The number of rows in each LUT corresponds to the number of possible classes while the columns are determined from the values of the input connections of the LUTs. While, an alternative and less data-intensive approach for when the data is not easily mapped to a dictionary representation is to make use of a hidden Markov model (HMM) to learn typical sequences of character orderings and other inter-letter correlations.

The n-tuple and HMM techniques have addressed the problem of extracting general features in a pattern recognition task. In this context, the n-tuple technique is facing two problems. The first one occurs when the data being discriminated are very similar. The other one regards tuple size; if the tuple size is too large, the system cannot generalize for many situations, but if it is too small, the system is saturated quickly(Bruce, 2012). In addition to these problems the selection of number and locations of tuples is random. In this context, the main problem that can face the HMM is the complexity caused by the object size that wants to be recognized.

Recently, fractal geometries have frequently been applied to many applications such as pattern recognition, texture analysis, segmentation, etc.(Allinson, 1997 ; Mohammed, 2012). It has been found that the application fractals on the classification process are concentrated either on the connection step as in (Marcilio, 1998;Hannan, 2009 ; Bishop, 1994), and for the feature extraction step as in (Patnala, 2009 ; Zhang, 2002).

In this paper, a combination of the two previous techniques will be adopted, and can be categorized as a third technique. Here, the fractal geometry has been used as a mask(ORing or NORing operation with recognized object for fractal and its inverse respectively) for optimize the feature extraction and reduced the complexity caused by the object size that wants to be recognized. It is worth to note that the choice of the fractal geometry type is dependent on the particular application. The proposed algorithm has based on using fractal geometry for treating the problem of location and number of tuples in n-tuple classifier and the complexity of HMM. As well, to address the problem of the memory size and then the recalling times in the classification process, this paper has used the Sierpiński Carpet and triangle geometry as a mask for a machine written letters and face pictures, to improve the classifiers response. The aim of this work is to improve the performance of n-tuple technique and HMM by the novel connection architecture based on fractal geometry mask for selecting useful cell features. Also, improve the performance of HMM within appropriate feature selection.

2. Recognition Algorithms

2.1 The Conventional n-tuple Operation

There are two stages in the operation of the n-tuple networks. First, they are taught appropriate data;this is the learn phase.Then, they are used to classify new data; this is called the test or analysis phase (Bishop, 1994).In general;n-tuple classifier systems operate in a multi-discriminator configuration as shown in Figure 1. where each discriminator contains from several hundred to several thousands of logic nodes (Bruce, 2013 ; Danilo, 2013).

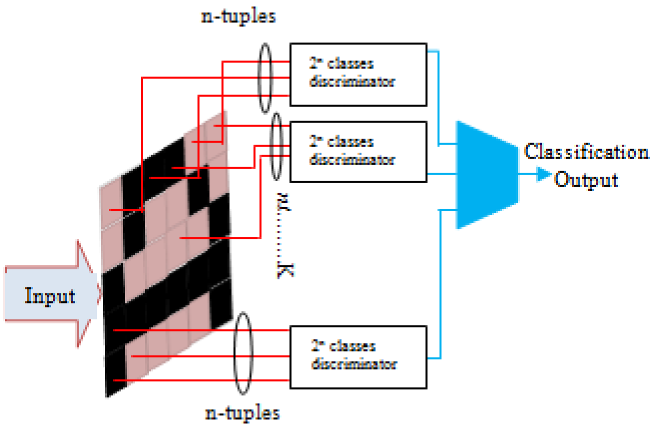


Figure 1. N-tuple K discriminator configuration

All memory sites start with 0 stored before training. For a feature present in any of the patterns of the training set, training consists simply of storing a "1" in the corresponding (addressed) memory site (Paulo, 2010).

Then, the conventional n-tuple operating relation, for deciding the Output Classifier (*OC*) of the unknown input pattern, can be described by;

OC =max[comparator(discriminator output)] (1)

where;

max[comparator(discriminator output)]- the highest value among the contents of RAM for each discriminator.

The discriminator response has been evaluated by calculating the confidence measure, and the success rate. For an input vector, of size *L*, the number of necessary RAM nodes is *R* with connectivity *nt* (width of the address bus (tuple-size)) that covers all inputs of the input vector. This should satisfy:

$R \times nt \geq L$ (2)

A simple confidence measure is the vote difference between the winning class and succeeding class. Two measures of confidence can be used; absolute confidence (*AC*) and relative confidence, (*RC*) as given by Equations (3, 4) (George, 1992 ; Bishop , 1994];

$$AC = \frac{most\ number\ of\ 'fires' - next\ highest}{number\ of\ tuples}$$

(3)

$$RC = \frac{most\ number\ of\ 'fires' - next\ highest}{most\ number\ of\ 'fires'}$$

(4)

Then the success rate (*SR*) is;

$$SR = \frac{number\ of\ correct\ classifications}{number\ of\ examplars} \times 100 \quad (5)$$

However, in the traditional scheme, the input connections for the different LUTs are chosen at random. The performance of the system increases with the number of LUTs but for a given size, the rate becomes very slow(Thomas, 1996). Therefore, the classification performance is a function of input mappings, and it is approximated by a normal distribution [Harald, 1993], where most the mappings give an average performance, but a small number of connections mappings offer a relatively better performance(Hannan, 2009).

2.2 Hidden Markov Model (HMM)

HMM has been considered as powerful tools for signal processing. HMM has been successfully used in speech recognition, recently, the application of HMM has been extended to include word and face recognition. An HMM topology is defined as the statistical behavior of an observable symbol sequence regarding a network of states, which represents the overall process behavior about movement between states of the process, and describes the inherent variations in the behavior of the recognizable symbols within a state (Abbas, 2004).

