Adaptive Online Multi-stroke Sketch Recognition Based on Hidden Markov Model

Chapter *in* Lecture Notes in Computer Science · January 2005

DOI: 10.1007/11739685_99 · Source: DBLP

Zhengxing Sun
Nanjing University

Adaptive Online Multi-stroke Sketch Recognition Based on Hidden Markov Model

Zhengxing Sun, Wei Jiang, and Jianyong Sun

State Key Lab for Novel Software Technology, Nanjing University, 210093, Nanjing
szx@nju.edu.cn

Abstract. This paper presents a novel approach for adaptive online multi-stroke sketch recognition based on Hidden Markov Model (HMM). The method views the drawing sketch as the result of a stochastic process that is governed by a hidden stochastic model and identified according to its probability of generating the output. To capture a user's drawing habits, a composite feature combining both geometric and dynamic characteristics of sketching is defined for sketch representation. To implement the stochastic process of online multi-stroke sketch recognition, multi-stroke sketching is modeled as an HMM chain while the strokes are mapped as different HMM states. To fit the requirement of adaptive online sketch recognition, a variable state-number determining method for HMM is also proposed. The experiments prove both the effectiveness and efficiency of the proposed method.

1 Introduction

Sketching is a natural input mode to help us convey ideas and guide our thinking process both by aiding short-term memory and by helping to make abstract problems more concrete [1]. Numerous researchers have been working on the subject of sketch recognition for many years either as a natural input modality [2][3][4] or to recognize composite sketches [5][6][7]. They can be mainly classified into two categories: feature-based and graph-based. Feature-based methods make use of some local or global features of sketchy shapes for sketch recognition. For example, Rubine [3] defined a gesture characterized by a set of eleven geometric attributes and two dynamic attributes. Fonseca et al [5] proposed a method of symbol recognition using fuzzy logic based on a number of rotation invariant global features. As one of the most prominent approaches to object representation and matching, graph-based methods have been recently applied to hand-drawn pattern recognition problems, such as in [6][7], where sketch recognition is formulated as a graph isomorphism problem. However, the poor efficiency of these recognition engines is always frustrating, especially for the newly added users and the multi-stroke sketchers. The difficulty comes from the fact that sketching is usually informal, inconsistent and ambiguous both in intra-person and inter-person settings in a given situation. To capture a user's sketching habit, adaptive sketch recognition is required [8], where the recognition engine should be trainable and adaptable to a particular user's drawing styles, especially for the multi-stroke sketchy shapes.

D.S. Yeung et al. (Eds.): ICMLC 2005, LNAI 3930, pp. 948 – 957, 2006.

Obviously, one solution for adaptive sketch recognition is to construct appropriate classifiers based on machine learning. In our previous researches, we have developed an adaptive sketch recognition method based on incremental SVM learning [9]. It can actively analyze the users' incremental data, and can largely reduce the workload of artificial labeling and the classifier's training time. While it has been proven to be both effective and efficient in our experiments, it can still deal with only single-stroke sketches since the dimension of feature vectors of SVM must be fixed for all shapes.

Hidden Markov Model (HMM) is one of the most successful stochastic modeling tools that have been used in the analysis of non-stationary time series [10]. It has been used with great success in the stochastic modeling of speech [10][11] for years. In recent years, it has also been widely used in handwriting recognition [12][13][14]. In this paper, we will present our experiments in adaptive online multi-stroke sketch recognition in terms of HMM, where we view the drawing pattern as the result of a stochastic process that is governed by a hidden stochastic model and identified according to its probability of generating the output, inspired by its success in speech recognition and handwriting recognition.

The rest of the paper is organized as follows: in Section 2, the principle of our method of adaptive online multi-stroke sketch recognition based on HMM is introduced in detail, including feature representation for multi-stroke sketchy shape, modeling multi-stroke sketching with HMM and determination of HMM state-number for adaptive sketch recognition. Some experiment results are evaluated in Section 3 and conclusions are given in the final section.

