



On-Line Sketch Recognition Using Direction Feature

Wei Deng, Lingda Wu, Ronghuan Yu, Jiazhe Lai

► **To cite this version:**

Wei Deng, Lingda Wu, Ronghuan Yu, Jiazhe Lai. On-Line Sketch Recognition Using Direction Feature. Paula Kotzé; Gary Marsden; Gitte Lindgaard; Janet Wesson; Marco Winckler. 14th International Conference on Human-Computer Interaction (INTERACT), Sep 2013, Cape Town, South Africa. Springer, Lecture Notes in Computer Science, LNCS-8119 (Part III), pp.259-266, 2013, Human-Computer Interaction – INTERACT 2013. .

HAL Id: hal-01504888

<https://hal.inria.fr/hal-01504888>

Submitted on 10 Apr 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

On-line Sketch Recognition Using Direction Feature

Wei Deng, Lingda Wu, Ronghuan Yu and Jiazhe Lai

Academy of Equipment , Beijing 101416 , China

dengwei@whu.edu.cn

Abstract. Sketch recognition is widely used in pen-based interaction, especially as the increasing popularity of devices with touch screens. It can enhance human-computer interaction by allowing a natural/free form of interaction. The main challenging problem is the variability in hand drawings. This paper presents an on-line sketch recognition method based on the direction feature. We also present two feature representations to train a classifier. We support our case by experimental results obtained from the NicIcon database. A recognition rate of 97.95% is achieved, and average runtime is 97.6ms using a Support Vector Machine classifier.

Keywords: Sketched symbol recognition, NicIcon database, multi-stroke shapes

1 Introduction

Sketching is a natural form of human communication. Sketch-based interaction is a fast and efficient means of capturing information by automatically interpreting hand-drawn sketches and can be an important part of the early design process, where it can help people explore rough ideas and solutions in an informal environment. Sketch recognition refers to the recognition of predefined symbols or free-form drawings(e.g., an unconstrained circuit drawing); in the latter case, the recognition task is generally preceded by segmentation in order to locate individual symbols. This paper focuses on the recognition of hand-drawn isolated symbols. With the growing popularity of devices with touch screens, there is increasing interest in building sketch-based user interfaces.

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 will have different drawing styles, such as the stroke-order, -number, and non-uniform scaling , as well as complex local shifts. Moreover, the style of the same individual at different times may differ. A good recognition algorithm should place few drawing constraints on users. Related research is that of handwriting recognition, such as handwritten digit and Chinese character recognition with many effective algorithms.

This paper presents a new approach to sketched symbol recognition based on the direction feature. It is insulated from stroke order and number. We also present two feature representations, which obtain good experimental results in the NicIcon database.

2 Related Work

One common approach to sketch recognition focuses on building structural shape descriptions. Its basis step is stroke segmentation and primitive recognition using temporal and spatial features. A sketched symbol can be represented as a tree or graph and the similarity between two sketches can be calculated by structural matching. Hammond and Davis [1] developed a hierarchical language to describe how diagrams are drawn, displayed, and edited. They used this language to perform automatic symbol recognition. The attributed relational graph (ARG) is an excellent statistical model to describe both geometry and topology of a symbol [2], and is insensitive to orientation, scaling, and drawing order. The advantage of structural methods is distinguishing similar shapes. Therefore, high accuracy in stroke segmentation and primitive recognition is necessary.

An alternative approach looks at the visual appearance of shapes and symbols [3] instead of the relation between geometric primitives. Ouyang and Davis[3] proposed a visual approach to sketched symbol recognition, which is similar in spirit to our proposed method. It used a set of visual features that captured on-line stroke properties like orientation and endpoint location. Almazan et al. [4] and Escalera et al. [5] described a framework to learn a model of shape variability based on the Active Appearance Model (AAM), and proposed two types of modified Blurred Shape Model (BSM) descriptors(DBSM and nrBSM) as basic shape features to learn the model. Eitz et al. [6] described a large scale exploration of human sketches. They analyzed the distribution of non-expert sketches of everyday objects and developed a bag-of-features sketch representation. Willems et al. [7] explored a large number of on-line features, which were sorted in three feature sets due to different levels of details. Recently Delaye and Anquetil [8] presented a set of 49 features, called HBF49, for the representation of hand-drawn symbols for use as a reference for evaluation of symbol recognition systems.

This paper presents an on-line sketched symbol recognition method using the direction feature. The key contributions of our method are:

1. A framework to recognize sketched symbols using the direction features. It is insensitive to the variation in the order and number of strokes.
2. Two feature representations. They are statistical vectors rather than structural representations of geometric primitives. This allows it to be robust to different drawing styles.
3. Obtaining high recognition accuracy in the NicIcon database using Support Vector Machine (SVM) classifier in real time.

