

A Structural Method for Online Sketched Symbol Recognition

Wei Deng, Lingda Wu, Chao Yang, and Zhongwen Zhao
Key Lab. of the Academy of Equipment, Beijing, China
Email: dengwei@whu.edu.cn

Abstract — The increasing availability of pen-based hardware has resulted in a parallel growth in research in sketch recognition. However, many challenges remain in terms of recognition accuracy, robustness to different drawing styles, rotational invariance, and the number training samples. To address these challenges, a new structural approach to online sketched symbol recognition was proposed, which focuses on the primitive correspondence between a input symbol and the reference one. This method is independent of stroke-order,-number, as well as invariant to scaling and rotation. Experiments on two datasets show the effectiveness of the proposed approach.

Index Terms—hungarian algorithm, primitive correspondence, sketched symbols recognition.

I. INTRODUCTION

Sketching is a natural form of human communication and has become an increasingly popular tool for interacting with user interfaces. It is a fast and efficient means of capturing information in many different domains. With the growing popularity of digital input devices, there is increasing interest in building sketch-based user interfaces that can automatically interpret freehand drawings.

However, many challenges remain in terms of intra-class compactness and inter-class separation due to the variability of sketching. Because it is likely that different people have different drawing styles, such as the order and number of strokes, symbol size, and complex deformation. Moreover, the style may differ even the same individual at different times. A good recognition algorithm should place few drawing constraints on users. A similar research is handwriting recognition, such as digit and Chinese character recognition, which has many effective algorithms. But generally it needs a large number of training samples and has no invariance to rotation. In some cases it is difficult or inconvenient to gain enough training data.

This paper presents a structural method for online recognition of hand-sketched symbols, which is independent of stroke-order,-number, as well as invariant

to scaling and rotation, and just need a few training samples.

A. Related Work

One common approach to sketch recognition focuses on building structural shape descriptions[1]. Here the base vocabulary is typically composed of simple geometric primitives such as lines, arcs, and ellipses. Paulson and Hommond[2] proposed effective method to recognize 8 types of primitives. Hammond and Davis[3] developed a hierarchical language to describe how diagrams are drawn, display, and edited. They then used these descriptions to perform automatic symbol recognition.

Another alternative approach looks at the visual appearance of shapes and symbols[1]. Oltmans[4] proposed a visual parts-based model that uses a library of shapes contexts (oriented histograms of gradients) to describe and distinguish the different symbols in their domain. Ouyang and Davis[1] proposed a visual approach to sketched symbol recognition. It used a set of visual features that capture on-line stroke properties like orientation and endpoint location. Recently Almazan[5] used two modifications of Blurred Shape Model(BSM) descriptor as basic shape, combined with Active Appearance Model(AAM) to learn a model of shape variability in a set of patterns.

II. OUR APPROACH

For the purpose of convenient presentation, the input sketched symbol is recorded as U (meaning Unknown), and the reference one denoted by R (meaning Reference). $U = \{u_i\}$, $i=1,2,\dots,m$, $R = \{r_j\}$, $j=1,2,\dots,n$. u_i and r_j denote primitives, $u_i, r_j \in \{\text{Line, Arc, Ellipse}\}$. m and n are primitive number of R and U respectively.

Definition: Primitive matching matrix(PMM) the PMM between U and R is defined as a matrix of $m \times n$ (denoted as S). The matrix element $S(i, j) = d(u_i, r_j)$. Where, the function $d(\cdot)$ calculates the matching cost between two primitives, using the information of primitive type, length, and other geometric features.

After getting matching matrix S of between U and R , we want to minimize the total cost of matching

$$H(\pi) = \sum_i S(i, \pi(i)) \quad (1)$$

subject to the constraint that the matching be one-to-one, i.e., π is a permutation. This is an instance of the square assignment problem, which can be solved in $O(N^3)$ time using the Hungarian method. The input to the assignment problem is a square cost matrix with entries $S(i, j)$. The result is a permutation $\pi(i)$ such that (1) is minimized.

