VOLUME 35 / NUMBER 3 / 2007

CONTROL AND INTELLIGENT SYSTEMS

(Continued on Back Cover)

ISSN 1480-1752

Published by ACTA PRESS

Control and Intelligent Systems, Vol. 35, No. 3, 2007

STRUCTURAL HIDDEN MARKOV MODELS BASED ON STOCHASTIC CONTEXT-FREE GRAMMARS

D. Bouchaffra* and J. Tan*

Abstract

We propose in this paper a novel paradigm that we named "structural hidden Markov model" (SHMM). It extends traditional hidden Markov models (HMMs) by considering observations as strings derived by a probabilistic context-free grammar. *These observations are related in the sense they all contribute to produce a particular structure.* SHMMs overcome the limit of state conditional independence of the observations in HMMs. Thus they are capable to cope with structural time series data. We have applied SHMM to data mine customers' preferences for automotive designs. A 5-fold cross-validation has shown a 9% increase of SHMM accuracy over HMM.

Key Words

Hidden Markov models, stochastic context-free grammars, structural information, fusion of statistics and syntax

1. Introduction

Hidden Markov models (HMMs) have been used since their first success in speech processing and recognition in the late 1980s [1–4]. Neighbour areas such as signal processing [5], and handwriting and text recognition [6] have also benefited almost at the same time from these stochastic models. HMMs spread to many other areas such as image processing and computer vision [7], biosciences [8], control [9], and others. However, the number of problems where HMMs can be applied is insignificant compared to all the problems we can encounter. *The main reason comes from the fact that HMMs have a clear conceptual framework and the ability to learn statistically, but they are unable to account for structural information of the sequence [10, 11].* The symbols of an input sequence are assumed to be state conditionally independent. Therefore, HMMs make no use of structure, either topological or conceptual [12]. This lack of structure inherent to standard HMMs has drastically limited the recognition and classification tasks of complex patterns. The reason is that a pattern contains some *relational information* from which it is difficult and sometimes impossible to derive an appropriate feature vector. *Therefore, the analytical approaches which process the patterns only on a quantitative basis but ignore the interrelationships (or structure) between the components of the patterns quite often fail.* Cai and Liu's approach integrates the statistical and structural information for unconstrained handwritten numeral recognition. Their method uses macro-states to model pattern structures [13]. However, besides the fact that this method uses statistical and structural information in two different steps, their methodology is application driven and therefore is *very specific*. Zhu and Garcia-Frias proposed two novel generative methods which make use of probabilistic context-free grammars and HMMs respectively to model the end-to-end error profile of radio channels [14]. However, they did not provide a tool to merge standard HMMs with probabilistic context-free grammars into a single probabilistic framework.

Because of the gap between statistics and syntax, we introduce in this paper a novel paradigm—*structural hidden Markov model* (SHMM), that embed grammatical rules to identify structural information. The structures are built through conclusions (or variables of a grammar) that accept the input strings (sequences of observations). In other words, statistics controls the distribution of the visible observation sequence whereas syntax informs about what these observations are forming as a whole: it is their structure.

The organization of this paper is as follows: Section 2 introduces the concept of SHMM. The problems assigned to an SHMM are defined in Section 3. An application of SHMM as well as experiments are explained in Section 4. Finally, the conclusion and the future work are laid in Section 5.

2. The Concept of SHMM

In this section, we introduce a mathematical description of the SHMM concept that goes beyond the traditional HMM as it emphasizes the structure (or syntax) of the visible sequence of observation.