Therefore, according to the observation output densities of the hidden states, the HMM can be classified into Discrete and Continuous. The parameters of Discrete Hidden Markov Model (DHMM) are(Rabiner, 1989].

The number of states in the model (*N*). The interconnections of the states depend on the topology of the model. The individual states are denoted as $S = \{S_1, S_2, \ldots, S_N\}$, and the state at time t as q_t .

The number of symbols (*M*) in each state or the range of values employed in the observation vectors (codebook). The individual symbols are denoted as $V = \{v_1, v_2, \ldots, v_M\}$. The transition matrix ($A = [NxN]$) which comprises the probabilities of the state transitions, where

$$A = \{a_{ij}\} \qquad 1 \le i, j \le N \tag{6}$$

$$\{a_{ij}\} = P(q_{t+1} = S_j | q_t = S_i), \; 1 \le i, j \le N \tag{7}$$

$$\sum_{j=1}^{N} a_{ij} = 1 , \qquad 1 \le i \le N \tag{8}$$

The emission matrix ($B = [NxM]$) or the observation symbols probability distribution in the states, where

$$B = \{b_j(k)\}, \qquad 1 \le j \le N \tag{9}$$

$$b_j(k) = P(O_t = v_k | q_t = S_j), \; 1 \le j \le N , 1 \le k \le M \tag{10}$$

$$\sum_{j=1}^{N} b_j(k) = 1 , \qquad 1 \le k \le M \tag{11}$$

Each observation vector (O_t) comprises a number of elements and $O = (O_1, O_2, \ldots, O_T)$ be the observation sequence at T different observation instances, whereas the corresponding state sequence be $S = (q_1, q_2, \ldots, q_T)$, where $q_t \in \{ 1 , 2 ,..., N \}$.

$\pi = [Nx1]$; the initial state vector, $\pi = \{ \pi_i\}$, where

$$\pi_i = P(q_1 = S_i), \qquad 1 \le i \le N \tag{12}$$

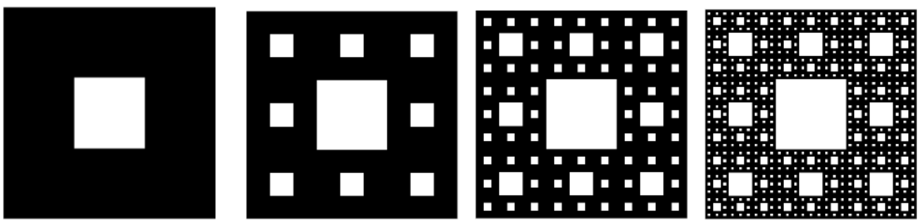
$$\sum_{i=1}^{N} \pi_i = 1 \tag{13}$$

The model is indicated by the following notation:

$$\lambda = (A, B, \pi) \tag{14}$$

3. Fractal Construction

The translational-dilation method (TMD) has been used for constructing the Sierpinski Carpet mapping [Nie, 1994]. Figure 2. has represented the different iterations Sierpinski carpet



(a) (b) (c) (d)

Figure 2. The Sierpiński carpet with three iterations (N=1,2,3 and 4) at a, b ,c and d respectively.

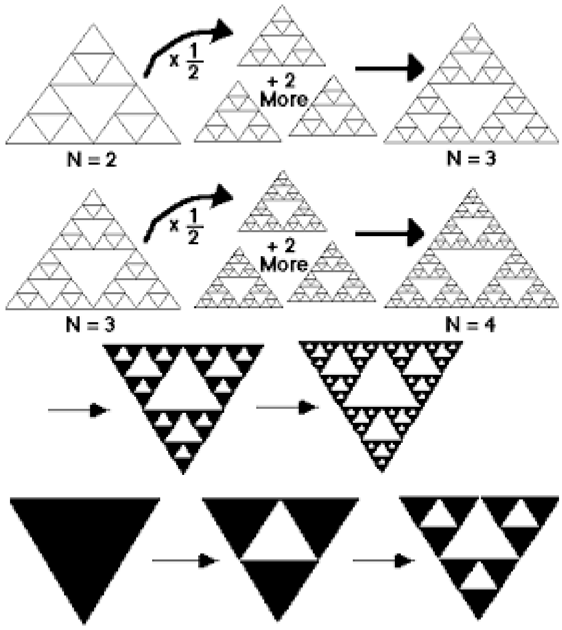
Where the relationship between the boxes and the corresponding iteration is;

No. of blacks boxes (*BB*)= 8^n, n=0, 1, 2........N (15)

total No. of boxes (*TB*)= 9^n , n=0, 1, 2,.......N (16)

No. of white boxes(*WB*) = *TB*-*BB* (17)

While, Figure 3. represents Sierpiński triangle fractal with a different dimension(Nie, 1994), which is used as a mask with HMM for face recognition. Where, this type of fractal is more suitable for face recognition than Sierpiński carpet.



(a) (b)

Figure 3. Sierpinski triangle with different dimension; a) normal, b) rotated by 180^o (for compatible with the face)

Where, Sierpinski triangle dimension is;

$$\frac{\log 3}{\log 2} = 1.584962500 \approx 1.585 \quad (18)$$

4. Proposed Algorithms; Recognition based on fractal technique

4.1 Digital Neural Network approach

The aim of the presented approach is to increase the robustness of the proposed algorithms, the functional capacity need to be raised, while maintaining the generalization ability, along with the training speed, and simple hardware implementation. The recognition system, shown in Figures. 4, 5. consists of two main parts:

Part I

1. Determine the recognizer function.
2. Select a suitable fractal geometry, which is suitable with the recognizer function.
3. Determine the suitable fractal degree (iteration), which is related to the required accuracy.

