A Novel Pen-Based Flowchart Recognition System for Programming Teaching*

Zhenming Yuan, Hong Pan, and Liang Zhang

School of Information Science and Engineering, Hangzhou Normal University,
16 Xuelin Street Xiasha District, Hangzhou, P.R. China
zmyuan@cs.zju.edu.cn, chin@zju.edu.cn, zhl@hznu.edu.cn

Abstract. The electronic white board and the tablet PC are bringing new technologies to modern education. This paper presents a pen-based flowchart recognition system for programming teaching, which uses hybrid SVM-HMM algorithm for sketch recognition. In this algorithm, ICA is used to reduce the dimension of features, a set of fuzzy SVMs are used as preliminary feature classifiers to produce fix length feature vector, which acts as a probability evaluator in the hidden states of Hidden Markov Models, and HMMs are employed as finally classifiers to recognize the unknown pattern. Experiment results show the hybrid algorithm has good learning and recognition ability. And based on this algorithm, an intelligent whiteboard system for programming teaching is designed to identify the sketches into the programming flowchart, and finally converts it into C language programs. User's evaluation shows it is natural for the teachers and the students with a flexible and effective interactive teaching pattern. Therefore, such system brings a new programming teaching patterns and help students to stride the obstacle between the flowchart and the programming language. Students can learn the abstract programming idea and the concrete coding skills effectively and efficiently by the visual comparative learning assisted by the intelligent whiteboard system.

Keywords: teaching system, sketch recognition, SVM, HMM, ICA.

1 Introduction

The electronic white board and the Tablet PC are bringing new technologies to modern education. Sketching on such equipments provides an intuitive user interface. It issues the challenges to allow the teachers sketching information on electronic white board with more intelligent manners which supports the recognition of sketches, while the traditional manner supported by the commercial electronic white boards only have drawing functions.

Recent researches are engaged in the development of Table PC based systems that interpret various visual languages [1], including mathematical expressions [2], chemical diagrams [3], digital circuits, mechanical systems [4], and UML class diagram [5].

* Project supported by the Natural Science Foundation of China (No. 60773051), and the Natural Science Foundation of Zhejiang Province (No. Y107631), China.

E.W.C. Leung et al. (Eds.): WBL 2008, LNCS 5328, pp. 55 – 64, 2008.

Such systems focus on the creation, manipulation and recognition of diagrammatic languages, which help instructors convert the physical diagrams into the subject concepts and furthermore facilitate the concept modeling for students. In these systems, the recognition of primal symbols, such as rectangle, circle, arc, etc., is the essential processing. However current recognition algorithms are almost based on geometric rule inference, which are sensitive to the noise of the sketches. The methods of sketch understanding can be classified into two categories. The first category recognizes the sketch without timing information. Such algorithms regard the sketch as image or graphics, and classify the sketches using rule-based or classify-based algorithms (such as SVM, etc.) [6][7]. The other category of recognition algorithms thinks it is important to take advantage of timing, and usually use Hidden Markov Models (HMM) and Dynamic Bayesian Networks to learn temporal patterns [8][9].

This paper proposes a hybrid SVM-HMM sketch recognition algorithm combining the two classifiers into an ideal one, which supports a single stroke recognizer for the flow chart tutoring system. HMM is good at dealing with sequential data, while SVM shows superior performance in classification. But HMM has some shortcoming such as the poor ability of classification, poor ability of pattern recognition, the high dependence on the statistical knowledge of the pre-experimentation, etc. SVM is a powerful supervised learning theory, coming from the theory of statistic learning [10]. By minimizing the sum of the empirical risk and the complexity of hypothesis space, SVM gives good ability of generalization for pattern recognition problems [11]. Furthermore, the former approach usually provides an intra-class measurement while the latter proposes inter-class difference. The goal of the recognition algorithm in this paper is to make use of the advantages of SVM and HMM to overcome some of their shortages.