3 Our Approach

We designed our approach primarily using the direction properties of symbols. An overview of the recognition process is shown in Fig. 1. First, preprocess the input trajectory, including resampling. Then the direction of every point is decomposed to four directions, horizontal, vertical, diagonal, and anti-diagonal. Feature representation is performed using the grid and direction feature. Finally, an SVM classifier is trained to perform classification.

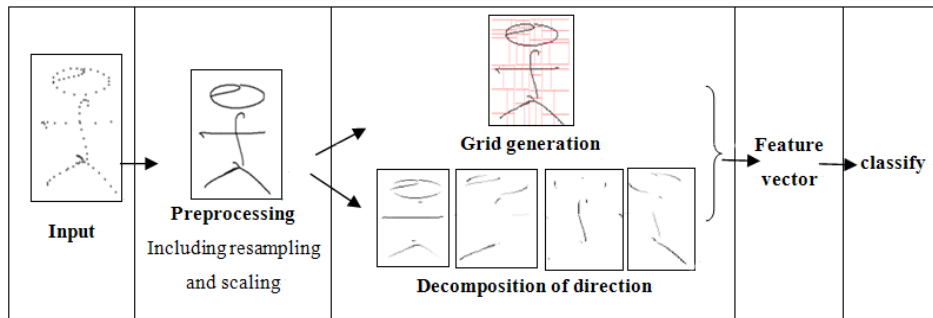


Fig. 1. System overview.

3.1 Preprocessing.

The preprocessing of the input sketch directly facilitates the pattern description and affects the quality of the description. We choose to normalize patterns in scale and translation, and to apply a trajectory resampling strategy. These operations are simple to perform and guarantee a better stability of extracted features, for any type of input pattern.

Since on-line strokes are typically sampled at a constant temporal frequency, the distance between neighboring points varies based on the pen speed. This produces more sampled points where the pen is typically slower. In order to make feature extraction more reliable, we resample each stroke at a constant spatial distance. In our experiments the sampling interval is set to one. Next we remove differences of scale and translation. The coordinates of stroke points are simply shifted and scaled such that all points are enclosed in a standard box. In experiments we set $x, y \in [0, 100]$. It means translating maximal dimension of a symbol to 100 with aspect ratio preserved [10]. We do not translate the on-line trajectory to image (such as in [3]). Therefore direction features are accurate with the absence of jagged edge.

3.2 Decomposition of direction

The so-called direction feature is widely used in character recognition. Similar to Ouyang and Davis [3], we decompose direction vectors of resampling points into four

directions, at 0, 45, 90 and 135 degrees. If the direction of a point lies between two standard directions, it is decomposed into two components in the two standard directions, as shown in Fig. 2. One major advantage of this representation is the independence of stroke direction. A stroke drawn left to right is the same as one drawn right to left. This is essential for free hand-drawn symbols.

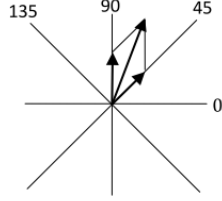


Fig. 2. Decomposition of direction

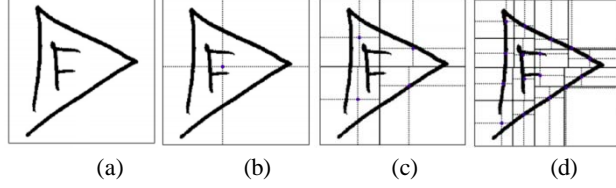


Fig. 3. The procedure of grid generation [4]. (a) original shape points. (b) four sub-regions split by geometric centroid. (c) and (d) iterative procedure to obtain grids of 4x4 and 8x8 respectively.

3.3 Grid generation

Grid is used to blur the features, for directly overcoming the local non-uniform variability. We learned that blurring should be considered as a means of obtaining latent dimensions(subspace) rather than as a means of reducing computational cost, though the effects might seem similar [9]. Moreover, the mesh size of 8x8 used in statistical approaches was determined by the optimum blurring parameter in light of the Shannon sampling theorem [9]. We refer to the grid generation by Almazan et al.[4]. First, we calculate the geometric centroid of all resampling points. Then the region is split into four sub-regions. Therefore, the points are split into four sets. Next, in each sub-region the iterative procedure works until we get a grid of size 8x8, as shown in Fig. 3.

