International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:8, No:4, 2014

Improved Dynamic Bayesian Networks Applied to Arabic on Line Characters Recognition

Redouane Tlemsani, Abdelkader Benyettou

***Abstract*—**Work is in on line Arabic character recognition and the principal motivation is to study the Arab manuscript with on line technology.

This system is a Markovian system, which one can see as like a Dynamic Bayesian Network (DBN). One of the major interests of these systems resides in the complete models training (topology and parameters) starting from training data.

Our approach is based on the dynamic Bayesian Networks formalism. The DBNs theory is a Bayesians networks generalization to the dynamic processes. Among our objective, amounts finding better parameters, which represent the links (dependences) between dynamic network variables.

In applications in pattern recognition, one will carry out the fixing of the structure, which obliges us to admit some strong assumptions (for example independence between some variables). Our application will relate to the Arabic isolated characters on line recognition using our laboratory database: NOUN. A neural tester proposed for DBN external optimization.

The DBN scores and DBN mixed are respectively 70.24% and 62.50%, which lets predict their further development; other approaches taking account time were considered and implemented until obtaining a significant recognition rate 94.79%.

***Keywords*—**Arabic on line character recognition, dynamic Bayesian network, pattern recognition.

I. Introduction

SINCE the Sixties, the man seeks "to learn how to read" for computers. This recognition task is difficult for the isolated handwritten characters because their forms are varied compared with the printed characters. The on line recognition makes it possible to interpret a writing represented by the trajectory of the pen.

This technique is in particular used in the electronic message minders of type Personal Digital Agenda. An electronic shelf and a special pen are necessary. The signal is collected in real time. It consists of a succession of co-ordinates of points, corresponding to the position of the pen with time regular intervals. Indeed, the on line signal contains dynamic information absent in the off line signals, such as the order in which the characters were formed, their direction, the pen down and pen up position [1].

So that the isolated character recognition is strongly precise, it is significant to as structure characters model the usually as possible. In this work, we consider that a character is composed of strokes and even their relationships were kept. The strokes are the conceptual elements, their space relations are conceptually significant, and which are usually robust against geometrical and significant variations for the distinctive characters of the similar forms.

A Bayesian network can model dependencies between several random variables in a probabilistic and graphic way representation. We used the Bayesian networks for their capacity to represent a Gaussian distribution of joined conditional probabilities on a set of random variables.

II. Dynamic Bayesian Networks

The Dynamic Bayesian Networks (DBN) prolongs the representation of Bayesian Networks (BN) to the dynamic processes. A DBN codes the Jointed Probability Distribution (JPD) of time evolution $X[t]=\{X_1[t]...., X_N[T]\}$ of variables. In other words, it represents the belief about the possible trajectory of the dynamic process X[t]. After a similar notation with the static representation of BN, the JPD for a finished interval of time [1,T] is factorized like:

$$p\left(X[1],....,X[T]\right)=\prod_{t=1}^{T}\prod_{i=1}^{n}P\left(X_i[t]\middle|\Pi_i[t]\right) \tag{1}$$

where $\prod_i[t]$, the $X_i[t]$ parents in the graph indicate DBN structure. The DBN graphic structure can be looked like concatenation of several dependent static BNs with the temporal arcs. We call each one of these static networks a section of time (a section of time is defined like collection of the set of X[t] in only one time T instantaneous and their parents associated $\prod[t]$ in the structure of graph) with DBN In the most general case, if no pretention are imposed on the fundamental dynamic process, the structure of the graph and the numerical parameterization of a DBN can be different for each time out in sections. In this case, the DBN is regarded as BN (static) with $T\times n$ variables and the coding of the JPD can be extremely complex.

A. Representation

In the literature the representation of DBN generally is employed for the first stationary order Markov processes. For this case, Friedman and others described a representation simplified in terms of two static head BNs definite above the variables of a simple time section like cited in [2]. The principal representation is based on the pretention of stationarity which implies that the structure and the parameters of DBN repeat. The JPD is coded by using a first network and an unrolled transition network.