Part II

Conventional n-tuple through fractal geometry mask (ORing) has been adopted to satisfy the recognition process.

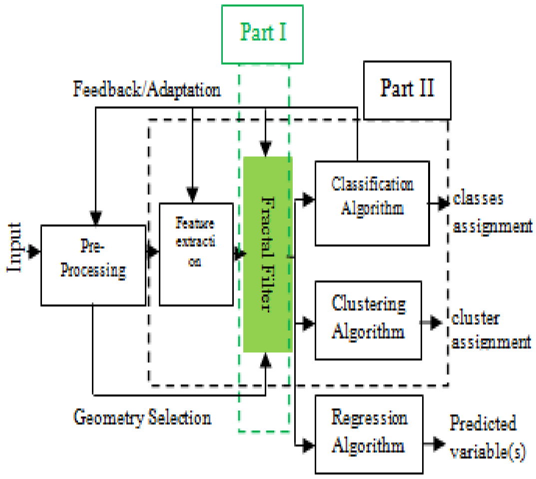


Figure 4. Block diagram of a proposed fractal based pattern recognition system

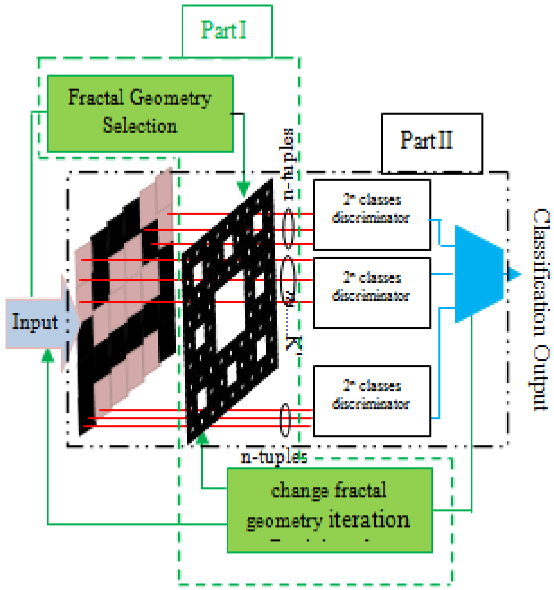


Figure 5. Operation of the proposed fractal-based pattern recognition system

In the experiments which were made, the images of English letters A-Z and a-z, with different font types (Twenty fonts) are used to evaluate the performance of the proposed algorithm. The simulation of the Sierpinski carpet has been carried out according to the algorithm in (Zhiyong, 2013). Its implementation has been made by using Matlab (R2014).

4.2 Discrete Hidden Markov Model (DHMM) approach

The preprocessing stage has been made by using a median filter with a fifth order to eliminate the effect of noise. The Support Vector De-composition (SVD) has been calculated, then using the top left element of U matrix (U_{11}). While, to represents the features, use the first two non-zero elements of the singular value vector Σ (Σ_{11} and Σ_{22}). Thus, each row (column) of the image (white box of the image mask when use fractal) is implemented by three values only. Therefore, the computational complexity of the system has been reduced.

Then, the quantization process of the features is done as follows; let $A = [U_{11} \quad \Sigma_{11} \quad \Sigma_{22}]$ be the vector of the three coefficients, and $L = [L_1 \quad L_2 \quad L_3]$ is the vector of quantization levels, where each coefficient would be quantized to a distinct level L_i. The difference between two successive quantized vectors is;

$$D = \frac{A_{max} - A_{min}}{L} \tag{19}$$

The maximum and minimum values are consequently replaced with new values depending on the comparison between the two vectors, whereas L remains constant. The final value of D is obtained when all differences between the successive vectors are computed. So the quantized vector of A is obtained as follows;

$$Q = \frac{A - A_{min}}{D} \tag{20}$$

The vector A is replaced with its quantized vector $Q = [Q_1 \quad Q_2 \quad Q_3]$ for all row (column) and each Q vector is implemented by a distinct label according to the following formula;

$$\text{Label} = Q_1 * W_1 + Q_2 * W_2 + Q_3 * W_3 \tag{21}$$

The vector $W = [W_1 \quad W_2 \quad W_3]$ is chosen experimentally with constant values for all calculations. The last step produces a sequence of integer values for each image which represents an input data for HMM.

In the training process, we use three states of left to right HMM topology. The initial estimations of the three parameters of the model (π, A, B) are chosen as follows (Hameed, 2016);

$\pi = [1 \quad 0 \quad 0]$
(22)

$$A = \begin{bmatrix} a_{11} & a_{12} & 0 \\ 0 & a_{22} & a_{23} \\ 0 & 0 & a_{33} \end{bmatrix} \qquad (23)$$

The possible values of a_{mn} has been calculated according to the constraint in (8). The matrix B is estimated using the following equation [Hameed, 2016];

$$B = \frac{1}{M} * \text{ones}(3, M) \qquad (24)$$

where M represents the number of observation symbols per state.

The Baum-Welch algorithm is used to train ten images for each person, and the model iterates twice.

In the testing process, each test image is treated similarly as a training image in obtaining observation vectors. The probability of the observation vector of the test image with all models trained is obtained using Viterbi algorithm. The model which gives a maximum probability is assumed to be an index to the identified person.