* Department of Computer Science and Engineering, Oakland University, Rochester, MI, 48309, USA; e-mail: {bouchaff, jtan}@oakland.edu

Recommended by Prof. May-Win Thein
(paper no. 201-1665)

In traditional HMMs, the visible observations are assumed to be *state conditionally independent*. Let $O = (o_1 o_2 \cdots o_T)$ be the observation sequence of length T and $q = (q_1 q_2 \cdots q_T)$ be the state sequence where q_1 is the initial state. Given a model λ, we can write: $P(O \mid \lambda) = \sum_{all\ q} P(O, q \mid \lambda)$, and $P(O, q \mid \lambda) = P(O \mid q, \lambda) \times P(q \mid \lambda)$, and using state conditional independence, we obtain

$$P(O \mid q, \lambda) = \prod_{t=1}^{T} P(o_t \mid q_t, \lambda).$$

However, there are several settings where the conditional independence assumption doesn't hold. For example, while standard HMMs perform well in recognizing amino acids and consequent construction of proteins from the first level structure of DNA sequences [15, 16], they are inadequate for predicting the secondary structure of a protein. The reason for the inadequacy comes from the fact that the same order of amino acid sequences have different folding modes in natural circumstances [8]. Therefore, there is a need to balance the loss incurred by this state conditional independence assumption.

Our thrust is that a complex pattern O can be viewed as a sequence of constituents O_i made of strings of symbols interrelated in some way.[1] Therefore, each observation sequence O is not only one sequence in which all observations are conditionally independent, but a sequence that is divided into a series of s strings $O_i = (o_{i_1} o_{i_2} \cdots o_{i_{r_i}})$ $(1 \le i \le s)$. The symbols of a string are related in the sense that they define a local structure C_j of the whole complex pattern. This structural information is captured through a probabilistic context-free grammar where the symbols o_i are evidences that contribute to the production of a structure C_j. For example, a cloud of points representing a sequence of observations O_i forms a round shape C_j with a certain probability $P(C_j \mid O_i)$. Similarly a sequence of phonemes produces a word with a certain probability depending on the context. The higher the complexity of a pattern, the higher the number of structures needed to describe this pattern locally. Furthermore, the statistical information is expressed through the probability distribution of the structural information sequence that describes the whole pattern. Therefore, statistics and syntax are merged together in one single framework. If $O = (O_1, O_2, \ldots, O_s) = (o_{1_1} o_{1_2} \cdots o_{1_{r_1}},\ o_{2_1} o_{2_2} \cdots o_{2_{r_2}}, \ldots, o_{s_1}, o_{s_2}, \ldots, o_{s_{r_s}})$ (where r_1 is the number of observations in subsequence O_1 and r_2 is the number of observations in subsequence O_2, etc.) and $C = (C_1, C_2, \ldots, C_s)$, then the probability of a complex pattern O given a model λ can be written as: $P(O \mid \lambda) = \sum_C P(O, C \mid \lambda)$. Therefore, we need to evaluate $P(O, C \mid \lambda)$. As the model λ is implicitly present during the evaluation of this joint probability, it is omitted. Thus we have:

$$\begin{aligned} P(O,C) &= P(C,O) = P(C \mid O) \times P(O) \\ &= P(C_1 C_2 \cdots C_S \mid O_1 O_2 \cdots O_S) \times P(O) \\ &= P(C_S \cdots C_2 C_1 \mid O_S \cdots O_2 O_1) \times P(O) \\ &= P(C_S \mid C_{S-1} \cdots C_2 C_1 O_S \cdots O_1) \\ &\quad \times P(C_{S-1} \cdots C_2 C_1 \mid O_S \cdots O_1) \times P(O) \qquad (1) \end{aligned}$$

[1] In other words, it is possible to decrease the resolution level of a complex pattern.

We assume that C_i depends only on O_i and C_{i-1} (as illustrated in Fig. 2.), and the structure probability distribution is a Markov chain of order 1. The reason behind this Markovian assumption comes from cognitive science. In fact, it is well established that when we perform an object recognition task, our brain relies partly on local interactions between sub-patterns describing these objects [17]. Local interactions can also be expressed statistically by the means of Markovian fields using Gibbs distributions [18, 11]. However, as pointed out in the introduction, our approach considers exclusively sequential processes that remain within the context of HMMs. Finally, we can recursively approximate (1) as:

$$P(O,C) \approx \prod_{i=1}^{S} P(C_i \mid O_i, C_{i-1}) \times P(O) \qquad (2)$$