R. Tlemsani is with the National Institute of Telecommunication and ICT of Oran Algeria (phone: +213 550669689; e-mail: rtlemsani@ito.dz).

A. Benyettou was with University of Sciences and Technology of Oran, Algeria (e-mail: aek.benyettou@univ-usto.dz).

The initial network codes the irregular structure in the border and indicates the initial states of surplus of X[1] distribution. The transition network codes the invariable probability transition time given by $P\left(X[t+1]|X[t]\right)$. The JPD for a finished time interval is obtained by unrolling the transition network for a sufficient number of times sections. The mechanism of unfolding is composed to present a set of variables for each time out in sections and to fold up the structure and the parameters of transition network on these variables. Rearranging the limits JPD is factorized above the networks initial and transition like:

$$p(X[1],\ldots\ldots,X[T]) = P_{B_I}\left(X[1]\right)\prod_{t=2}^{T-1} P_{B\to}\left(X_i[t]|X[t-1]\right) \tag{2}$$

where $P_{B_I}(.)$ and $P_{B\to}(.)$ are the densities of probability coded by the initial and transition networks, respectively.

B. Inference in DBNs

The problem of inference in DBNs is similar to the problem of inference of BN such as desired the quantity is the posterior marginal distribution of a set of hidden variables indicated an order of the observations (updated of belief): $P\left(X_h[t]|X_0[1],\ldots,X_0[\tau]\right)$ where $X[t]=\left\{X_h[t],X_0[t]\right\}$ is a set of time evolution variables in which $X_o[t]$ and $X_h[t\ ]$ indicate observed variables and hidden, respectively. The time series inference is generally under the name filtering $(\tau=1)$, smoothing $(\tau>1)$ and the forecast $(\tau<1)$ according to the time window of observation used in calculations.

A direct approach to imply probabilities in a DBN, is to build an enormous static BN for the desired number of time sections and then to employ the general algorithms of inference for static BNs. However, this requires that the end of about a time be known a priori. Moreover, the data-processing complexity of this approach can extremely require (particularly in terms of memory). Consequently, in general, the DBN inference is carried out by using the recursive operators who update the belief state of DBN while the new observations become available. The principle is similar to the message-passing algorithm for static BNs. The idea is with the messages defined on a Markov cover of the variables which D-separates the past from the future and employs a process towards the procedure forward- backward to distribute all the obviousness along the DBN detailed in [2]-[4]. This technical requires only one time window of the variables to be maintained in the memory. These algorithms are indeed generalization of the algorithm (Baum-Welch) towards forward-backward well-known mentioned in [5] in special HMMs and cases JLO algorithm explained in [6].

III. Modeling

In this part, we consider that a character is composed of strokes and their relationships. The strokes are direct elementary lines or almost rights which have directions distinct from the lines connected in the writing order. The relationships of the strokes indicate the dependencies of the positions between the strokes obtain an influence on the others strokes.

A. Static Model

An example of stroke is composed of points. Consequently, a stroke model is composed of point models with their relationships, called "Within Stroke Relationships" (ISRs).

Fig. 1 shows the recursive example of stroke construction. To the first recursive iteration (D=1), IP1 is added to median model points of all the stroke examples. It has the WSR of the final points (arcs of EP0 and EP1 with IP1). To the second recursive iteration (D=2), IP2 and IP3 are added for median points of the strokes partial lifts and right-hands side, respectively. Moreover, they have the WSR of the final points of the partial strokes. Fig.1 (c) is the prolonged model of stroke.

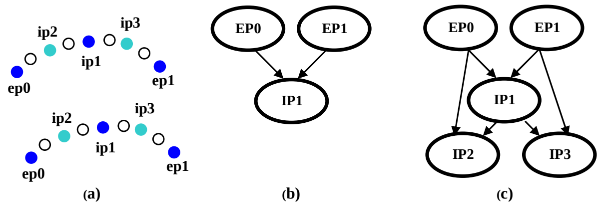

Fig. 1 The recursive construction of a stroke model(a) Example for ip1's: median point of stroke ip2s et ip3's: those of the strokes partial lefts and right, (b) stroke model depth d = 1,(c) Stroke Model depth d=2

With this recursive process, a model of stroke can as many have point models according to needs. In this part, the recursively depth *d=3* is selected for all the stroke models.