3.4 Feature representation

The representation schemes of the input sketch and the model database are important since the classification method depends strongly on them. We present two feature representations. The size of the grid is 8x8, and the sub-direction feature is $S = \{x_i, y_i, dir_{1i}, dir_{2i}, dir_{3i}, dir_{4i}\}$, $i=1,2,\dots,n$, where n is number of resampling points; $\{x_i, y_i\}$ is coordinate of point P_i ; $dir_{1i}, dir_{2i}, dir_{3i}$ and dir_{4i} describe sub-direction strength of P_i in above four directions, respectively. One simple feature representation is to cumulate sub-direction strength in each grid cell. That is

$$V = \{v_j\}, \quad v_j = \left\{ \sum_{p_i \in m_j} dir_{1i}, \sum_{p_i \in m_j} dir_{2i}, \sum_{p_i \in m_j} dir_{3i}, \sum_{p_i \in m_j} dir_{4i} \right\}$$

where $i=1,2,\dots,n$, $j=1,2,\dots,8^2$; and m_j is the j^{th} cell in the grid, P_i is the i^{th} point in sampling points, v_j is direction feature of m_j . So we can form a 8x8x4 dimensional

vector, which can be used in a statistical classifier. We denoted this representation as ‘Dir’, compared with below ‘DirBSM’.

Another feature representation is presented, called Directional Blurred Shape Model (DirBSM), is presented here. BSM[4,5] is a shape descriptor, whose main idea is to describe a given shape by a probability density function encoding the probability of pixel densities of a certain number of shape sub-regions. Given an image of a particular symbol, each pixel contributes to compute the BSM descriptor. This is done by dividing the given image in a $k \times k$ grid (cells). Then, each cell receives votes not only from the shape pixel located inside its corresponding cell, but also from those located in the adjacent cells. Thereby, every pixel contributes to the density measure of its sub-region cell, and its neighboring ones. This contribution is weighted according to the distance between the point and the centroid of the cell receiving the vote.

However, in our approach the shape is not an image but a set of shape points with four direction attributes. We have modified the BSM descriptor to DirBSM. For each direction attribute, we calculate a BSM vector, as illustrated in Fig. 4. Then four vectors can form a final shape descriptor, which is the feature representation of a symbol.

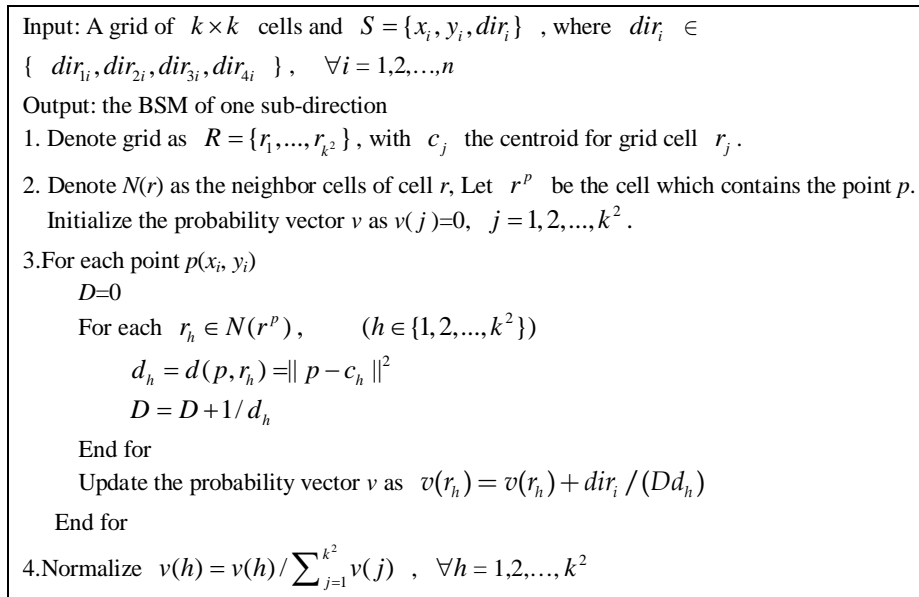


Fig. 4. Getting BSM vector of one direction feature

To modify the feature distribution for improving the classification performance, all the measurements in the feature vectors are transformed by variable transformation $y=x^p$ [10]. In our experiments we tested different values of p to obtain optimal parameter.

4 Experiments

In this section, we show the performance of the proposed method for sketch recognition tasks using above two feature representations.