We now evaluate $P(C_i \mid O_i, C_{i-1})$ as follows:

$$\begin{aligned} P(C_i \mid O_i, C_{i-1}) &= \frac{P(O_i C_{i-1} \mid C_i) P(C_i)}{P(O_i C_{i-1})} \\ &= \frac{P(O_i \mid C_{i-1} C_i) P(C_{i-1} \mid C_i) P(C_i)}{P(O_i \mid C_{i-1}) P(C_{i-1})}. \end{aligned}$$

As O_i does not depend on C_{i-1}, we have:

$$\begin{aligned} P(C_i \mid O_i, C_{i-1}) &= \frac{P(O_i \mid C_i) P(C_{i-1} \mid C_i) P(C_i)}{P(O_i) P(C_{i-1})} \\ &= \frac{P(C_i \mid O_i) P(O_i) P(C_i \mid C_{i-1}) \times P(C_{i-1}) P(C_i)}{P(C_i) P(C_i) P(O_i) P(C_{i-1})} \\ &= \frac{P(C_i \mid O_i) P(C_i \mid C_{i-1})}{P(C_i)}. \qquad (3) \end{aligned}$$

From (2) and (3), we have:

$$P(O,C) \approx \prod_{i=1}^{S} \frac{P(C_i \mid O_i) P(C_i \mid C_{i-1})}{P(C_i)} \times P(O) \qquad (4)$$

Now we can define an SHMM as follows:

Definition 1. A structural hidden Markov model is a quintuple $\lambda = (\pi, \mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D})$, where π is the initial state probability vector, $\mathcal{A}$ is the state transition probability matrix, $\mathcal{B}$ is the state conditional probability matrix of the visible observations, $\mathcal{C}$ is the posterior probability matrix of a structure given a sequence of observations, $\mathcal{D}$ is the structure transition probability matrix.

An SHMM is characterized by the following elements:

- **N**, the number of hidden states in the model. We label the individual states as $1, 2, \ldots, N$, and denote the state at time t as q_t.
- **M**, the number of distinct observations o_i.

- π, the initial state distribution, $\pi=\{\pi_i\}$, where $\pi_i=P(q_1=i)$ and $1\leq i\leq N$, $\sum_i \pi_i=1$.
- $\mathcal{A}$, the state transition probability distribution matrix, $A=\{a_{ij}\}$, where $a_{ij}=P(q_{t+1}=j\,|\,q_t=i)$ and $1\leq i,j\leq N$, $\sum_j a_{ij}=1$.
- $\mathcal{B}$, the state conditional probability matrix of the observations, $B=\{b_j(k)\}$, in which $b_j(k)=P(o_k\,|\,q_j)$, $1\leq k\leq M$ and $1\leq j\leq N$, $\sum_k b_j(k)=1$.
- **F**, the number of distinct structures.
- $\mathcal{C}$ is the posterior probability matrix of a structure given its corresponding observation sequence, $\mathcal{C}=P(C_j\,|\,O_i)=c_i(j)$. For each O_i, we have: $\sum_j c_i(j)=1$.

  A particular application requires a particular grammar G_h in which a structure C_j is assigned to O_i via a rule R_k which is written as: G_h: $C_j \stackrel{R_k}{\Longleftarrow} (O_i=o_{i_1},o_{i_2},\dots,o_{i_r})$. As we are using a probabilistic context-free grammar, there is only one structure (the most likely) that is assigned to a string O_i.
- $\mathcal{D}$, the structure transition probability matrix. $\mathcal{D}=\{d_{ij}\}$, where $d_{ij}=P(C_{t+1}=j\,|\,C_t=i)$, $\sum_j d_{ij}=1$, $1\leq i,j\leq F$.

Fig. 1 depicts a representation of an SHMM. We now define the problems that are involved in an SHMM.