2 Adaptive Sketch Recognition Based on Hidden Markov Model

2.1 Feature Representation for Multi-stroke Sketching

There have been many features used for representing the characteristics of a sketchy shape, including "Rubine features" [3], "turning function" [8], "curvature" and "pen speed" [6], "normalized curvature" [12], "centroidal radius" [5], "intersection type" and "number of strokes" [7], and so on. Some of them are prominent in describing the local characteristics of graphical symbols, such as "intersection type" and "number of strokes"; some are outstanding in outlining the global structure of symbols, such as "curvature"; and some may be only adaptable to simple-structural graphics or one-stroke drawn symbols, such as "Rubine features" and "turning function".

During our research experiments [1][7][8][9], we have realized that the sketchy shape in multi-stroke sketch recognition is closely related to both the symbol structures and peoples' drawing habits. That is to say, features used to represent sketches must include both the geometrical features of symbols and the dynamic features of a user's drawing. Accordingly, we consider the features selected for online multi-stroke sketch recognition must satisfy the following three criteria:

1. the features must contain both geometric (spatial) and dynamic (temporal) characteristics of a sketchy shape,
2. the features must be able to represent the spatial relationships between strokes, and

3. the features do not need to depict too much detailed local information of the symbols.

To satisfy these criteria, we define a composite feature in a seven-dimension vector, as shown in Table 1, which combines a few geometrical and dynamic features often used in graphics recognition, such as “centroidal radius”, “curvature”, “speed” and their means and standard variances. Each of them is briefly described as follows: “Centroidal radius” is the cumulated distances between every point in each of the strokes and the centroid of the graphical symbol. It can describe the characters of an engineering sketch. In our experiments, we choose only 20 points that are uniformly distributed on each of the strokes. “Pen speed” represents the ratio of distance between the current point and the previous point to the time spent during drawing the two points. “Curvature” indicates the cosine of the corner angle at the current point between two lines connected respectively to the previous point and the next point. In addition, we define a “pen-direction” with the slope of a virtual line, which connects the end-point of the previous stroke to the start-point of the current stroke, as a one-dimension vector to reflect the tendency of pen-movement between two continuous strokes.

Table 1. Component of our Composite Feature

Feature	Feature Description	Feature Characteristics	
f_1	Mean of centroid-radius	Global	Geometric
f_2	Standard deviation of centroid-radius	Global	Geometric
f_3	Mean of all pen speeds	Global	Dynamic
f_4	Standard deviation of all pen speeds	Global	Dynamic
f_5	Mean of all curvatures	Global	Geometric
f_6	Standard deviation of all curvatures	Global	Geometric
f_7	Pen-direction between two continuous stokes	Local	Dynamic

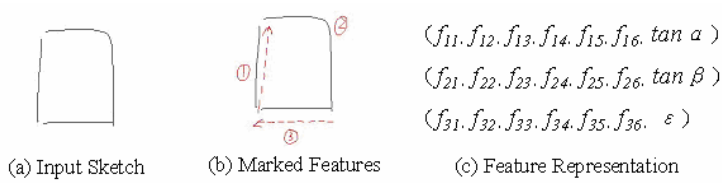


Fig. 1. An example of the definition of our composite feature

During the training and recognition stage, we extract combined features from each of the strokes. The seventh feature is a variable vector according to the stroke type. If the stroke is not the last stroke of the sketch, the seventh feature is the “pen-direction”. On the contrary, the seventh feature is the perimeter ratio of the sketch to its closure. Fig. 1 illustrates the definition of our composite feature. Fig. 1(a) shows a sketch drawn by a user. Fig. 1(b) indicates that a sketch is composed of three strokes, where the left dashed arrow represents the "pen direction" between the first stroke and the second, the bottom dashed arrow represents the "pen direction" between the second stroke and the third, and their elevations are α and β respectively. The perimeter ratio of the sketch to its closure is defined as ε. The vector representation of the sketch by combined feature is shown in Fig. 1(c).