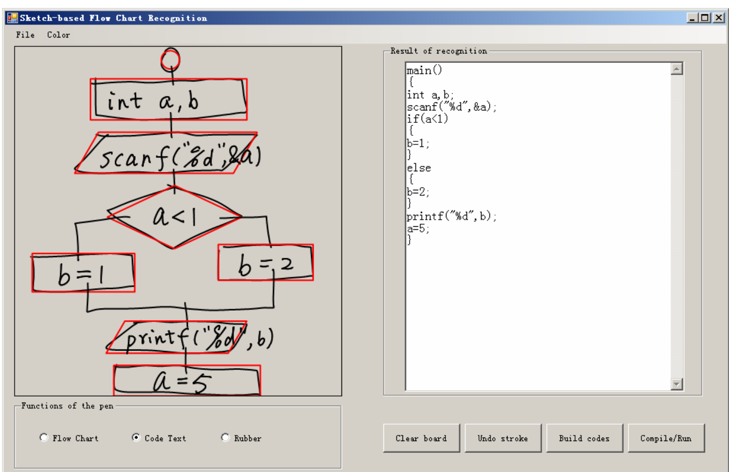


Fig. 1. This shows the pen-based programming teaching system. Left panel is the sketching area on which teacher draws the symbols of a flowchart and some program instructions. The red symbols are the refined results of recognition. Right panel is the C programming code transformed from the left flowchart. Teacher may use the right-bottom buttons to run the codes.

This paper also introduces a programming flowchart teaching system based on pen interface (Fig.1), which can recognize teachers' free-hand sketches as flow chart diagrams and transform them into C programs. Firstly, the system is built on pen-based interface, which can support the teachers to present the flowcharts and C programs simultaneously on the electronic whiteboard by sketching on the Pen-based HCI. Secondly, the system can recognize the free-hand sketches using the graphics recognition algorithms in this paper, and translate the sketching flowcharts into corresponding C programs. By transforming the graphics into the symbols of the domain-specific knowledge automatically and comparing the correspondence between them, it is helpful to build connection between the abstract concept and the concrete graphics during the learning activities. By user evaluation in the course of programming, it seems that the practical system is very useful for students to learn the structural programming ideation.

The remainder of the paper is organized as follows. Section 2 describes the feature extraction and reduction of the flowchart pen-based sketch. Section 3 and Section 4 details the hybrid SVM-HMM recognition algorithm, which is the kernel of the pen-based teaching system. Section 5 provides experimental results and discussions. Section 6 presents conclusions.

2 Feature Extraction

2.1 Stroke Features

Feature extraction is a process of transforming free-hand sketching to a feature vector which represents the geometric and the action characteristics of drawings. The algorithm includes three steps: sampling, pre-processing, and feature computing. After employing sampling firstly, each stroke is represented by a finite set of ordered pairs of $<p_i, t_i>$, where p_i is the position (x_i,y_i) at the time of t_i. The feature vector of a stroke consists of the feature vectors of all sample points, each of which is represented by the local features:{direction (d_i), curvature(c_i),length ratio from start point(l_i), speed(s_i), distance from center of gravity(dc_i), angle from center of gravity(a_i)}.

$$d_i = \arctan(\frac{y_{i+\delta} - y_{i-\delta}}{x_{i+\delta} - x_{i-\delta}}) - \arctan(\frac{y_{i+\delta} - y_i}{x_{i+\delta} - x_i}) \tag{1}$$

$$c_i = \left| \sum_{j=i-\delta}^{j=i+\delta-1} \varphi(d_{j+1} - d_j) \right| / length(p_{i-\delta}, p_{i+\delta}) \tag{2}$$

$$l_i = length(p_i, p_{start}) / length(p_{end}, p_{start}) \tag{3}$$

$$s_i = length(p_{i+\delta}, p_{i-\delta}) / (t_{i+\delta}, t_{i-\delta}) \tag{4}$$

$$dc_i = \sqrt{(y_i - center_y)^2 + (x_i - center_x)^2} / \max(width, height) \tag{5}$$

$$a_i = \arctan((y_i - center_y)/(x_i - center_x)) \tag{6}$$

where ($center_x$, $center_y$) is the position of center of gravity of the stroke, (*width*, *height*) is the minimum box of the stroke, *length*() is the length along the stroke.