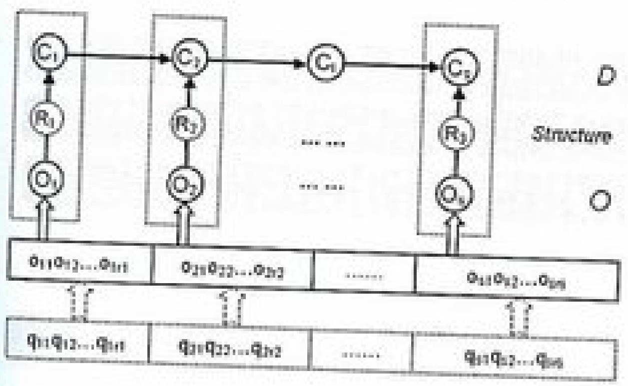

Figure 1. A graphical representation of an SHMM.

3. Problems Assigned to an SHMM

There are four problems that are assigned to an SHMM: (i) probability evaluation, (ii) statistical decoding, (iii) structural decoding, and (iv) parameter estimation. We will show how to solve them in this section.

3.1 Probability Evaluation

Given a model $\lambda=(\pi,\mathcal{A},\mathcal{B},\mathcal{C},\mathcal{D})$ and $O=(O_1,\dots,O_s)$, an observation sequence, we evaluate how well does the model λ match O. From (4), this probability can be expressed as:

$$P(O\,|\,\lambda)=\sum_C P(O,C\,|\,\lambda)=\sum_C\left\{\prod_{i=1}^{s}\frac{c_i(i)\times d_{i-1,i}}{P(C_i)}\times\Phi\right\} \quad (5)$$

where $\Phi=\sum_q \pi_{q_1}b_{q_1}(o_1)a_{q_1q_2}b_{q_2}(o_2)\dots a_{q_{T-1}q_T}b_{q_T}(o_T)$.

3.2 Statistical Decoding

In this problem, we determine the optimal state sequence $q^*=\arg\max_q(P(O_i,q\,|\,\lambda))$ that best "explains" the sequence of symbols within O_i. This problem is similar to problem (ii) of the traditional HMM and can be solved using Viterbi [1] algorithm as well.

3.3 Structural Decoding

This is the most important problem as we attempt to determine the "optimal structure of the model". An example of an optimal sequence of structures[2] is: <round, curved, straight, ..., slanted, ..., >. This sequence of structures helps describing objects. Autonomous robots based on this learning machine can for example be trained to recognize the components of a human face described as a sequence of shapes such as: <round (human head), vertical line in the middle of the face (nose), round (eyes), curved (mouth), ..., >. Similarly, a customer's opinion of an automobile is composed of his/her opinion of the front, the side and the rear of this automobile. These partial opinions describe the whole external view of the automobile and impact significantly the customer's purchase decision.

In this problem, we determine the optimal structure sequence $C^*=<C_1^*,C_2^*,\dots,C_t^*>$ such that:

$$C^*=\arg\max_C(P(O,C\,|\,\lambda))$$

We define:

$$\delta_t(i)=\max_C P(O_1,O_2,\dots,O_t,C_1,C_2,\dots,C_t=i\,|\,\lambda)$$

that is, $\delta_t(i)$ is the highest probability along a single path, at time t, which accounts for the first t strings and ends in structure i. Then, by induction we have:

$$\delta_{t+1}(j)=\left[\max_i \delta_t(i)d_{ij}\right]c_{t+1}(j)\frac{P(O_{t+1})}{P(C_j)} \quad (6)$$

Similarly, this latter expression can be computed using *Viterbi* algorithm. However, we estimate δ in each step *through the structure transition probability matrix*. This optimal sequence of structures describes the structural pattern piecewise.

3.4 Parameter Estimation

3.4.1 Statistical Parameters

In this problem, we try to optimize the model parameters $\lambda=(\pi,\mathcal{A},\mathcal{B},\mathcal{C},\mathcal{D})$ to maximize $P(O\,|\,\lambda)$. The re-estimation phase of the parameters $\{\pi_i\}$, $\{a_{ij}\}$, $\{b_j(k)\}$ and $\{d_{ij}\}$ is conducted as in traditional HMMs, using the Baum-Welch optimization technique [2].