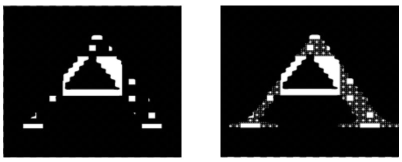
5. Parametric Studies

5.1 N-tuple masked by a fractal.

According to the operation of the proposed algorithm, as in Figures. 3, 4., the selection of iteration has been represented an important parameter in the recognition process. Where the input letter image will be the same size. Therefore, to obtained the optimal iteration degree; Table I show the iteration with size and recognition percentage. However, from this table, the optimal iteration is third degree. Furthermore, Figure 6., is represents image letter masked by a fractal.

Table I: Iteration Degree Against The Size And Recognition Percentage.

Iteration degree	No. of recognized letter	Image size pixels(boxes)	Recognition %
1	13	9	50%
2	23	81	88.46%
3	24	729	92.3%
4	24	6561	92.3%
5	23	59049	88.46%
6	23	531441	88.46%



(a) (b)

Figure 6. Latter image masked by fractal a) third iteration b) fifth iteration

image size = 729 pixels
iteration(*nt*) = 3
No. of blacks boxes (*BB*)= 8^3 =512
total No. of boxes (*TB*)= 9^3 =729

No. of white boxes(WB) = TB-BB = 9^3 - 8^3 = 214
number of the white boxes (cell)=214
number of the black boxes (cell)=729

The proposed algorithm depicted in fig. 5. has been modeled, and its performance has been evaluated with the following assumptions;

L=$N \times M = 27 \ \times 27$: $dimensions\ of\ input\ matrix$ (image size in pixels).
Assume the pixels that form the n-tuple are not correlated.
Number of cells in n − tuple (nt) = 10
Number of tuples in conventional algorithm (Rc) = 72
Number of tuples in proposed algorithm(Rp) = 21
Therefore, the memory size, $C_{n\text{-}tuple}$ needed will be;

$$C_{n-tuple} = 2^n \times \frac{L_{ip}}{n} \qquad (25)$$

where: L_{ip} Image size that has been processed.
For conventional n-tuple, the memory size is;

$$C_{\text{conventional}\ algorithm} = 2^{10} \times N \times M = \ 1024 \times \frac{729}{10} = 74649\ bits$$

While, for the presented algorithm, the memory size is;

$$C_{proposed\ algorithm} = 2^{10} \times WB = \ 1024 \times \frac{214}{10} = 21913\ bits$$

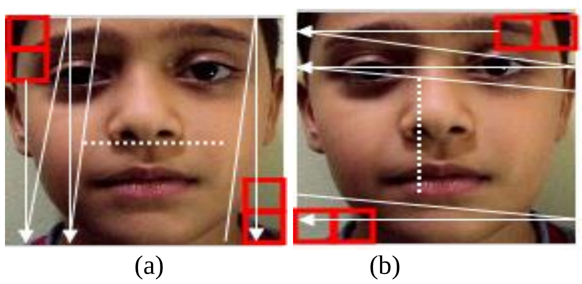
The benefit of the proposed algorithm is to reduce the required memory size. The memory size reduction percentage is given by;

$$\frac{C_{proposed\ algorithm}}{C_{n-tuple}} = \frac{21913}{74649} \times 100 = \ 29.35\% \qquad (26)$$

From these calculations, the proposed algorithm does not require a large memory. The number of total bits required for the proposed algorithm is 21913 bits while that for conventional n-tuple is 74649 bits. Therefore, the memory size has been reduced by 29.35%. The testing time (recalling time) will be reduced by the same percentage as that for the memory size. Also, the location of tuple is appointed.

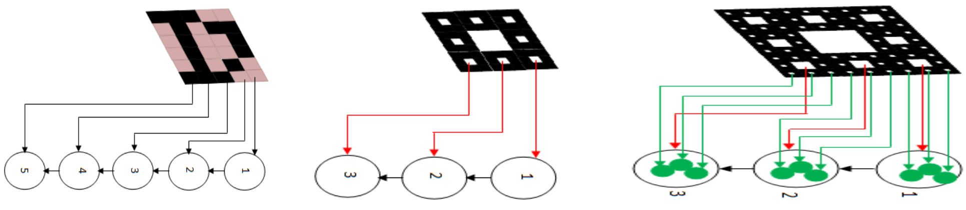
5.2 HMM masked by a fractal (HMMF).

To improve the operation of HMM by reducing the computational complexity, fractal masking is used and then compared with conventional HMM. Figure 7. represents the HMM states process for face recognition. While Figure 8. represents the Sierpinski carpet which used as character mask for character recognition by using HMM with different iterations.



(a) (b)

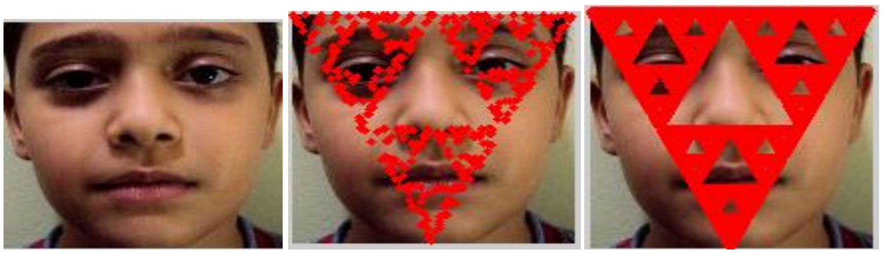
Figure 7 HMM states image size (179x296x3), (a) vertical state, (b)horizontal state.