It is worth the sorrow to note that the models of point to great recursively depths, do not incur the problem of non adequate model. Because when the depth is large, the partial strokes become much shorter and linear. Consequently, ISRs become much stronger and the joined probabilities of the additional point models obtain more close the probability of only one. The joined probability is obtained from those of the models of point. Let us suppose that a model S has the depth D and an example of stroke is points length T: O(1)…,O (T).To match, the example of stroke is periodically taken in the 2d-1 median points. They are indicated like IP1, IP2...., IP2d- 1 according to the order of the process of recursive taking away.

Then, IPi examples of point are matched with the models IPi of point.T he joined probability is calculated as follows by the local Markov property of the conditional probabilities in the Bayesian networks:

$$\begin{aligned} P\left(S=O(1),\ldots,O(t)\right) &= P\begin{pmatrix} EP_0=O(1), EP_1=O(t), \\ IP_1=ip_1,\ldots,IP_{2^d-1}=ip_{2^d-1} \end{pmatrix} \\ &= P\left(EP_0=O(1)\right)P\left(EP_1=O(t)\right)\times\prod_{i=1}^{2^d-1}P\left(IP_i=ip_i\setminus pa(IP_i)\right) \end{aligned} \tag{3}$$

where the pa(IPi) is the configuration of the nodes parents which the arcs of dependence like in IPi.

B. Dynamic Model

An example of character is composed of the strokes. Moreover, the close connections exist between them. Consequently, a character model is composed of the stroke models with their relationships, called "Inter Stroke Relationships (ISRs)".

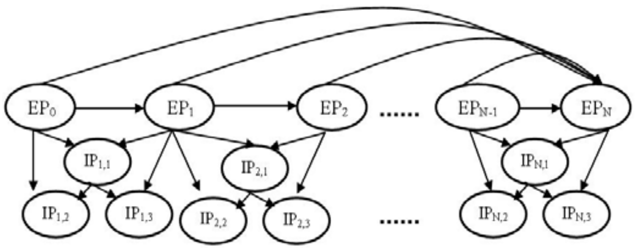

Fig. 2 The representation by Bayesian network of a character model with N strokes and depth d = 2

In Fig.2, EP_0 is the first point model written in a character.

The point models of first stroke are written in the order of $IP_{1,2}$, $IP_{1,1}$, $IP_{1,3}$.Then, the models of point of the second strokes are written in the order of EP_1, $IP_{2,2}$, $IP_{2,1}$, $IP_{2,3}$. Alternatively, the following strokes are written in the same way. In conclusion, EP_N is the last model of point written in a character showed in [7]-[9].

The model of probability of a character is calculated by the enumeration of all the possible segmentations of stroke. Let us suppose that a model BN of character has N stroke model and an entry of character with T points: O(1)…,O(T). Since the entry does not have the information of border, various segmentations are possible. One poses an example of stroke segmentation by$\gamma = (t_0, t_1\ldots, t_N)$, $t_0 = 1<t<\ldots <t_N = T$, and the set of totality by Γ. Then the probability model of a character is given as follows:

$$
\begin{aligned}
&P\big(O(1),...,O(t)\setminus BN\big)\\
&= \sum_{\gamma=(t_0,...,t_N)\in\Gamma} P\big(S_1 = O(t_0,t_1),...,S_N = O(t_{N-1},t_N)\big)\\
&=\sum_{\gamma\in\Gamma}\prod_{i=1}^{N} P\big(S_i = O(t_{i-1},t_i)\setminus S_1 = O(t_0,t_1),...,S_{i-1} = O(t_{i-2},t_{i-1})\big)\\
&=\sum_{\gamma\in\Gamma}\prod_{i=1}^{N} P\big(S_i = O(t_{i-1},t_i)\setminus EP_0 = O(t_0),...,EP_{i-1} = O(t_{i-1})\big)
\end{aligned}
\tag{4}
$$