[2] Called conclusions within a grammar context.

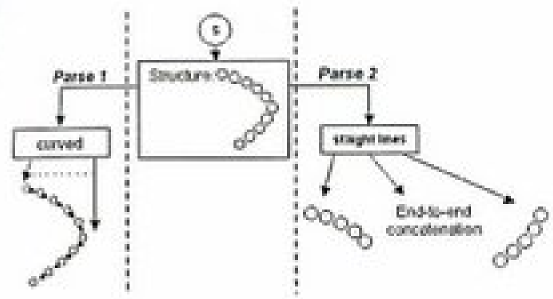

Figure 2. Two different interpretations (parses) of a structure using the context-free grammar.

3.4.2 Structural Parameters

The most difficult problem of SHMM parameter estimation is the evaluation of $P(C_j \mid O_i)$. To estimate this term, we used a "stochastic context-free grammar" (SCFG) [19, 20] that recognizes structures. In this SCFG each production is augmented with a probability. The probability of a derivation is then the product of the probabilities of the productions used in that derivation. For example, in Fig. 2, the structure "S" could be interpreted as "curved" (Parse 1) with a certain probability or "straight lines" (Parse 2) with another probability. Depending on the application at hand, one interpretation could be more plausible than the other. Similar to the decoding process, a Viterbi algorithm is used to find the best parse. Usually, it is the user who constructs an appropriate grammar based on personal knowledge and experience regarding a particular application. The probabilities of rules are estimated from data using the maximum likelihood criterion [21]. To construct such a grammar, a set of primitives is selected depending on the type of data involved in the application. *The primitives should provide a reasonable description of the patterns and their structural relations.* An incoming string O_i derives a structure C_j, this derivation is expressed as: $C_j \overset{R_n}{\Leftarrow} \alpha_{n-1} \ldots \overset{R_3}{\Leftarrow} \alpha_2 \overset{R_2}{\Leftarrow} \alpha_1 \overset{R_1}{\Leftarrow} O_i$, where R_i are the production rules, α_i $(1 \le i \le n-1)$ is an intermediate step of the derivation from O_i to structure C_j and α_{i+1} is obtained from α_i. We consider each step of the derivation as a rule R_i. The production rule R_i is activated with the probability $P(R_i)$. Assuming all rules are independent, we can estimate $P(C_j \mid O_i)$ as: $P(C_j \mid O_i) \approx \prod_{i=1}^{n} P(R_i)$, where $P(R_i)$ is estimated through the maximum likelihood.

4. Application: Customer Preferences' Prediction

Automotive companies place a great emphasis on exterior appearance design. They always make real-size models of a car's exterior design, show them to a lot of customers, and collect survey data of their opinions and feelings. This feedback is then sent to the design department. According to the information contained in the feedback, the exterior design engineers refine their designs. The purpose of this application is to build a computational method that helps engineers to improve their designs, speed up their job by releasing them from the tedious manual information processing, and eventually make cars that match the need of customers.

4.1 Automotive Data Collection and Clustering

We collected 500 images of regular cars (no trucks or vans) with their three exterior views (front, side, and rear, i.e., 1500 images). A pre-processing phase of car images has been performed to eliminate some features such as color, lamp shapes, and tires. We painted the body with white background color, and extracted the contours of the three views. Then we presented these contours to 300 university students and asked them to give their opinions on the three views of a car viewed separately. Opinions are adjectives that express students' feelings of the car views at first sight. Every contour is assigned different opinions by different students; 300 students would probably give as much as 300 different opinions to one contour. We adopted the "majority voting" method to obtain a unique opinion that is assigned to a contour. Thus we obtained 1500 adjectives (some of them are identical) clustered with synonymy using the online lexical reference system WordNet [22]. Each centroid of a cluster is called a *perception*. Each respondent's opinion (adjective) belongs to one and only one perception.