2.2 Modeling Multi-stroke Drawing with a Hidden Markov Model

In Hidden Markov Models, the observed pattern is viewed as the result of a stochastic process that is governed by a hidden stochastic model. Each stochastic model represents a different class pattern capable of producing the observed output. The goal is to identify the model that has the highest probability of generating the output. One aspect that distinguishes Hidden Markov Models is their strong temporal organization; processes are considered to be the result of time-sequenced state transitions in the hidden model and the expectation of a particular observation is dictated by the current state in the model and (usually) the previous state.

In online sketch recognition, drawing sketches, especially drawing multi-stroke sketches, can be regarded as a time-sequenced process. Different users have different drawing styles. The input sketches for the same shape are quite different from user to user (e.g., when drawing a multi-stroke sketch, some users like to draw it in one sequence while others like to draw it in another), and even from time to time. Therefore, Hidden Markov Models can be used to model different sketches and they can easily represent the user's drawing styles.

The Hidden Markov Model topology used in pattern recognition can be divided into two categories: the chain topology and the network topology. HMM chain is a simple structure. It is easy to implement and is widely used in recognizing simple symbols, e.g. gesture recognition. An HMM network is constructed by grouping and interconnecting HMM chains and is largely used in recognizing handwritten characters [14]. To date, there has been no serious study or guidance in the use of HMM in sketch recognition, and it is the first time that we have used HMM in multi-stroke sketch recognition. In this paper, we have selected the simple HMM chain topology, as shown in Fig. 2, because it has been shown to be successful in speech and handwriting recognition.

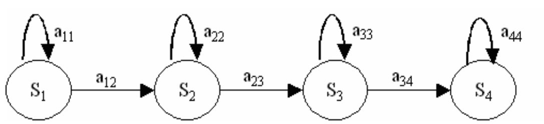


Fig. 2. Adaptive Hidden Markov Model Topology

According to the characteristics of the drawing sketch, the position and structure of the current stroke is usually dependent on the previous stroke, and the position and structure of the next stroke is dependent on the current stroke. We assume that the stroke of one sketch drawn by a user is only correlated with the previous stroke and the next stroke. Therefore, in our approach, we use a first-order left-to-right HMM chain, as shown in Fig. 2, to model each sketch. It is strictly causal: the current state depends only upon previous states. The experiments in handwriting recognition showed that this topology leads to high recognition accuracy.

In the training stage, we extract all composite features from each of the strokes and use the method mentioned in the previous section to determine the HMM state-number and some other parameters. Models are trained using the well-known iterative segmental training method based on Viterbi decoding. The transition probabilities indicate the relationships between strokes. The HMM chain can represent the user's drawing habit very well.

In the recognition stage, the recognizer calculates the probabilities using the trained HMM and returns the recognition results in the sequence of probabilities from high to low to the user.

2.3 HMM State-Number Determination for Adaptive Sketch Recognition

HMM needs enough free parameters to accommodate complexity of target patterns and to represent properties of the patterns. However, in practice, available training samples are usually limited, so it is often difficult to obtain enough free parameters. In our approach, we focus on one design parameter: the number of states in the HMM.

The number of HMM states is an important design parameter. For instance, a state could correspond to a certain phonetic event in a sketch recognition system. Thus, in modeling complex patterns, the number of states should be increased accordingly. When there are insufficient numbers of states, the discrimination power of the HMM is reduced, since more than one signal should be modeled on one state. On the other hand, the excessive number of states can generate the over-fitting problem when the number of training samples is insufficient compared to that of the model parameters.

There are two approaches to determining the HMM state-number used in handwriting recognition. The first is using a fixed state-number, which means using the same HMM state-number while training each category of samples. The second is using a variable state-number, which means the handwritten characters are divided into sub-components according to some given criterion (usually they are divided by strokes). Each subcomponent is modeled by one single HMM state.