2.2 Feature Reduction

The feature of a stroke is an $m \times n$ matrix $\boldsymbol{x}$, which m is the number of features of one sample point, and n is the number of sample points. In the high-order statistic, Independent Component Analysis (ICA) not only can distinguish the signal of mixing but also can draw the lower characteristic in the signal effectively. Here we replace the primitive features with the ICA features and $\boldsymbol{x}$ can be regarded as the result of a linear mixture model in ICA:

$$x = As \tag{7}$$

where components of $\boldsymbol{s}$ are independent sources and unknown, $\boldsymbol{A}$ is unknown, and $\boldsymbol{x}$ is the observation. ICA tries to estimate the matrix $\boldsymbol{W}$ in the reconstruction model:

$$y = Wx \tag{8}$$

Regarding matrix $\boldsymbol{x}$ as the observation value of random vector, it can be taken by the maximum m eigenvalues of $\boldsymbol{x}$ as a matrix P_m. Then $\boldsymbol{x}$ can be replaced with P_m^T, and apply the ICA algorithm as followings:

$$WP_m^T = y \Rightarrow P_m^T = W^{-1}y \tag{9}$$

where each row of $\boldsymbol{y}$ represents an independent base vector. Finally a stroke can be represented by a row feature vector $\boldsymbol{F}$, which is consisted of the coefficients projected onto the linear combination of independent base vector in $\boldsymbol{y}$.

3 SVM with Probability Output

3.1 Binary SVM with Probability Output

In traditional SVM with the training set $\{c_i, d_i\}_{i=1}^{l}$, the classification results are $d_i = \{1, -1\}$, where l is the size of training set. So it can only be applied to binary classification problems. For the convenience of composing the hybrid model with HMM, a fuzzy SVM is introduced to change the output form into the probability value. In general, a SVM has following form:

$$f(x) = sign(\sum_{i=1}^{l} y_i \lambda_i^* K(x, x_i) + b^*) \tag{10}$$

where the parameters are obtained by maximize the objective function:

$$Q(\Lambda)=\sum_{i=1}^{l}\lambda_i-\frac{1}{2}\sum_{i=1}^{l}\ \sum_{j=1}^{l}\lambda_i\lambda_j y_i y_j K(x_i,x_j) \tag{11}$$

The restriction condition is:

$$\sum_{i=1}^{l}\lambda_i y_i=0 \qquad 0\le\lambda_i\le C \tag{12}$$

By resolving the quadratic programming problem, SVM try to maximize the margin of separation, at the same time minimize the training error. According to Vapnik's statistic learning theory, the risk boundary satisfies $\forall\alpha\in\Lambda$,

$$R(\alpha)\le R_{emp}(\alpha)+\sqrt{\frac{h(\log(2l/h)+l)-\log(\eta/4)}{l}} \tag{13}$$

holding probability $1-\eta$, where $0\le\eta\le1$, Λ is the parameter space, *h* is the dimension of $f(\cdot,\alpha)$. For SVM and giving input vector, larger separation between the positive samples and negative samples, less the upper boundary of VC dimension h, which means to reduce the complex in hypothesis space.

To carry out the fuzzy SVM, a Sigmoid function maps the SVM of each independent topological relation to posterior probability [7]:

$$P(S_i\mid X)=\frac{1}{1+\exp(A_i X_{out}+B_i)} \tag{14}$$

where A_i and B_i is the Sigmoid parameters of each SVM, X_{out} is the distance between any query *X* and SVM.