Because it is very difficult to acquire a large automotive data set, we generated 10 artificial samples of size 500 using the *bootstrap* re-sampling technique [23]. Then, we combined them and obtained an artificial data set containing 10 times the data as our original data set, which means that we have 5000 strings of contours and 5000 conclusions for "front", "side", and "rear" views respectively. Finally, we divided this generated data set into two parts, two-thirds for training and one-third for testing.

4.2 Chain-Code Representation and Local Structures

The observations are the automotive contours represented by chain-code [24] strings. We used the standard implementation of the chain code based on the eight directions (0–7). Each local structure C_j is a certain shape represented by a subsequence of the contour chain-code string. To find the local structures, contours are segmented as sequence of O_i using the shape convexity criterion. The sign of the second derivative is computed for each point located on the contour. Then three rules were used to determine where to segment the contour: *(1) If the second derivative changes its sign on a particular point of the contour, then this point is a boundary of the segment, otherwise we continue to extend the segment. (2) If the second derivative of a point is 0, then we consider its sign remains either positive or negative as the second derivate of its preceding point on the contour. (3) An exception of rule (1) is: if the length of a segment is less than threshold "6", then we consider this segment as an extension of its preceding segment. If this segment is the first on the contour, then we concatenate it with its following segment.*

Fig. 3 shows how these three rules work. The black points form segment 1. The gray points form segment 2.

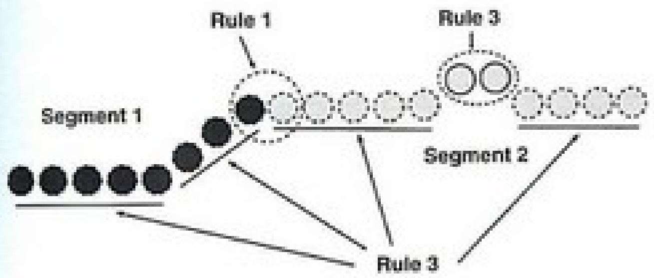

Figure 3. This example shows how two segments of a piece of a contour are formed according to the three rules.

The two gray points with solid boundaries are in segment 2.

4.3 Training and Testing Results of SHMM

Once the optimal segmentation has been computed for the training data, we partitioned the set of segments into classes of equivalences. The partitioning was done as follows:

- Computer the edit distance between each pair of chain-code strings of segments.
- Cluster the segments. Each cluster is a class of equivalence.
- Determine the representative value of each equivalence class. The representative value of a class is a chain-code string that has the shortest average distance to all other members of that class.

For a segment O_i, we computed its edit distances to all representative values of classes of equivalence. We chose the shortest one, and let its corresponding class of equivalence be C_j. If the number of segments in C_j is L, then $c_i(j)$ is estimated by: $c_i(j) = N_L/L \times 100\%$, where N_L is the number of O_i's nearest L neighbours that are in C_j. Thus, matrix $\mathcal{C}$ was constructed after partitioning. A contour is represented by a sequence of local structures after partitioning. Then we used the Baum-Welch optimization technique to estimate matrix $\mathcal{D}$. Other parameters, π, $\mathcal{A}$, $\mathcal{B}$, were estimated like in traditional HMMs.

Once the SHMM for this application is built, we used the testing data set to evaluate the accuracy. We have selected five perception categories in this application which are the five clusters of adjectives obtained using the lexical database WordNet. Therefore five models λ_i have been generated, and each model is built to learn one perception category. The number of samples in each category is the cardinal of each cluster. Currently the categories we have obtained and their numbers of samples are: ugly—165, ordinary—323, nice—60, attractive—487, beautiful—465. The best model λ^* for each side is computed via: $\lambda^* = \arg\max_{\lambda_i} P(O \mid \lambda_i)$.