(a) (b) (c)

Figure 8 Fractal HMM; a) HMM without fractal, b) Fractal (with N=2) HMM, c) Fractal (with N=3) HMM

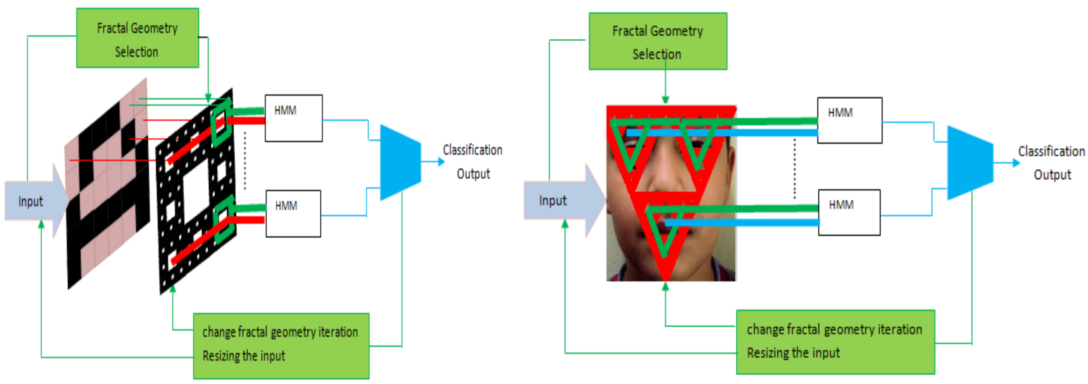
Figure 9. represents the HMM masked by Sierpinski triangle which proposed for face recognition than Sierpinski carpet.



(a) (b) (c)

Figure 9 face image with fractal masking, a) normal, b) with Sierpinski triangle 3rd iteration, c) with Sierpinski triangle 4th iteration

As stated above for Figure 4 and 5., Figure 10. represents the proposed algorithm for use HMM masked with fractal geometry type according to the application. Where, it is used Sierpinski carpet and triangle for character and face recognition respectively. The main steps in the HMMF for face recognition are; the first step, after the crop of face three points are determined these are; top left-right and down mid-points. The second step is construct the Sierpinski triangle compatible with the above three point. The third step is the selection of iteration according to the acquired features



(a) (b)

Figure 10 Proposed Pattern recognition based on HMM masked by fractal; a) character recognition, b) face recognition

6. Results and Discussion

This section is describing the simulation results. Table II represents the recognition of the proposed algorithms against the conventional n-tuple (with, and without frequency of occurrence) and HMM algorithms for twenty different fonts, Table III represents the average results of Table II.

Table Ii. A Comparison Between Proposed And Co Nventional Algorithms.

Font Name	Test Set	Letters Recognition				
		n-tuple			HMM	
		Fractal based Algorithm	Conventional Algorithm		Conventional Algorithm	Fractal based Algorithm
			K1	*K2*		
Algerian	A.......Z	16	15	20	17	16
Agency FB	A.............Z	16	11	13	17	17
Batang	A.....Z	20	15	15	19	20
Bell MT	A.......Z	22	22	19	22	23
Brush Script MT	A.....Z	2	1	0	3	4
Comic Sans MS	A.....Z	9	13	11	9	11
Copperplate Gothic Bold	A....Z	22	19	22	24	24
DFKai-SB	A....Z	17	12	12	19	20
Eras Light ITC	A..........Z	23	14	13	24	22
Footlight MT Light	A......Z	21	20	18	22	22
Goudy Stout	A..Z	4	6	2	7	6
Gulim	A......Z	19	18	20	20	22
Kalinga	A.......Z	17	15	12	22	22
Lucida Fax	A.....Z	21	18	19	22	22
Meiryo UI	A.....Z	7	7	11	9	11
Microsoft JhengHei	A......Z	23	19	22	23	24
NSimSun	A....Z	15	11	12	15	15
Mongolian Baiti	A......Z	24	24	24	24	24
Perpetua	A.....Z	18	19	12	19	20
Tw Cen MT	A......Z	18	16	15	20	20
Number of recognized letters		16.7	14.75	14.6	17.85	18.25 with reduced the complexity by 60%

K1= number of characters recognized by conventional n-tuple algorithm.
K2= number of characters recognized by conventional n-tuple with the frequency of occurrence algorithm.

Table Iii. Average Recognition For Twenty Fonts

Algorithms / Measurement	Conventional n-tuple	Conventional n-tuple with Frequency of occurrence	Proposed Algorithm (Fractal)
Average Recognition (%)	56.73	56.15	64.2

As can be seen, from Tables II and III, the recognition of the proposed algorithm has overperformed the conventional algorithms. Where it can be observed that, the proposed algorithm n-tupleF is advancing the conventional n-tuple by 12.5% and 11.6% with and without frequency of occurrence respectively. In addition, the improvements noted in the HMMF proposed algorithm is 2.19% in recognition side, while it is 60% in complexity reduction. This improvement is due to the suitable cell selection and then the tuple connections. The absolute confidence, (*AC*), and relative confidence, (*RC*), have been calculated for the proposed algorithm and the conventional n-tuple with, and without frequency of occurrence as shown in Table IV. These calculation have been made for the Mongolian Baiti font, where the algorithms has been gained highest recognition with it. These calculations have declared the guard zone between the right pattern and the nearest erroneous one. In this respect, this test has shown that, the proposed algorithm is better than the conventional n-tuple by 65.37% and 67.55% for *AC* and *RC* respectively. Furthermore, in the case of conventional n-tuple with a frequency of occurrence, the proposed algorithm is lowest by 7.61% for *AC*, while the proposed algorithm is advancing it by 61.93% for *RC*, as shown in Table V.