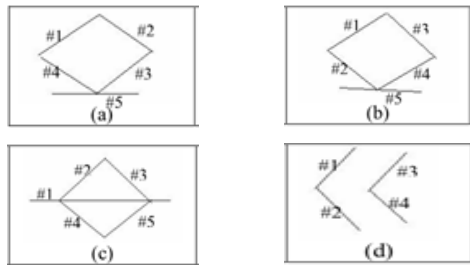


Figure 1. The illustration of primitive matching

Fig.1 illustrates the meaning of primitive matching. (a) is input symbols, and (b),(c),(d) are three references. Moreover, the digits express the primitive subscript. The purpose of matching is to get optimal primitive correspondence between (a) and (b),(c),(d). For (a) and (b), it is $(\#1,\#2,\#3,\#4,\#5) \rightarrow (\#1,\#3,\#4,\#2,\#5)$.

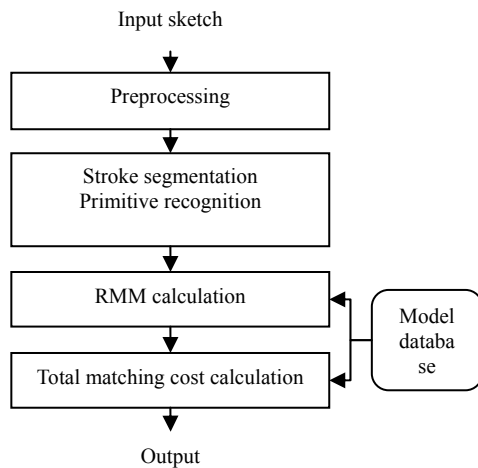


Figure 2. Overview of sketch recognition

A practical sketch recognition system is depicted diagrammatically in Fig. 2. The input to the system is a sequence of sketched trajectories. It involves the following steps: preprocessing, stroke segmentation & primitive recognition, RMM calculation, total matching cost calculation. The model database contains the reference models for classification.

A. Preprocessing

The preprocessing of the trajectory of input pattern directly facilitates pattern description and affects the quality of description. The preprocessing tasks of online sketch patterns include resampling and shape normalization. Resampling plays an important role on our method. Distance between adjacent points in the trajectory should evenly distributed. Normalization of shape trajectories to a standard size is also adopted in our method, as almost every character recognition system. We choose moment normalization technology. Its steps

are: the centroid of input pattern is shifted to the center of standard box and the second-order moments are scaled to a standard value ρ^2 .

B. Stroke Segmentation & Primitive Recognition

Many online structural recognition methods make the processing of stroke segmentation and primitive recognition as foundational work. Because primitives are views as the smallest operating unit. We use PaleoSketch algorithm in [2] for its good performance. But we only include three primitives—line, ellipse and arc. Its main idea is to extract some geometric features of stroke trajectory, which are then compared to predefined thresholds. We survey it simply here. Firstly, initial corners are detected. Then test a sub-stroke in the following steps. Line Test: Fitting a least squares line to the sub-stroke points and calculating the orthogonal distance squared. This is one feature. The other feature is feature area of the line. Now we add a third feature which is the ratio of trajectory length to chord length. Ellipse Test: Circle is treated as a particular ellipse. The algorithm first calculate the ideal major axis, center, and minor axis. The two features of NDDE(normalized distance between direction extremes) and DCR(direction change ratio) are effective. Another feature is also feature area. Arc Test: We first calculate the ideal center point of the arc using a series of perpendicular bisectors. Then we calculate the ideal radius of the arc by taking the average distance between the sub-stroke points and the center point. An arc must not be closed or overtraced and must have a high NDDE value and low DCR. In the three test steps, each feature has its own threshold.