3.2 Multi-class SVM

The probability of (14) is still a binary problem. To deal with multi-class problem, we use one to one combinations of binary SVM. We construct binary SVMs for one class C_i to each other class $\{C_j|1\le j\le M,\ i\ne j\}$. The output of each SVM is calculated by equation (14), which can be combined into a feature vector for class C_i:

$$V(x)=[P_{i1}(C_i\mid x),P_{i2}(C_i\mid x),\ldots,P_{ij}(C_i\mid x),\ldots,P_{iM-1}(C_i\mid x)]^T \tag{15}$$

where, $P_{ij}(C_i|x)$ is the output of SVM distinguishing class C_i and other class C_j, $i\ne j$, *M* is the number of classes in the problem. Then this feature vector is transformed to a probability output of class C_i by a Gaussian Model:

$$\begin{aligned}P(C_i\mid x)&=N(V(x),\mu,\Sigma)\\&=((2\pi)^{-d/2}\mid\Sigma\mid^{-1/2})\exp[-\frac{1}{2}(V(x)-\mu)^T\,\Sigma^{-1}(V(x)-\mu)]\end{aligned} \tag{16}$$

Finally, we can obtain a feature vector containing every probability output of classes:

$$O = \{P(C_1 | x), P(C_2 | x), ..., P(C_M | x)\} \tag{17}$$

4 SVM-HMM Hybrid Approach

4.1 The Hybrid Model

This paper proposes a hybrid model combining SVM and HMM. The model is composed of three steps. First, the raw sketch data is sampled and the features of stroke are calculated. To solve the problem of variable length of one stroke and to eliminate the redundancy of feature information, ICA is used to transform the feature matrix into fix length feature vector. Second, a set of SVMs are used as preliminary feature classifiers. The outputs of SVMs are converted into the form of posterior probability, which acts as a probability evaluator in the hidden states of hidden Markov models. Third, a HMM are employed as finally classifiers to recognize the unknown pattern. Consider *M* classes $l = \{l_1, l_2, ..., l_M\}$ with their respective *M* groups of SVMs and a group of HMM $\lambda = \{\lambda^1, \lambda^2, ..., \lambda^M\}$, where each group of SVM is corresponding to the feature vector of class *i* with *M*-1 binary SVMs, and each HMM is a left-to-right HMM with the same length *M*.

An HMM $\lambda(A, B, \pi)$ is a stochastic process for producing a sequence of observed values. It is specified by three parameters A, B, π . ***A*** is the transition probability matrix $a_{ij} = P(\omega_{t+1} = j | \omega_t = i)$, ***B*** is the observation probability distribution $b_{jk} = P(v_t = k | \omega_t = j)$, and π is the initial state distribution. $\omega^T = \{\omega_1, \omega_2, ..., \omega_N\}$ is the set of HMM states and $v^T = \{v_1, v_2, ..., v_M\}$ is the set of observations. In hybrid model, the probability outputs of SVM act as the posterior probability of observation of HMM. After training, the model parameters of HMM $\lambda(A, B, \pi)$ are obtained, where B is the Gaussian parameters of the probability outputs.

4.2 Training Processing

The process of training is: 1) estimate the parameters of SVMs, 2) produce the new feature vectors formed by probability outputs of the trained SVMs, and 3) estimate the parameters of HMM. Given a set of samples, the training algorithm of multi-class SVM is composed of a number of binary SVMs, the parameters of which can be obtained by solving a convex quadratic programming problem subject to linear constraints. One classifier C^{ij} for every pair of distinct classes <*i*,*j*>. Each classifier is trained with the samples in the i^{th} class with positive labels, and the samples in the j^{th} class with negative labels. Given a set of observation sequences produced by SVMs $v = \{v^1, v^2, ..., v^k\}$, Baum-Welch algorithm [8] is used to estimate the parameters of every HMM. For each HMM, the outputs of the samples of one symbol are averaged, and the proportion of the number belonging to one symbol to the number of total samples determines the initialize probability distribution π .