where $O(t_i, t_j) = O(ti), O(t_i+1)\ldots,O(t_j)$. The joined probability given by preceding strokes is calculated as follows:

$$
\begin{aligned}
&P\big(S_i = O(t_{i-1},t_i)\setminus EP_0 = O(t_0),...,EP_{i-1} = O(t_{i-1})\big) =\\
&\begin{cases}
P\big(EP_i = O(t_i)\setminus O(t_0),...,O(t_{i-1})\big)\\
\prod_{j=1}^{2^d-1} P\big(IP_{i,j} = ip_{i,j}\big(O(t_{i-1},t_i)\big)\setminus pa\big(IP_{i,j}\big)\big) \ \ if\ i>1,\\
P\big(EP_0 = O(t_0)\big)P\big(EP_1 = O(t_1)\setminus O(t_0)\big)\\
\prod_{j=1}^{2^d-1} P\big(IP_{i,j} = ip_{i,j}\big(O(t_{i-1},t_i)\big)\setminus pa\big(IP_{i,j}\big)\big) \ \ if\ i=1,
\end{cases}
\end{aligned}
\tag{5}
$$

where$ip_{i,\,j}(O(t_{i-1}, T_i))$ are the j[st] point sample of $O(t_{i-1}, t_i)$.While substituent(4) for (5), the probability of the model it is only one product of the joined probabilities of EPs and IPS:

$$
\begin{aligned}
&P\big(O(1),...,O(t)\setminus BN\big) =\\
&\sum_{\gamma\in\Gamma}\prod_{i=0}^{N} P\big(EP_i = O(t_i)\setminus O(t_0),...,O(t_{i-1})\big)\\
&\times\prod_{i=1}^{N}\prod_{j=1}^{2^d-1} P\big(IP_{i,j} = ip_{i,j}\big(O(t_{i-1},t_i)\big)\setminus pa\big(IP_{i,j}\big)\big)
\end{aligned}
\tag{6}
$$

The joined probabilities of EPs can be interpreted by probabilities of the stroke positions total and those of IPs with probabilities of the local stroke forms.

IV. Recognition and Training

A. Recognition Algorithm

A handwritten character is identified by finding the model of character which produces the highest posterior probability given entry. When the Iist model of character is indicated BNi and the points entrance as O(1)…, O(T), then the recognition problem can be formulated as follows:

$$
\begin{aligned}
&\arg\max_i P\big(BN_i\setminus O(1),...,O(T)\big)\\
&=\arg\max_i \frac{P(BN_i)\,P\big(O(1),...,O(T)\setminus BN_i\big)}{P\big(O(1),...,O(T)\big)}\\
&=\arg\max_i P(BN_i)\,P\big(O(1),...,O(T)\setminus BN_i\big)
\end{aligned}
\tag{7}
$$

The model character probability is described previously. To calculate it, all possible stroke segmentations Γ are considered. To prevent the time exponential complexity, we suppose that it can be brought closer by the character joined probability of the most probable segmentation γ^* in Γ as follows:

$$
P\big(O(1),...,O(T)\setminus BN_i\big)\approx\max_{\gamma\in\Gamma} P\big(S_1 = O(t_0,t_1),...,S_N = O(t_{N-1},t_N)\big)
\tag{8}
$$

To carry out the probability calculation of the handy model in time, we need one pretention for research ofγ^*.By matching a stroke, all the possible segmentations of its strokes preceding should be considered because dependencies of inter-stroke. For the simplicity of research, we suppose that the joined probability of a stroke is highest with the most probable configuration of the previous strokes. Then, the "dynamic programming search algorithm" can as follows be adopted: (see [10]-[12])