Neither of the two methods mentioned above is fit for online multi-stroke sketch recognition because sketch has its own characteristics compared with handwritten character. First, the spatial relationships between strokes of a given sketch are more complex than that of the handwritten character. If we use a fixed state-number, we need to segment the sketch into subcomponents. The spatial relationships between strokes, which contain important sketching style information, will be broken, and the recognizer cannot capture enough information to represent the user's sketching habits. Obviously, the recognition accuracy will have high sensitivity to the segmentation process. Second, a number of standard character databases are present. In addition, in the handwritten characters are some fixed, predefined and well-known graphic objects among writers and readers, which have strict definition for strokes and stroke-sequence, so we can analyze all characters in the standard character databases and obtain the number of subcomponents, which are often used in different characters. In sketch recognition, there is no such standard database, so we cannot analyze all of the sketches and enumerate all of the constitutive subcomponents of sketches, and we cannot determine the state's number according to the number of subcomponents.

As mentioned above, we must find a new approach to determine the number of HMM states in multi-stroke sketch recognition. Although the sketches drawn by different users are very different from each other, they are all drawn stroke by stroke, which are then joined one by one. The stroke-number of one given sketch is different from every other among different sketching styles. Even if the numbers of strokes are the same, the structure of each stroke will be different from every other. Stroke is a natural representative of a user's sketching styles. The recognition performance will be upgraded if we make better use of the information contained in these strokes. In this paper, we proposed an adaptive HMM based on a variable state-number number for the purpose of generating a description of a multi-stroke sketch. In this approach, the number of HMM states is determined by the structural decomposition of the target pattern. Sketch is structurally simplified as a sequence of strokes.

The main idea behind the proposed approach is to use a single HMM state to model each stroke. While collecting samples, the recognition system will automatically store the stroke-number of each sample (which is defined to be *Snumber*). Before we train the HMM, we analyze the stored numbers and find out the maximum emergent number (which is defined to be *Tnumber*) for each category of sketches. We consider *TNumber* to be the state-number of HMM, because the samples corresponding to *TNumber* are frequently drawn by the user and they can represent the user's drawing habit. Then we train the HMM as follows:

i). If *Tnumber* > *SNumber*, we segment the last stroke of the sketch into *TNumber-SNumber+1* segments on average, and then model the remaining strokes and these segments to *TNumber* HMM states.

ii). If *Tnumber* < *SNumber*, we group the last *SNumber-TNumber+1* strokes to one virtual stroke, and then model the remaining strokes and the virtual stroke to *TNumber* HMM states.

Using this proposed approach to determine the HMM state-number, the inner structure of the HMMs is easily altered according to different users. Moreover, the approach does not need too much intervention by the user. After the user has become familiar with the input environment and the structure of sketches they usually draw, the user will draw one given sketch almost in the same style, and the stroke-number will become equal to *TNumber*. The recognizer will then seldom segment the sketch drawn by the user. Compared with other approaches, our approach is fit for online multi-stroke sketch recognition.

3 Experiment Results and Evaluation

The purpose of our experiments is to evaluate the effectiveness and efficiency of the recognition approach we have proposed above. In order to evaluate our proposed state-number determination approach, we carry out experiments for comparing the performance of variable state-number with that of fixed state-number. We also perform experiments to evaluate the performance of an adaptive online sketch recognition method based on our designed HMM. Our experiment environment is Pentium 4 1.6G CPU, 256MB memory, Windows 2000, Visual C++ 6.0.

By analyzing users' input strokes and some familiar graphics-based design software, we have set 9 classes of sketch, including straight line, arc, ellipse**,** poly-line 1, poly-line 2, poly-line 3, triangle, quadrangle and pentagon, as shown in Fig. 3. These are the most commonly used classes of sketch in the sketching process.

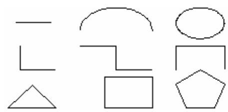


Fig. 3. All Nine Classes of Sketches

For the data collection, we collected two users' samples of these sketches. When using some present sketch recognition systems, the users are forced to draw only one stroke for simple shapes. However, a successful system should not restrict the user's drawing styles. For comparison, we asked the first user to draw these sketches in one-stroke and the second user to draw these sketches freely. The numbers of each class of sample are listed in Table 2.