Table Iv. Calculated Ac And Rc For The Proposed And Conventional Algorithms For Mongolian Baiti Font

Letter	Proposed Algorithm (Fractal)		Conventional n-tuple algorithm			
			Without Frequency of occurrence		With Frequency of occurrence	
	AC	RC	AC	RC	AC	RC
A	0.242	0.353	0.072	0.091	0.209	0.098
B	X	X	X	X	X	X
C	0.003	0.007	0.035	0.042	0.104	0.047
D	0.041	0.074	X	X	0.099	0.048
E	X	X	X	X	X	X
F	0.014	0.018	0.053	0.060	0.133	0.058
G	0.024	0.044	0.005	0.006	X	X
H	0.072	0.104	X	X	0.037	0.018
I	0.245	0.241	0.064	0.077	0.229	0.099
J	X	X	0.029	0.038	X	X
K	0.142	0.177	0.005	0.007	X	X
L	X	X	X	X	0.098	0.044
M	0.247	0.280	X	X	X	X
N	0.125	0.150	0.064	0.081	0.166	0.082
O	X	X	X	X	0.012	0.006
P	0.105	0.123	0.008	0.010	0.064	0.030
Q	0.043	0.068	0.009	0.012	X	X
R	0.131	0.169	0.046	0.058	0.064	0.033
S	0.136	0.184	0.084	0.101	0.155	0.073
T	0.006	0.009	0.017	0.020	0.105	0.046
U	0.242	0.353	0.072	0.091	0.209	0.098
V	X	X	X	X	X	X
W	0.003	0.007	0.035	0.042	0.104	0.047
X	0.041	0.074	X	X	0.099	0.048
Y	X	X	X	X	X	X
Z	0.014	0.018	0.053	0.060	0.133	0.058
average	0.0722	0.0943	0.0250	0.0306	0.0777	0.0359

X- not recognized

Table V. Measurement Comparison Between Proposed And Conventional Algorithms

Measurements / Conventional algorithms	*AC*	*RC*	*SR*	Proposed algorithm
n-tuple	65.37% ↑*	67.55% ↑	11.6% ↑	
n-tuple with Frequency of occurrence	7.61% ↓	61.93% ↑	12.57% ↑	

*- The arrows location is indicated the algorithm side, where the proposed algorithm in the arrows left the side, while another algorithm is on the right side. Also, the direction of these arrows is indicated the Progress and delayed of each algorithm to the other.

Table VI describes the switching recognition ability of the proposed algorithm between the white and black boxes (cells), which represents the normal and inverse fractal geometry cases. This point is important in implementation, where, it is the selection operation between the white and black boxes in the connection treatment of the selected tuples (ORing and NORing operation). Also, HMM masked by fractal (HMMF) is presented in this table for comparison.

Table Vi. Recognition Of The Proposed Algorithm For White And Black Boxes

Font Name	Test Set	n-tuple				HMMF Rec%
		White boxes	Rec%	Black boxes	Rec%	
Algerian	A.......Z	16	61.54	15	57.69	61.54
Agency FB	A............Z	16	61.54	11	42.31	65.4
Batang	A.....Z	20	76.92	16	61.54	76.9
Bell MT	A.......Z	22	84.62	22	84.62	88.4
Brush Script MT	A.....Z	2	7.690	2	7.69	15.3
Comic Sans MS	A.....Z	9	34.62	12	46.15	42.3
Copperplate Gothic Bold	A....Z	22	84.62	16	61.54	92.3
DFKai-SB	A....Z	17	65.38	13	50	76.9
Eras Light ITC	A....... Z	23	88.46	23	88.46	84.6
Footlight MT Light	A......Z	21	80.77	18	69.23	84.6
Goudy Stout	A..Z	4	15.38	7	26.92	23
Gulim	A......Z	19	73.08	20	76.92	84.6
Kalinga	A.......Z	17	65.38	16	61.54	84.5
Lucida Fax	A.....Z	21	80.77	23	88.46	84.6
Meiryo UI	A.....Z	7	26.92	8	30.77	42.3
Microsoft JhengHei	A......Z	23	88.46	22	84.62	92.3
NSimSun	A....Z	15	57.69	17	65.38	57.69
Mongolian Baiti	A......Z	24	92.31	23	88.46	92.3
Perpetua	A.....Z	18	69.23	17	65.38	76.92
Tw Cen MT	A......Z	18	69.23	18	69.23	76.92
No. of recognized letters and recognition percentages		16.7	64.2%	15.95	61.3%	70.1685%

To complete the picture, the effect of the noise environment on the recognition ability for the best recognition font (Mongolian Baiti) has been demonstrated in Table VII. Figure. 11. presents the performance of the proposed algorithm and conventional algorithms with noise. From Table VII and Figure 11, it is clear that the proposed algorithm over-perform the conventional n-tuple and HMM algorithms. This over-performance is undoubtedly attributed to the adoption of fractal technique.

Table Vii. Average Recognition Of The Presented Algorithms In Noise For Mongolian Baiti Font

Algorithms / Measurement	Conventional n-tuple	Conventional n-tuple with Frequency of occurrence	Proposed Algorithm	
			n-tupleF	HMMF
Average Recognition	36.11%	34.8%	58.5%	68.3761%

To put our work in the proper place in the work map, the face recognition application by using proposed algorithm has been compared with other work as shown in Table VIII. The comparison proved the priority of our work. Where, our work satisfies the recognition percent with reduction of computing complexity.