International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:8, No:4, 2014

S_i **:** $i^{ième}$ modèle de stroke
$\gamma_i(t)$: la segmentation la plus probable quand S1,…..,Si et O(1,t) sont matchés.
$\delta_i(t)$: la probabilité jointe donnée $\gamma_i(t)$.
Initialization
$\delta_0(1)=1$, $\gamma_0(1)=\{\}$
Stroke matching
for t=2 *to* T
for i=1 *to* N
$\delta_i(t) = max_{1\leq b<t} P(S_i = O(b,t) \setminus \gamma_{i-1}(b)).\ \delta_{i-1}(b)$
$b^* = argmax_{1\leq b<t} P(S_i = O(b,t) \setminus \gamma_{i-1}(b)).\ \delta_{i-1}(b)$
$\gamma_i(t) = \gamma_{i-1}(b^*) \cup \{t\}$
end
end
Probability of the character model
$P(O(1),\ldots,O(T) \setminus BN_i) \approx \delta_N(T)$

Fig. 3 Dynamic programming search algorithm

B. Training Algorithm

In this part, the structure of dependence is determined by a original model starting from knowledge a priori and the experiments. The recursively depth of the stroke models is selected equal to Three (*d=3*). The number of models is given starting from the typical number of stroke in the character. The conditional parameters of probability are formed by training data. They are the linear regression matrixes W'S ($W=[w_{i,j}]$) and covariancesΣ's for the points models. If all point models are matched the point following the example, then they can be estimated starting from the conventional statistical algorithms of regression with the maximum object of maximum probability ML "likelihood" [13]. Let us suppose that the point P depends on P1…, P_K and there are N training samples. One notes the i^{st} sample of P have p(i) and the values of dependent variable by $z(i) = [x(i)_1, y(i)_1 \ldots, x(i)_k, y(i)_k, 1]$. Then, [13] detailed this estimation as follows:

$$\Sigma = \frac{1}{N}\sum_{i=1}^{N} p^{(i)}\left(p^{(i)}\right)^T - \frac{1}{N}W\sum_{i=1}^{N} z^{(i)}\left(p^{(i)}\right)^T \quad (9)$$

$$W = \left(\sum_{i=1}^{N} p^{(i)}\left(z^{(i)}\right)^T\right)\left(\sum_{i=1}^{N} z^{(i)}\left(z^{(i)}\right)^T\right)^{-1} \quad (10)$$

During the training of the character model, the Re-estimate of the parameters and it required of the most probable segmentation in strokesγ* is repeated alternatively like cited in [11]-[13].

This approach is similar to the training algorithm EM (Expectation-Maximization). Being given the parameters (W and Σ),γ* is updated. Then, with the news γ*, the parameters Re-are estimated. The detailed algorithm is as follows:

- *Step 1*: To initialize the character model with the initial data (part of the examples of the manually segmented strokes).
- *Step 2*: To seek the most probable segmentation γ* of the totality of the characters of training not segmented by using the algorithm of required the previous one.
- *Step 3*: To estimate the parameters (W and Σ) on the examples partitioned byγ*.
- *Step 4*: To repeat stages 2 and 3 until the sum of probabilities of the model will not change any more (stability).

V. Experiences, Results and Analyses

In our experiences, we used NOUN database. This database is developed to initiate research and develop on line Arabic recognition systems. It contains 2800 isolated characters, in Arabic. It was collected near approximately 25 different scriptwriters, each one wrote all Arabic alphabet letters five times. This database was developed between SIMPA laboratory (Signal Image Speech Laboratory) at University of Sciences and the Technology USTO-MB and the LaRATIC laboratory (Research Applied and TICs Laboratory) at the National Institute of Telecommunications and Information Technologies and Communication INTTIC in Oran, Algeria.

A. Manual Segmentation for Isolated Characters

The aquisition made dynamically using a graphics tablet is had to digitalize. This one has a resolution specifies and samples at a speed selected writing. A time of adaptation is necessary to the script writer to be able to write has little close correctly despite everything for its treatment the character will have to be segments in trace i.e. in strokes elementary. Fig.4 shows the various segmentations implemented and applied for three Arabic letters.