If the predicted model is λ_p and the true model is λ_t, then our precision is defined as: $Precision = \sum \delta(\lambda_p - \lambda_t) / \mid input\ patterns \mid$, where $\delta(x-a)$ is the Kronecker symbol which is "1" if $x = a$, and "0" otherwise. The denominator $\mid input\ patterns \mid$ represents the total number of patterns. The numerator is the number of correctly classified contours and the denominator is the number of all three exterior view contours in the testing data set.

We have compared the SHMM approach with the traditional HMM classification technique. The accuracy computation in the case of the HMM is based on the comparison between the predicted category and the true category (from survey) for each view *separately*. However, the accuracy in the case of the SHMM is based on the comparison of the predicted and the true sequence assigned to the three views *at once*. The design engineers are interested in discovering the flaws from the three views separately rather than from the whole car. For example, in the case of HMM, if the front view perception of *Cadillac ext* is predicted as "attractive" while the true category as "ordinary" then we have an error of classification. Thus, the HMM was applied to each view of the car separately. Each view contributes to the prediction result without interfering with other views.

We divided the images of the 500 cars into five sets, each of which contains images of 100 cars. Then we selected four sets for training and the remaining one for testing. We repeated this procedure 5 times with each time selecting a different set for testing. Table 1 shows the accuracy of each round and the overall accuracy.

Table 1
Performance Comparison Between HMM and SHMM Classifiers in Five Rounds of Training and Testing

Model Round	HMM (%)	SHMM (%)
1	73	81
2	78	80
3	66	85
4	70	79
5	73	82
Overall	72	81.4

5. Conclusion and Future Work

We have presented a novel mathematical paradigm that extends traditional HMMs by merging syntactical and statistical information into a single probabilistic framework. Our approach relates visible observations through their contribution to a same structure of a syntactic rule. Doing so, SHMMs bypass the state conditional independence assumption inherent to traditional hidden Markov modeling. The SHMM represents a preliminary fusion between statistics and syntax [25]. The automotive application shows that the SHMM concept is promising as it has outperformed the traditional HMM classifier. However, this is an undergoing research, more data need to be collected, and comparisons with other classifiers are necessary to measure the global contribution of SHMMs.

References

[1] L.R. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition, *Proc. of the IEEE*, 77(2), Philadelphia, PA, 1989, 257–286.

[2] L. Rabiner & B.H. Juang, *Fundamentals of speech recognition* (Heidelberg: Prentice Hall, Signal Processing Series, Alan V. Oppenheim Series, 1993).

[3] M.J.F. Gales, Cluster Adaptive Training of Hidden Markov Models, *IEEE Trans. on Speech and Audio Processing*, *8*(4), 2000.

[4] I. Sanches, Noise-compensated hidden Markov models, *IEEE Trans. on Speech and Audio Processing*, *8*(5), 2000.

[5] G. Fan & X.G. Xia, Improved hidden Markov models in the wavelet-domain, *IEEE Trans. on Signal Processing*, *49*(1), 2001.

[6] D. Bouchaffra, V. Govindaraju, & S.N. Srihari, Postprocessing of recognized strings using nonstationary Markovian models, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, *PAMI*, 21(10), 1999.

[7] J. Li, A. Najmi, & R.M. Gray, Image classification by a two-dimensional hidden Markov model, *IEEE Trans. on Signal Processing*, *48*(2), 2000, 517.

[8] K. Asai, S. Hayamizu, & H. Handa, Prediction of protein secondary structures by hidden Markov models, *Computer Application in the Biosciences (CABIOS)*, *9*(2), 1993, 141–146.

[9] D. Hernandez-Hernandez, S.I. Marcus, & P.J. Fard, Analysis of a risk-sensitive control problem for hidden Markov chains, *IEEE Trans. on Automatic Control*, *44*(5), 1999, 1093.

[10] R. Duda, P. Hart, & D. Stork, *Pattern classification* (New York: Wiley, 2001).

[11] B.D. Ripley, Pattern recognition and neural networks (New York: Cambridge University Press, 1996).