Table 2. Number of samples collected from User 1 and User 2

Types	Straight Line	Arc	Poly-line 1	Poly-line 2	Poly-line 3
User 1	801	800	800	800	800
User 2	811	820	798	851	802
Types	**Triangle**	**Quadrangle**	**Pentagon**	**Ellipse**	**Total**
User 1	800	800	800	801	7,203
User 2	848	846	817	815	7,408

3.1 Comparison Between the Fixed and Variable State-Number HMM

In the previous section, we considered that variable state-number HMM is better than fixed state-number HMM in sketch recognition. The experiment in this section will confirm the conclusion. Because sketches, which have a greater number of strokes can easily lead to a multi-drawing-style, we choose polygon samples drawn by User 2 for the experiment. From the samples drawn by the second user, we obtain their drawing habits in the form of stroke-sequences, as shown in Fig. 4.

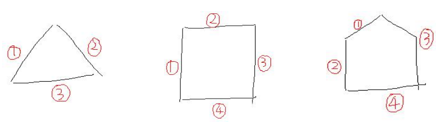


Fig. 4. Polygon Drawn by User 2

In the fixed state-number experiment, given the state-number to be 1, 2, 3 and 4, we obtain four different recognition precisions. In the variable state-number experiment, we use the state-number determining approach mentioned above to determine the state-number (in this experiment, the numbers of states are 2, 3 and 4) for each class.

Fig. 5. shows the result of our experiments. As we can see, the red pentagram corresponds to the precision of triangle recognition (using variable state-number), while the green one is for quadrangle and the blue one is for pentagon. The other points represent results for the fixed state-number experiment.

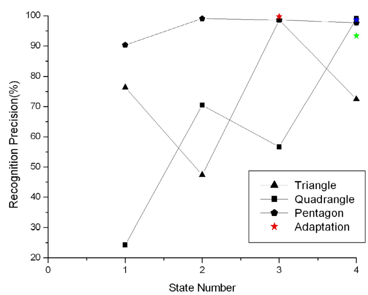


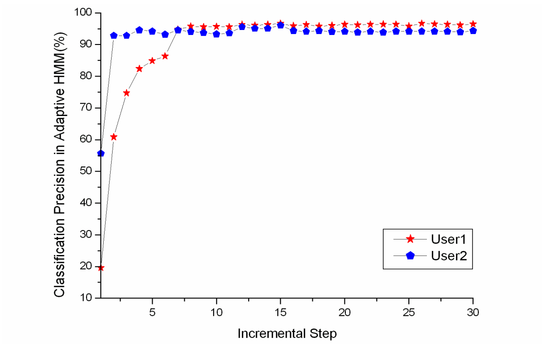
Fig. 5. Comparison of precisions of the fixed and variable state-number HMM

From Fig. 5., we can see that the recognition precision of different classes depends on state-number when a fixed state-number approach has been used. For one given fixed state-number, the recognition precisions of different classes of sketch are fluctuant and lower at the same time. Comparatively, when using a variable state-number approach, the recognition precisions of each class are nearly stable. Accordingly, we make a conclusion that a variable state-number approach can adapt different drawing styles, and it is better than a fixed state-number approach in on-line multi-stroke sketch recognition.

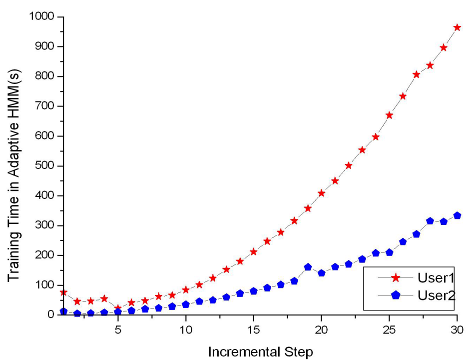
3.2 Performance of Adaptive HMM-Based Sketch Recognition

In this experiment, 60% of the first user's samples are used for training, while 75% of the second user's are used for training, and the remaining samples are used for testing. We divide the samples into 30 training sets for each user. The first five training sets contain 1%, 1%, 2%, 3% and 3% of the total samples of each user. Each of the remaining 25 training sets contains 3.6% of the total samples of each user. The experiment results are shown in Fig. 6.