C. RMM Calculation

The function $d(.,.)$ in the definition of PMM plays an key role. If U and R are in the same class, and u_i and r_j are corresponding primitive pair, intuitively the two primitives should be the same type and have similar length. That means if u_i is a line, r_j should be a line and has similar length with u_i . To obtain rotational invariant, we introduce a parameter θ to denote the rotational angle. Without loss of generality, let $m \geq n$. First, we find the primitive in U which has the longest distance between its center and the center of U (This primitive is denoted as u_d). We think this distance is longer, the direction is more stable. Because we do not know which primitive in $\{r_1, r_2, \dots, r_n\}$ does u_d correspond to, we build one PMM for the every possible primitive pair respectively. It has three steps. (1)We calculate θ for every pair. Let us denote them as $\theta_1, \theta_2, \dots, \theta_n$ for correspondence pair $u_d \leftrightarrow r_1, u_d \leftrightarrow r_2, \dots, u_d \leftrightarrow r_n$, respectively. (2)Then for every θ_k ($k=1,2,\dots,n$), we rotate U by θ_k to perform shape alignment. (3)PMMs for every θ_k is built, they are $\{PMM_k, k=1,2,\dots,n\}$. Where, matching cost of every possible correspondence between u_i and $r_j, d(u_i, r_j)$, are calculated by the following formulas. (Denoting the two endpoints of u_i as Pu_1, Pu_2 , and midpoint as Puc .

Analogously they are Pr1, Pr2 and Prc for r_j . Length() and C() represent the length and type of a primitive, respectively. Dui, U_{rj} represent the direction of u_i and r_j respectively. And $|\cdot|$ is Euclidean distance.)

$$d(u_i, r_j) = \begin{cases} \infty & (C(u_i) \neq C(r_j)) \\ d_L & (C(u_i) = C(r_j) = \text{Line or } C(u_i) = C(r_j) = \text{Arc}) \\ d_E & (C(u_i) = C(r_j) = \text{Ellipse}) \end{cases} \quad (2)$$

$$dL = \begin{cases} \infty & |D_{u_i} - D_{r_j}| > \beta \\ \min(dis1, dis2) \cdot \frac{\max(\text{length}(u_i), \text{length}(r_j))}{\min(\text{length}(u_i), \text{length}(r_j))} & \text{else} \end{cases} \quad (3)$$

where

$$dis1 = |P_{u1} - P_{r1}| + |P_{u2} - P_{r2}| + |P_{uc} - P_{rc}| \quad (4)$$

$$dis2 = |P_{u1} - P_{r2}| + |P_{u2} - P_{r1}| + |P_{uc} - P_{rc}| \quad (5)$$

$$d_E = 3 \cdot |P_{uc} - P_{rc}| \cdot \frac{\max(\text{length}(u_i), \text{length}(r_j))}{\min(\text{length}(u_i), \text{length}(r_j))} \quad (6)$$

After above three steps, we calculate total matching cost using the below method for every PMM in $\{PMM_k, k=1,2,\dots,n\}$. The matching cost between U and R is

$$Cost(U, R) = \max_{k=1,2,\dots,n} (E(U, R, PMM_k)) \quad (7)$$

where the function $E()$ is described below.

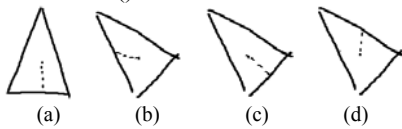


Figure 3. Performance of rotational invariance

Fig. 3 illustrates the process of rotational invariance. (a) is the input sketch. (b), (c) and (d) are the same reference. The dotted line specifies ud . When calculating the matching cost of the two symbols, we rotate (a) to align its dotted line to the dotted line in (b), (c) and (d) respectively. So we can get three PMMs. Thus the maximum of the three total matching cost is the true cost used to measure similarity.

D. Total Matching Cost Calculation

However, the primitive number of U and R may not be equal. We can add dummy primitive to a symbol with a constant matching cost ϵ . In our experiment, $\epsilon = 0$. So the primitives matching to the dummy are unmatched or superfluous actually. There are two different situations. On the one hand, U and R are in the same class. As the difference of drawing style, it is possible that some primitives in U may not appear in R , and vice versa. The algorithm should have robust handling of this case. On the other hand, U and R are in different classes. Because there is an observation that in the former, the unmatched primitive is always shorter than that in the latter, we add penalties of unmatched primitives, which have direct relation to the length, to the total matching cost between U

and R . So we can get the final matching cost between U and R .