4.3 Recognition Processing

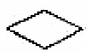
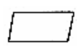
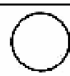
The recognition processing is performed along the three steps as described above. Suppose a free-hand sketch symbol is feed into the system, a reductive row feature vector $\boldsymbol{F}$ is feed into a set of SVM groups, and produce a new feature vector v defined by probability outputs as the observation sequence of HMMs $\lambda = \{\lambda^1, \lambda^2, ..., \lambda^M\}$. Finally, the unknown symbol v is assigned to the class having the highest probability estimation $\log P(v \mid \lambda^i)$.

5 Experiments

5.1 Recognition Algorithm Experiments

In our experimental, we construct a practical sketch-based C programming teaching system, which can recognize 6 distinct flow chart symbols: {operation, decision, I/O, connector, termination, line} as Table 1.

Table 1. The symbols of the flow chart

Name	Operation	Decision	I/O	connector	termination	line
Symbols						

The experimental database consists of 1200 flow chart symbols drawn by 10 peoples, in which each kind of symbol is drawn 20 times by one person. We ran separate tests for 5 training data groups, which are respectively composed of 10,20,30,40,50 samples from each symbol class. And the remaining samples are used for the test sets in each group. The recognition rate is illustrated in Fig.2. We observed that the recognition rate is low by 10 training samples. With the number of training samples, the recognition rate increase quickly, and keep high when more than 30 training samples. The hybrid algorithm shows good recognition ability under small samples and represents strong learning ability. Furthermore, the hybrid algorithm is compared with the recognition algorithm based on pure traditional HMMs, and also shows better performance as listed in Table 2.

5.2 User Evaluation for Programming Teaching

An experimental intelligent whiteboard system is designed using the hybrid SVM-HMM recognition algorithm for a C programming course. During the user evaluation for programming teaching, the teachers and students are classified into two test groups. The first group uses our intelligent whiteboard system to teach the loop structure in the C programming course. In this group, the students were trained by the

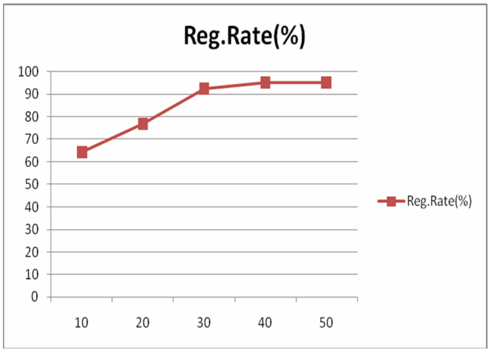


Fig. 2. The recognition rate of different sample groups shows that it increases with the number of training samples and keeps high over 30 training samples

Table 2. Comparative results of SVM-HMM with HMM approach show the hybrid method has better performance than traditional HMM approach

Symbol	SVM-HMM hybrid approach	Traditional HMM approach
operation	97.21	96.40
decision	96.30	95.27
I/O	96.31	94.89
connector	92.15	90.31
termination	93.28	91.44

visual comparative learning, in which the teacher drew the loop structure flowchart on left of the electronic whiteboard, the corresponding C codes were generated automatically, and the students learned the concept and coding of the loop structure through comparing the simultaneously changes between the flowcharts and the codes. The second group uses traditional teaching method to learn the same contents by the blackboard and the PowerPoint slides.

After the semester, a questionnaire survey was conducted to the testers about the teaching and learning effects. Teachers found that the intelligent whiteboard affords a more efficient teaching method than that of the traditional ones. In addition, students are fond of using the intelligent whiteboard to accomplish their learning tasks.

From the analysis of the user evaluation, it is indicates that the natural interaction, the efficiency and the satisfaction have received great reputation. Graphics are usually used to present the flowchart notions clearly, such as the object of the flowchart and the relations between objects. Free-hand sketch is one of the natural and effective approaches to the expression of intentions, which helps the teachers to show the programming concepts as naturally as how they do on paper. The combination of pen-based HCI and whiteboard enable students to blend learning efficiently. By observing the corresponding relationships between the sketching flowchart and the codes, the students can obtain the immediate feedback, which makes the whiteboard natural for the teachers and the students with a flexible and effective interactive pattern. Therefore, students can learn the abstract programming idea and the concrete coding skills effectively and efficiently by associating the graphics with domain concepts.