From Fig. 6., we can see that our proposed method obtains a high performance; in multi-stroke sketching (for sketches drawn by the second user), the recognition precision reaches 95% after the second training, while in one-stroke sketching (sketches drawn by the first user), the recognition precision reaches 95% after the seventh training. The results show that our proposed adaptive online sketch recognition method based on HMM has a good performance in adaptation to user's drawing habits, especially under multi-stroke sketch and small training sets.



(a) Recognition precision of adaptive Hidden Markov Model



(b) Training time of adaptive Hidden Markov Model

Fig. 6. Experiment results of our proposed method

4 Conclusion

In this paper, we develop an HMM-based method for multi-stroke sketch recognition, where the drawing sketch is viewed as the result of a stochastic process that is governed by a hidden stochastic model and identified according to its probability of generating the output. To capture a users' drawing habit, a composite feature representation of each stroke is defined. To implement the stochastic process of online multi-stroke sketch recognition, multi-stroke sketching is modeled as an HMM chain while the strokes are mapped as different HMM states. To cater for the requirement of adaptive online sketch recognition, a variable state-number determining method for adaptive HMM is proposed. The experiments prove both the effectiveness and efficiency of the proposed method.

Acknowledgement

The work described in this paper is supported by the grants from the National Natural Science Foundation of China [Project No. 69903006 and 60373065] and the Program for New Century Excellent Talents in University of China (2004).

References

1. Zhengxing Sun and Jing Liu: Informal user interfaces for graphical computing, Lecture Notes in Computer Science, Vol. 3784, 2005, page 675-682.
2. Landay J. A. and Myers B. A.: Sketching Interfaces: toward more human interface design. IEEE Computer, Vol. 34, No. 3, 2001, page 56-64.
3. Rubine D, Specifying gestures by example, Computer Graphics, Vol. 25, 1991, page 329-337.
4. Newman M. W., James L., Hong J. I., et al: DENIM: An informal web site design tool inspired by observations of practice, HCI, Vol. 18, 2003, page 259-324.
5. Fonseca M. J., Pimentel C. and Jorge J. A.: CALI - an online scribble recognizer for calligraphic interfaces, In: AAAI Spring Symposium on Sketch Understanding, AAAI Press, 2002, page 51-58.
6. Chris Calhoun, Thomas F. Stahovich, Tolga Kurtoglu, et al: Recognizing multi-stroke symbols, AAAI Spring Symposium on Sketch Understanding, AAAI Press, 2002, page 15-23.
7. Xiaogang Xu, Zhengxing Sun, Binbin Peng, et al: An online composite graphics recognition approach based on matching of spatial relation graphs, International Journal of Document Analysis and Recognition, Vol. 7, No.1, 2004, page 44-55.
8. Zhengxing Sun, Wenyin Liu, Binbin Peng, et al: User adaptation for online sketchy shape recognition, Lecture Notes in Computer Science, Vol. 3088, 2004, page 303-314.
9. Zhengxing Sun, Lisha Zhang and Enyi Tang: An incremental learning algorithm based on SVM for online sketchy shape recognition, Lecture Notes in Computer Science, Vol. 3610, 2005, page 655-659.
10. Rabiner L. R.: A Tutorial on Hidden Markov Models and selected applications in speech recognition. Proceedings of the IEEE, Vol.77, No.2, 1989, page 257-286.
11. Jen-Tzung Chien: On-line unsupervised learning of hidden Markov models for adaptive speech recognition, Proceedings of Vision, Image and Signal Processing, Vol. 148, No. 5, 2001, page 315-324.
12. Hu Jianying: Michael K Brown and William Turin, HMM-based online handwriting recognition, IEEE Transactions on PAMI, Vol. 18, No. 10, 1996, page 1039-1045.
13. Mitsuru Nakai, Naoto Akira, Hiroshi Shimodaira, et al: Sub-stroke approach to HMM-based On-line Kanji Handwriting Recognition, International Conference on Document Analysis and Recognition, 2001, page 491-495.
14. Jay J.Lee, Jah Wan Kim and Jin H. Kim: Data-driven design of HMM Topology for on-line handwriting recognition. In: Hidden Markov models: applications in computer vision, World Scientific Series, 2001, page 107-121.