After getting the PMMs between U and R , we can use the model of assignment problem to obtain the primitive correspondence determination with Hungarian algorithm. Firstly we add $(m-n)$ dummy primitives to R and then get the one-to-one primitive correspondence between U and R .

As discussed above, the primitive number of U and R may not be equal. In some domains an graphical symbol may be part of another symbol. One sample is illustrated in Fig. 4 (a) is input and (b),(c) are references. (c) can be part of (b). In this case if we only take the result of Hungarian algorithm into consideration, the matching cost of (a)-(b) will be bigger than (a)-(c). If we fix the primitive number of a symbol, this problem can be solved. But we can not do this. Because there tends to be a great deal of noise at the beginning and ending of a stroke (like tails). They are superfluous primitives. To get appropriate compromise of accuracy and robustness, we punish the unmatched primitives(denoted as $\{o_t\}, t=1,2,\dots, m-n$), each of which is paired with a dummy primitive, by the following formulae.

$$Cost(U, R) = \max_{k=1,2,\dots,n} (E(U, R, PMM_k)) = \max_k (D(U, R, PMM_k)) + \alpha \cdot \sum_{t=1}^{m-n} \text{length}(o_t) \quad (8)$$

where $E(U, R, PMM)$ is the final matching cost between U and R with the PMM, $D(U, R, PMM)$ is the result of Hungarian algorithm, α is the experiential coefficient. The smaller $Cost(U, R)$ is, the more similar they are.

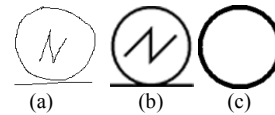


Figure 4. One input symbol and two references

III. EXPERIMENTAL RESULTS AND DISCUSSION

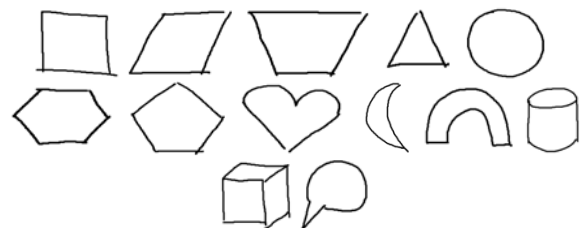
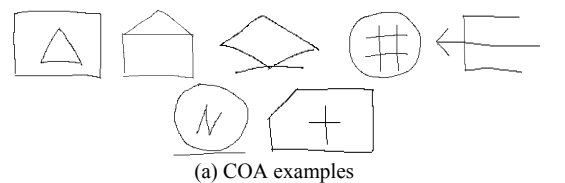


Figure 5. Examples of sketches

We evaluate our approach on two dataset: PowerPoint shape [7], course-of-action (COA) [8]. The former includes 13 types of simple symbols, while the later

composed of 120 geometric symbols taken from COA by us. Some examples are shown in Fig. 5. Each of them composed of lines, arcs and ellipse. In the experiment, each reference for a symbol is created by constructing an “average” PMM from 10 training examples. The data was collected using the Wacom Bamboo Pen Small CTL-660, at a spatial resolution of 100lines/mm and a accuracy of 0.25mm and the pen processing speed is 133points/s. Every sketching sample is drawing carefully, without restriction on the stroke order or stroke number.

Each symbol was normalized, as described in section II, using the value $\rho^2 = 5000$. Then the symbol was rotated to get a number of PMMs to obtain the minimum matching cost as described above, where $\beta=25^\circ$. Next we set $\alpha=0.33$ to punish the unmatched primitives using formulae(8). Fig. 6 shows one example of the experiment. The number below the graphics is the matching cost between input and reference.

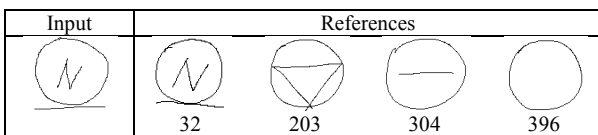


Figure 6. An example of sketch recognition

Table I shows the recognition result of our experiment. Where Top1, Top2 and Top3 denote the cumulative recognition rate of the first one, two and three candidates, respectively.