6 Conclusions

A pen-based flowchart system for programming teaching is presented in this work, which makes use of the advantages of SVM and HMM to overcome some of their shortages. In this work, ICA is used to reduce the dimension of features, a set of fuzzy SVMs are used as preliminary feature classifiers to produce fix length feature vector, which acts as a probability evaluator in the hidden states of Hidden Markov Models, and HMMs are employed as finally classifiers to recognize the unknown pattern. Experiments show the algorithm is good enough for flowchart sketch recognition.

This work also introduces the pen computing into programming teaching, which can recognize teachers' free-hand sketches as flow chart diagrams and translate them into C programs automatically. The system provides the users with a visual teaching environment that facilitates the construction of top-down design flow charts and the implementation and simulation of algorithms as flow charts. It helps students to stride the obstacle between the flowchart and the programming language, and to learn the structural programming ideation effectively.

References

1. Sezgin, T.M.: Overview of Recent Work in Pen-Centric Computing: Vision and Research Summary. In: Workshop on Pen-Centric Computing Research. Brown University (2007)
2. LaViola, J., Zeleznik, R.: MathPad2: A System for the Creation and Exploration of Mathematical Sketches. ACM Trans. Graphics (Proc. Siggraph 2004) 23(3), 432–440 (2004)
3. Ouyang, T.Y., Davis, R.: Recognition of Hand Drawn Chemical Diagrams. In: Proceedings of AAAI, pp. 846–851 (2007)
4. Stahovich, T.: Sketchit: a sketch interpretation tool for conceptual mechanism design. Technical report, MIT AI Laboratory (1996)
5. Qiu, L.: SketchUML: The Design of a Sketch-based Tool for UML Class Diagrams. In: Proceedings of World Conference on Educational Multimedia, Hypermedia & Telecommunications (ED-MEDIA) (2007)
6. Landay, J.A., Myers, B.A.: Sketching Interfaces:Toward More Human Interface Design. Computer (2001)
7. Liang, S., Sun, Z., Li, B.: Sketch retrieval based on spatial relations. Computer Graphics, Imaging and Vision: New Trends (2005)
8. Sezgin, T.M., Davis, R.: HMM Based Efficient Sketch Recognition. In: Proceedings of the International Conference on Intelligent User Interfaces (IUI 2005), San Diego, CA (2005)
9. Stahovich, T., Landay, J., Davis, R.: AAAI Sketch Understanding Symposium, Stanford, CA, March 25-27 (2002)
10. Vapnik, V.: The Nature of Statistical Learning Theory. Springer, New York (1995)
11. Burges, C.J.C.: A tutorial on support vector machines for pattern recognition. Knowledge Discovery and Data Mining 2(2) (1998)
12. Platt, J.C.: Probabilistic outputs for support vector machines for pattern recognition. In: Smola, A., Barlett, P., Scholkopf, B. (eds.) Advances in Large margin Classifiers. Kluwer Academic Publishers, Boston (1999)

13. Rabiner, L.R.: A tutorial on Hidden Markov Models and selected applications in speech recognition. Proceedings of the IEEE 77(2) (1989)
14. Costagliola, G., Deufemia, V., Polese, G., Risi, M.: A Parsing Technique for Sketch Recognition Systems. In: Proceedings of the 2004 IEEE Symposium on Visual Languages and Human Centric Computing (VLHCC 2004) (2004)
15. Huang, B.Q., Du, C.J., Zhang, Y.B., Kechadi, M.: A Hybrid HMM-SVM Method for Online Handwriting Symbol Recognition. In: Proceedings of the Sixth international Conference on intelligent Systems Design and Applications (Isda 2006), vol. 01 (2006)