Table Viii Comparison Proposal Work With Previous Studies For Face Recognition

Year	Ref. No.	Feature extraction method	No. of States	Training Time for one image (second)	Testing Time for one image (second)	Recognition Rate (%)
2010	P. Zhang, 2002	WT+PCA	5	NA	1.2	96.5
2011	J. M. Bishop, 1994	DWT	5	1.3	0.8	100
2012	Patnala, 2009	DCT	3	12.6	71.7	82.76
2013	Bruce, 2013	DGWT	7	NA	NA	99
2014	Paulo, 2010	GF+LDA	15	NA	NA	97.5
Present work	HMMF	SVD	Var			99.98
	n-tupleF	No. of tuple =10				98

Finally, for optimization, the recognition percentage with computation complexity, Figure 12. presents the recognition against the iteration number. From this figure it is clear the different behavioral between the n-tupleF and the HMMF because process philosophy between them is different. Where, n-tupleF deal with pixel value while HMMF deal with the essence of the pixel.

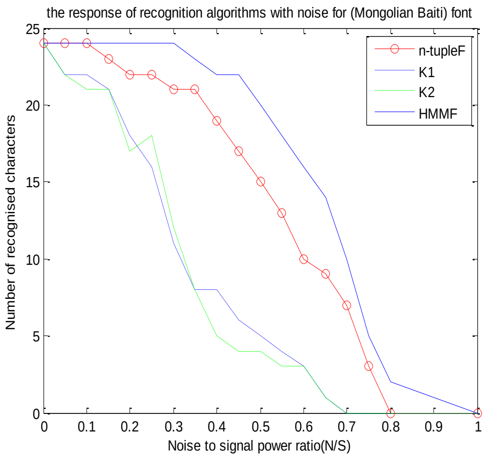


Figure 11: The response to the proposed and conventional algorithms against noise.

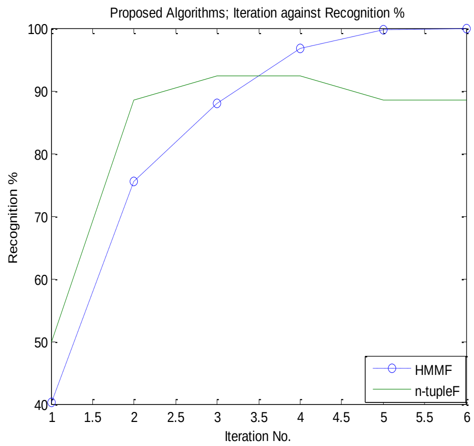


Figure 12 Recognition against Iteration number

7. Conclusions

In this paper, the Sierpiński Carpet fractal geometry, up to the third iteration, has been proposed for use in machine letter recognition, while Sierpinski triangle with the fourth iteration has been proposed for face recognition with n-tuple and with HMM respectively. The results show that the proposed n-tuple fractal based techniques overperformed the conventional techniques in term of memory size, recalling time and recognition percentage. Where, the memory sizes and recalling time are reduced by 29.35%, while the recognition improves the conventional n-tuple by 12.5% and 11.6% with and without frequency of occurrence respectively. Also, the confidence measurements appear that the over-perform of the proposed algorithm than the conventional algorithm by 65.37% and 67.55% for *AC* and *RC* respectively. The conventional n-tuple with a frequency of occurrence is higher than the proposed algorithm by 7.61% for *AC*, and lower than it by 61.93% for *RC*. The proposed algorithm is overperform the conventional algorithms by 12.57% and 11.6% for success rate (*SR*) with and without frequency of occurrence respectively. Also the performance of the proposed algorithm has shown to be better than that of the conventional algorithm in the presence of noise by 1.6 times. In this context, HMMF is over-performed the n-tupleF by 14.4% for character recognition, while it is robustness than the n-tuple in noise environment by 8%. Also, HMMF has been Showed progress over conventional HMM in previous studies. The successful usage of the proposed algorithm on machine letter and face recognition is based on the specific selection of fractal geometry related to the application. Also, the reality of n-tuple and HMM process, where HMM deal with the essence of pixel feature which can be represents as pixels feature like intensity and color. Then, the n-tuple deal with the value of pixel feature represents the location pixels feature in the x-y plane.

Acknowledgements

The author would like to express his thanks to Prof. Jawad K. Ali, University of Technology-Iraq, for the valuable discussions concerning fractal geometry and continuing support.

References

Abbas H. Hassin, Xiang L. Tang, Jia F. Lui, and Wei Z. , Jule 2004,"Printed Arabic Character Recognition Using HMM", J. Computer Scintific and Technlogy, Vol. 19, No. 4, pp. 538-543.

Allinson N M and A Kolcz, 1997, "A Principled Approach to N-tuple Recognition System", IEE Conference on Pattern Recognition, pp. 201 - 210, DOI: 10.1049/ic:19970125.

Bishop J. M. and Mitchell R. J. , 1994, "Auto-associative memory using n-tuple techniques", IET Journals & Magazines, Intelligent Systems Engineering, Volume: 3, Issue: 4, pp. 222 - 229.

Bishop J. M. and Mitchell R. J. , Winter 1994, "Auto-associative memory using n-tuple techniques", IET JOURNALS & MAGAZINES, Intelligent System Engineering, Vol. 3, Issue: 4, pp. 222- 229.

Bruce A. Wilkie and Radek Holota , 2013, " The comparative recognition of monochrome and color images using networks of n-tuple and Min/Max nodes utilizing 'Grouped Nodes' ", IEEE Conference , International Conference on Applied Electronics (AE), pp. 1 - 6.

Bruce A. Wilkie and Radek Holota, 2012, " The utilization of grouped n-tuple logic nodes in artificial neural Single Layer Nets, Enhanced pattern recognition characteristics ", IEEE Conference International Conference on Applied Electronics (AE), pp. 335 - 340.