However, some symbols such as Fig. 7 are not recognized correctly. The reason is that there are two short lines in the symbol. The direction of short lines can vary in a relatively large range when user draws them. But in our algorithm we set a fixed threshold β to model the variance. It is not very reasonable and should be improved in the future.

TABLE I.
THE ACCURACY OF OUR EXPERIMENTS

	Top1	Top2	Top3
COA	91.8%	96.4%	98.0%
PowerPoint shape	95.3%	96.8%	98.2%



Figure 7. An example of unrecognized symbol

IV. CONCLUSION

In this paper, a structural method for sketch symbol recognition is proposed based on primitive correspondence using geometric features. The trajectory of input pattern is translated into primitives by stroke segmentation and primitive recognition. By the definition of PMM, we use the model of assignment problem of get the primitive correspondence. In this process, we also give the matching cost of two primitives using the simple geometric features. The invariant of rotation is also taken

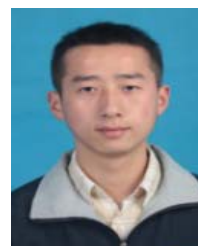
into consideration. Our approach permits both the stroke-number and stroke-order variations.

The proposed method is applicable to the carefully drawn symbols, similar to regular script in handwriting recognition. The advantages of our algorithm can be summarized as follows: (1) independent on stroke number and order; (2) structural method has good distinction of similar shapes; (3) This method is very simple, and can achieve fine accuracy and speed.

However, some disadvantage are listed below: (1) It is sensitive to the result of primitive recognition; (2) Have not taken neighbor primitives into consideration when matching two primitives. Considerable research still needs to be done to improve this technique including: finding more accurate algorithm for stroke segmentation and primitive recognition; taking probability into the matching processing and measure the structure element (primitives and their matching costs) probabilistically.

REFERENCE

- [1] T. Y. Ouyang and R. Davis, “A visual approach to sketch symbol recognition,” in *Proc. 21st International Joint Conference on Artificial Intelligence*, 2009, pp. 1463-1468.
- [2] B. Paulson and T. Hammond, “PaleoSketch: Accurate primitive sketch recognition and beautification,” in *Proc. Intelligent User Interfaces*, 2008.
- [3] T. Hammond and R. Davis, “Ladder, a sketching language for user interface developers,” *Computer & Graphics*, vol. 29, no. 4, pp. 518-532, 2005.
- [4] M. Oltmans, “Envisioning sketch recognition: A local feature based approach to recognizing informal sketches,” Ph.D thesis, Massachusetts Institute of Technology, Cambridge, MA, May 2007.
- [5] J. Almazan, A. Fornes, and E. Valveny, “A non-rigid appearance model for shape description and recognition,” *Pattern Recognition*, vol. 45, pp. 3105-3113, 2012.
- [6] C. L. Liu and M. Nakagawa, “Online Recognition of Chinese Characters: The State-of-the-Art,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 2, pp. 198-213, 2004.
- [7] H. Hse and R. Newton, “Sketched symbol recognition using Zernike moments,” in *Proc. 17th International Conference on Pattern Recognition*, Cambridge, 2004, pp. 367-370.
- [8] T. Hammond, D. Logsdon, J. Peschel, J. Johnston, P. Taelle, and A. Wolin, *et al.*, “A sketch recognition interface that recognizes hundreds of shapes in course-of-action diagrams,” in *Proc. CHI, Extended Abstracts on Human Factors in Computing Systems*, Atlanta, Georgia, USA, April 2010.



Wei Deng was born in 1986. He is currently a Ph.D. student at Key Lab. of the Academy of Equipment, Beijing, China. His research work is mainly focused on sketch recognition and multimedia technology.



Lingda Wu was born in 1962. She is a professor at Key Lab. OF the Academy of Equipment. Her research interests are mainly in virtual reality technology and multi-media technology.



Chao Yang was born in 1981. He is a assistant researcher at Key Lab. of the Academy of Equipment. His research interests are mainly in virtual reality technology and virtual simulation.



Zhongwen Zhao was born in 1974. He is a associate professor at Key Lab. of the Academy of Equipment. His main research interests include hardware design and requirements, data transmission, and system analysis and integration.