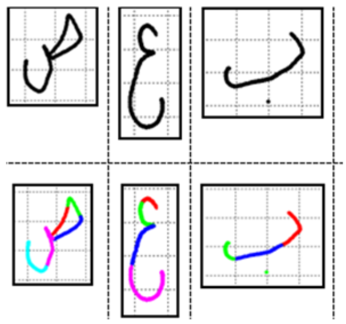

Fig. 4 Treatment and segmentation of characters "Ba", “Ain” and “Sad”

B. Dynamic Bayesian Network Experience

For our experiments one implemented and checking our Bayesian model on all Arabic letters set. Fig.7 shows the confusion matrix histogram of the recognition rates and they are relatively significant rates on such under corpus and it is an effectiveness returns to the safeguarded elements space and Gaussian probabilities of each point model. The total rate of recognition attempt 66.78%.

C. Neural Verifier

The method implemented in this article is a probabilistic model which is summarized with the concept of the dynamic Bayesian networks and after having obtained γ* the vector of probabilities by the dynamic programming algorithm, instead of taking the maximum value, the researchers in pattern recognition often use a verifier operator on this level to adapt the system. We prefer integrate the neural networks like tool for checking of the forms.

Neuronal architecture used is shown in the figure above (28 neurons in entry, 16 in the hidden layer and 28 neurons in the layer of exit), in entry assignment, the probabilities vectors obtained by the dynamic network Bayesian.

At exit a binary vector indicates the class of associated nature.

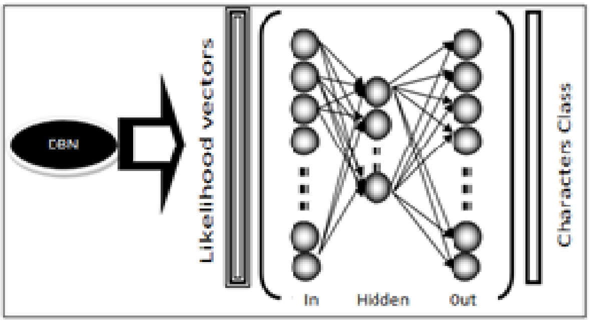

Fig. 5 Neuronal verifier operator

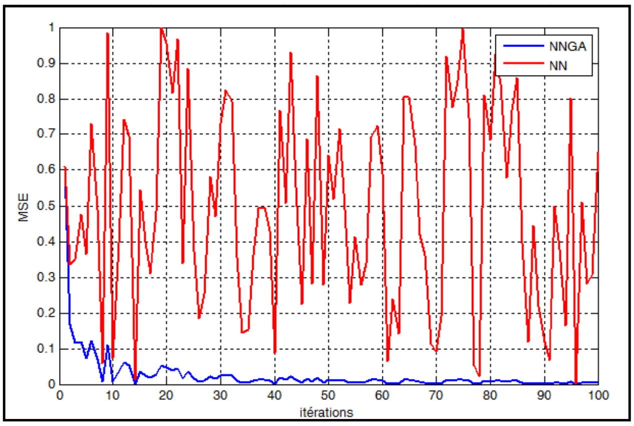

Fig. 6 Error evolution Graph

In Fig.6, one clearly notices of the average quadratic error convergence obtained starting from training neuronal NN.

The histogram below shows the confusion matrix of the recognition rates, for the Bayesian model with a neuronal verifier operator and one notices on the level of some characters such as “Alef” and “Ba” there is a total performance, but there are also non given characters such as 55% of the letter “Fa” and 65% of the letter “Jim” and that return to the probabilistic mechanism of the Bayesian networks. The total rate of recognition is 66.87%.

D. Dynamic Research Algorithm by Points Sets

The idea of this algorithm is to divide the writing signal in points sections, whereas this algorithm gives a segmentation to the each points section, the old algorithm consists in assigning to the each point level in the signal, a segmentation of this last, which is very slow for a on line recognition system. For this reason two approaches is proposed here:

1. Static Points Set SSPS

One fixes a number of static points for all the observations, (example: the total number of the points of the signal T=48, the beach P=3, old algorithm DRA buckles 47 times to give the final segmentation whereas the new loop requires that the number ((T/P)+1) is rounded with the higher close entireties).