[12] M.C. Gemignani, *Elementary topology*, Second edition (New York: Dover Publications, Inc 1990).

[13] J. Cai & Z.Q. Liu, Integration of structural and statistical information for unconstrained handwritten numeral recognition, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, *PAMI*, *21*(3), 1999.

[14] W. Zhu & J.G. Frias, Stochastic context-free grammars and hidden Markov models for modeling of bursty channels, *IEEE Transactions on Vehicular Technology*, *53*(3), 2004.

[15] A. Krogh, M. Brown, I.S. Mian, K. Sjolander *et al.*, Hidden Markov models in computational biology: Applications to protein modeling, *Journal Molecular Biology*, *235*, 1994, 1501–1531.

[16] S.R. Eddy, Profile hidden Markov models, *Bioinformatics*, *14*(9) 1998, 755–763.

[17] J.A. Fodor, & Z.W. Pylyshyn, Connectionism and cognitive architecture: A critical analysis, *Cognition*, *28* 1988, 3–71.

[18] F. Bartolucci & J. Besag, A recursive algorithm for Markov random fields, *Biometrika*, *89*(3), 2002, 724–730.

[19] D. Jurafsky, C. Wooters, J. Segal, A. Stolcke *et al.*, Using a stochastic context-free grammar as a language model for speech recognition, *Proc. ICASSP'95*, Detroit, USA, 1995, 189–192.

[20] A. Stolcke, An efficient probabilistic context-free parsing algorithm that computes prefix probabilities, *Computational Linguistics*, *21*(2), 1995, 165–201.

[21] Zhiyi Chi, Stuart Geman, Estimation of probabilistic context-free grammars, *Computational Linguistics*, *24*(2), 1998.

[22] C. Fellbaum, WordNet: An electronic lexical database, Bradford Book (Cambridge, MA: MIT Press, 1998).

[23] B. Efron, *In the jackknife, the bootstrap and other resampling plans* (Philadelphia: SIAM, 1982).

[24] H. Freeman, On the encoding of arbitrary geometric configurations, *IRE Trans. on Electronic Computers*, vol. EC-10, June 1961, 260–268.

[25] D. Bouchaffra & J. Tan, The concept of structural hidden Markov models: Application to mining customers' preferences for automotive designs, *17th Int. Proc. Conf. on Pattern Recognition (ICPR)*, Cambridge, UK, 23–26 August 2004.

Biographies

Djamel Bouchaffra is an Assistant Professor in the Computer Science and Engineering Department at Oakland University, Michigan. His area of expertise is pattern recognition, machine learning, artificial intelligence and soft computing. His current research consists of developing mathematical models that merge statistics with logic (or syntax). His goal is to study, analyze, and classify complex patterns where structure is predominant. Prof. Bouchaffra is an Associate Editor for the journal of "*Pattern Recognition*" published by Pergamon. He has written several papers in premier journals (such as *IEEE TPAMI*) and conferences (such as *CVPR*, *ICPR*). He is a regular reviewer for *IEEE* journals such as *PAMI* and *NN* transactions. He was one of the general chairs and organizers of the *IEEE ICSIT'2005* conference. He has chaired several sessions in pattern recognition conferences. He is the Director of the Pattern Recognition and Machine Intelligence (PARMIL) Laboratory. Dr. Bouchaffra is a Senior member of IEEE. More information about Dr. Bouchaffra's activities can be found in: http://www.oakland.edu/~bouchaff

Jun Tan received his MS degree in Computer Science in 2001 from Nanjing University, China. He is now a PhD candidate in the Department of Computer Science and Engineering at Oakland University. His research interests include structural hidden Markov model, data mining, medical image processing, etc.

For Complete information regarding IASTED conferences, please contact the IASTED Secretariat office

Building B6, Suite #101
2509 Dieppe Ave SW
Calgary, AB, T3E 7J9
Canada

Tel: 403-288-1195
Fax: 403-247-6851
E-mail: calgary@iasted.com
Website: www.iasted.org

(Continued from Front Cover)