Danilo S. Carvalho, Hugo C. C. Carneiro, Felipe M. G. Fran¸ ca, and Priscila M. V. Lima, April 2013, "B-bleaching : Agile Overtraining Avoidance in the WiSARD Weightless Neural Classifier", ESANN 2013 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), pp.24-26.

George Bolt, James Austin and Gary Morgan, May 1992, "Uniform Tuple Storage in ADAM", Elsevier, Artificial Neural Networks, Pattern Recognition, Volume 13, Issue 5, pp. 339–344.

George BOLT, James AUSTIN and Gary MORGAN, May 1992, "Uniform Tuple Storage in ADAM", Elsevier, Pattern Recognition Letters, Artificial Neural Networks, Volume 13, Issue 5, pp. 339–344.

Hajian A. A. Nezhad and Mozaffari S. , Sep. 2014, "Font Discrimination using Fractal Dimensions", Iranian Journal of Electrical & Electronic Engineering, Vol. 10, No. 3, pp. 176-187.

Hameed R. Farhan, Mahmuod H. Al-Muifrajem and Thamir R. Saeed , January 2016, " Face Recognition using Maximum Variance and SVD of Order

Statistics with only Three States of Hidden Markov Model", International Journal of Computer Applications (0975 – 8887) Volume 134 – No.6.

Hannan M.A. Bin Azhar, Farzin Deravi and Keith Dimond, 2009 , " Hybridisation of GA and PSO to Optimise N-tuples ", IEEE, Proceedings of International Conference on Systems, Man, and Cybernetics, pp. 1815 - 1820, DOI: 10.1109/ICSMC.2009.5346854.

Harald Huning, " In search of connections in weightless neural networks", citessex, in AACHENER BEITRAGE ZUR INFORMATIK, 3; Neural networks

Jim Austain, 1996, " High Speed Image Segmentation Using a Binary Neural Network", Springer, Neurocomputation in Remote Sensing Data Analysis, pp 202-213.

Jim Austin, 1990, " Uncertain Reasoning with RAM Based Neural Network", Journal of intelligent system, Vol. 2, No. 4, pag. 121-154, Publisher: Freund Publishing (UK).

Marcilio C. P. de Souto and Paulo J. L. Adeodato, 1998, "Learnability in Sequential RAM-based Neural Networks", IEEE Conference, Vth Brazilian Symposium on Neural Networks, pp. 20 - 25, DOI: 10.1109/SBRN.1998.730988.

Mazin Al Hadidi and Rami S.Razouq, 2012, "Hand Written Digits Recognition Using Digital Learning Network", Elsevier, International Conference on Future Computer Supported Education, SciVerse Science Direct, Volume 2, pp. 367-371.

Mohammed Talibi-Alaoui and Abderrahmane Sbihi, 2012, " Fractal Features Classification for Texture Image Using Neural Network and Mathematical Morphology", Proceedings of the World Congress on Engineering , Vol I WCE, July 4 - 6, ISBN: 978-988-19251-3-8, London, U.K.

Nie H. B., Yu B. M. and Yao K. L. , 1994, "Critical Percolation Probabilities for the Next-Nearest-Neighboring Site Problems on Sierpinski Carpets", communication theory physics 21, international academic publishers, Vol. 21, No. 3, pp 281-288, April 30.

Patnala S. R. Chandra Murty, E. Sreenivasa Reddy, and I. Ramesh Babu, September 2009," Iris Recognition System Using Fractal Dimensions of Haar Patterns", International Journal of Signal Processing, Image Processing and Pattern Recognition Vol. 2, No.3, pp. 75-84.

Paulo J. L. Adeodato, and Rosalvo F. Oliveira Neto, 2010, "pRAM n-Tuple Classifier - a New Architecture of Probabilistic RAM Neurons for Classification Problems", IEEE, International Joint Conference on Neural Networks (IJCNN), pp. 1 - 7, DOI: 10.1109/IJCNN.2010.5596779.

Rabiner L. , 1989, “A tutorial on hidden markov models and selected application in speech recognition”, In Proc. of IEEE, vol. 77, No. 2, IEEE, PP. 257-286.

Stonham T.J., Wilkie B.A. and Masih L. , 1987, " Higher Order Adaptive Nnetworks Some Aspects of Multi-class and Feed-back Systems", AVC, doi:10.5244/C.1.34, pp. 245-249.

Teresa B. Lumdermir, Andre De Carvlho, Antonio P. Braga and Marcilio C. P. De Souto, 1999, " Weightless Neural Models: A Review of Current and Past Works", Neural Computing Surveys 2, pp. 41-61.

Thomas M. Jorgensen: Year: 1996, " A RAM Based Neural Net with Inhibitory Weights and its Application to Recognising Handwritten Digits", IEEE, Proceedings., International Workshop on Neural Networks for Identification, Control, Robotics, and Signal/Image Processing, pp. 228 - 236.

Yong Wanga & Hui Guoa, 2015, "Shape Recognition of Tyre Marking Points Based on Support Vector Machine", IETE Technical Review, pp. 123-130, 13 Jan 2015Volume 32, Issue 2.

Zhang P., Bui T.D., and Suen C.Y., Year: 2002 "Recognition of Similar Objects Using 2-D Wavelet-Fractal Feature Extrac", IEEE Proceedings of 16th International Conference on Pattern Recognition, Volume: 2 pp. 316 - 319.

Zhiyong Zhu and Enemi Dong, October 2013" Simulation of Sierpinski-type fractals and their geometric constructions in Matlab environment", WSEAS Transactions on Mathematics, Issue 10, Volume 12, pp. 992-1000.