2. Dynamic Points Set DSPS

One fixes here a percentage of points instead of a fixed number for all the observations, thus the number of points of the beaches changes according to the number T total of the points of one observation. (Example: the total number of the points of the T=48 signal, percentage of the Pr=5% points then beach P=((T.Pr)/100), this number is rounded with the higher close entireties). The idea of this new concept is summarized in Fig.7.

One implemented three approaches (DRA, DRA by static points set and DRA by dynamic points set) applied to the characters set of “Alef” to “Ya” and at exit one obtained the time of segmentation of the observation indicated (see Fig.7). The result shows effectively that the segmentation method based on the static points set is faster compared to old algorithm DRA. In addition, this approach by points set showed a better overall noticed speed. One can show this effectiveness because the basic concept, which gives a partition possibility by, points, set and not a single point.

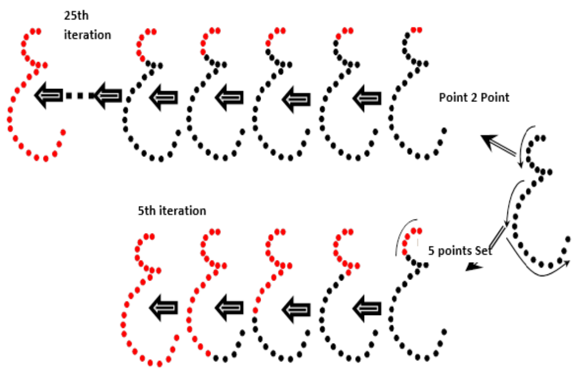

Fig. 7 Dynamic Research algorithm (DRA) by points sets

VI. Discussion and Comparison

The first to be realized is basic construction containing Arab letters samples. Indeed, in our first initial data base version, the minimal strategy was carried out a recognition system.

We implemented in the first experiment our DBN model and the results were acceptable but not completely reliable. We distinguished thereafter a parameter to treat its influence on the training and the test relative which is the network depth relating to architectural complexity. A d=3 value was selected in the second experiment like adequate value. For training hybrid system DBN-NN, a remarkable improvement containing test was quoted. What shows us, neuronal utility use of verifier operator with an internal optimization mechanism?

Another optimization approach in our system was underlined and carried out on the segmentation algorithm level. The segmentation by points set is a robust concept for a time saving of the data-processing realization.

In a comparison situation, Fig.8 shows the system progression of an improvement to another by fixing depth network at d=3 like better complexity topology parameter. Hybrid Model DBN-NN carried out the venial increase in the recognitions rates. Effectively, model DBN-DSPS pushed back the rates with justifiable and remarkable steps. The obtained recognition rates in the first experiment begin with 39.03% for the Kha character and pass for the Qaf character on average into 68.32% and reached the maximum rate of recognition 98.33% for the Alef character. It is interesting to note that on the level of the more share of the characters the rate did not exceed the 70%. Indeed from the modeling view point, the approach will accept other additions.

Cavity and thereafter, we thought an improvement hybrid step which appeared effective but not sufficient to reach the reliability desired on our system. And it is posted with small improvements rates for example: the characters Jim, Fa and Waw respectively give DBN to DBN-NN of the weak increases 1.41%, 2.60% and 1.02%.

This insufficiency pushed research towards the internal Bayesian network mechanism.

An improvement DBN-NN with DBN-DSPS reaches 49.10% for the Kha character and average by 21.32%. This optimization idea of segmentation algorithm was developed and it provided as awaited better results by report/ratio the traditional model.

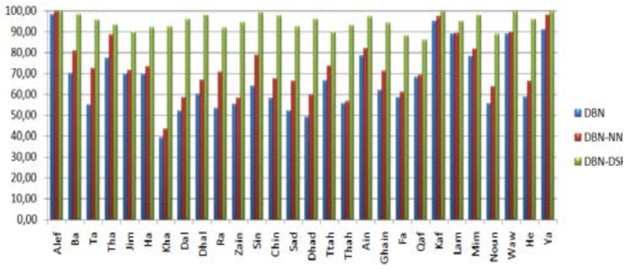

Fig. 8 Confusion matrix histogram for the three approaches

In general, the confusion matrixes in Fig.8 confirm practically these conclusions. With the difference in other systems like [14], the total recognition rate is equal 95.17% (higher rate at the rate provided by our system which equal to 94.79%) but this justifies that in this work, they did not use the totality characters forms and they used the characters classes which present the same geometrical form like Ba, Ta and Tha. In our system, we kept the characters without regrouping form and that will facilitate recognition task and will give more reliability and less treatment thereafter. Two objectives in this work are associated the data base creation and the recognition system realization of Arabic letters.

VII. Conclusion

In handwritten recognition, there are two distinct recognitions, with problems and solutions different: on line and off line handwriting recognition but there are combined between the two approaches. So that the character recognition isolated is strongly precise, it is significant to modeling the characters structure usually as possible. In this work, we considered that a character is composed of strokes and even their space relationships kept. The segmentation in stroke is not a single but it gave effectiveness.

The use of the graphic models, such as the dynamic Bayesian networks, us a made it possible to treat the characters isolated set by keeping their spatial information and the writing order from each character.

References

[1] M. Deviren. “Dynamic Bayesian networks for speech recognition”. In Proceedings of AAAI 2002, SIGART/AAAI Doctoral Consortium, Edmonton, Canada, 2002.
[2] G. Zweig. “Speech Recognition with Dynamic Bayesian Networks”. PhD thesis, University of California, Berkeley, Spring 1998.
[3] N. Friedman, K. Murphy, and S. Russell. “Learning the structure of dynamic probabilistic networks”. In UAI'98, Madison, Wisconsin, 1998.
[4] K.P. Murphy. “Dynamic Bayesian Networks: Representation, Inference and Learning”. PhD thesis, UC Berkeley, Computer Science Division, 2002.
[5] U. Kjaerulff. “A computational scheme for reasoning in dynamic probabilistic networks”. In Proceedings of the Eighth Conference on Uncertainty in Artificial Intelligence, pages 121-129, San Mateo, 1992. Morgan Kaufmann.
[6] L.R. Rabiner and R.W. Schafer. “Digital Processing of Speech Signals”. Prentice Hall, Englewood Cliffs, NJ, USA, 1978.
[7] F.V. Jensen, S.L. Lauritzen, and K.G. Olesen. „Bayesian updating in recursive graphical models by local computations”. Computational Statistics and Data Analysis, 1990.
[8] K.Nathan, H. Beigi, J. Subrahmonia, G.J. Clary, H. Maruyama, “Real-time on-line unconstrained handwriting recognition using statistical methods”, Proceedings of IEEE ICASSP, Detroit, USA, Vol. 4, 1995, pp. 2619-2622.
[9] T. Starner, J. Makhoul, R. Schwartz, G. Chou, “On-line cursive handwriting recognition using speech recognition methods”, Proceedings of IEEE ICASSP, Adelaide, Australia, Vol. 5, 1994, pp. 125-128.
[10] B.-K. Sin, J. Kim, Ligature modeling for online cursive script recognition, IEEE Trans. Pattern Anal. Mach. Intell. 19 (6) (1997) 623-633.
[11] H. Kim, J. Kim, “Hierarchical random graph representation of handwritten characters and its application to Hangul recognition”, Pattern Recognition 34 (2) (2001) 187-201.
[12] C.-L. Liu, I. Kim, J. Kim, “Model-based stroke extraction and matching for handwritten Chinese character recognition”, Pattern Recognition 34 (12) (2001) 2339-2352.
[13] I.-J. Kim, J. Kim, “Statistical utilization of structural neighborhood information for oriental character recognition”, Proceedings of the Fourth IAPR International Workshop on Document Analysis Systems, Rio de Janeiro, Brazil, 2000, pp. 303-312.
[14] N.Mezghani, A.Mitiche, and M.Cheriet, “Bayes classification of online Arabic characters by Gibbs modelling of class conditional densities", IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol 30, No. 7, pp. 1121-1131, 2008.