4.1 NicIcon database

The NicIcon database [11] is composed of 26,163 sketched symbols of 14 classes in the domain of crisis management and incident response systems. An example is shown in Fig. 5. The database comes from 34 different writers and it is commonly used for on-line symbol recognition, but off-line data is also available. The database is already divided into three subsets (TRAINING, TEST and EVALUATION sets) for both writer dependent and independent settings. Approximately, 36%, 24% and 40% of symbols are contained in the training, test and evaluation sets, respectively. We have selected the on-line data that is writer-independent, because it is the most similar to human-computer interaction.

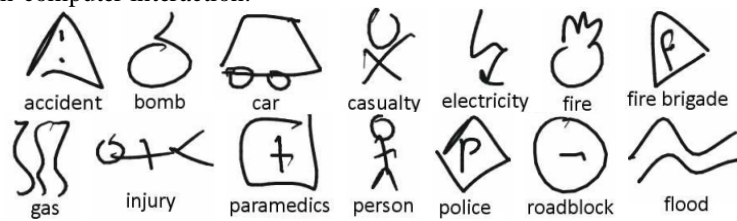


Fig. 5. Samples of 14 different classes of NicIcon database

We compare two kinds of feature representations (Dir and DirBSM). Training samples are used to train a classifier and test samples are used to test recognition accuracy. We select the LibSVM toolbox with default parameters.

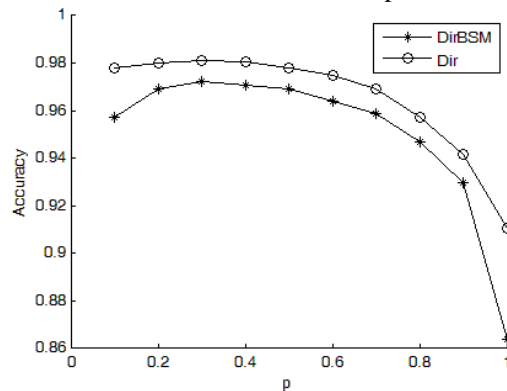


Fig. 6. Getting optimal p using training set and test set

In order to get optimal meta-parameter p, we train a classifier using the TRAINING set and test its performance using the TEST set. Fig. 6 shows the accuracy with

parameter p . The optimal p is $p=0.3$. So all the measurements (both in training and testing) in the feature vectors are transformed by variable transformation $y=x^{0.3}$ [10]. Then we use both the TRAINING and the TEST sets for training the classifier and the EVALUATION set for measuring the recognition performance (T+V/E). The accuracy rates are 95.57% (DirBSM) and 97.95% (Dir), respectively.

4.2 Results and discussion

In Fig. 6, when p is small, the accuracy of Dir is better than that of DirBSM. The reason for this may be that the BSM descriptor blurs the shape and cannot distinguish shapes in details.

As Table 1 indicates, on this dataset we achieved an best accuracy rate of 97.95%, compared to other results in the literatures [4,7,8].

Table 1. Accuracy compared with other methods

Methods	Accuracy
(a) combination of features[7]	92.63% *
(b) NRAM+nrBSM(SVM)	94.29% *
(c) NRAM+nrBSM(NN)	95.18% *
(d) HBF49[8](SVM)	97.44%
our approach(SVM)	95.57 (DirBSM)
	97.95 (Dir)

*The values of accuracy rate in (a), (b) and (c) come for reference[4]. And the experiments in (b) and (c) used off-line data in NicIcon database.

For improvements in the recognition system, we analyzed misrecognition factors in the experiments. The top four false-positive symbols are accident, bomb, paramedics and electricity. It means there are other symbols that were misrecognized as them. And the top four false-negative ones, which were misrecognized as other symbols, are fire, casualty, police and fire brigade. Factor-1 is the large distortion in sketching. Some of these sketches cannot even be recognized by people, as shown in Fig. 7. Factor-2 is the presence of extra strokes or missing ones. Factor-3 is the similarity in direction features of sketches, mainly between the symbols gas and flood. Some distorted symbols are similar to different patterns, when they are represented by direction features. These may be resolved in terms of introducing other useful features.



Fig. 7. Misrecognized samples. (a) too large distortion; (b) and (c) distortion of triangle becoming a rectangle; (d) and (e) extra stroke; (f) distortion of rhombus becoming a trapezium

4.3 Runtime Performance

Finally, we evaluate the runtime performance of our approach. We use Matlab to program this method on a 3.10*2 GHz machine (Inter Core™ i5-2400). The average runtime of recognizing one symbol is 97.6ms, including extracting feature and classifying with SVM-RBF. This method should therefore realistically be usable in real-time human-computer interaction.

5 Conclusions

This paper presents an on-line sketch recognition method based on the direction feature. It is a statistical method, which uses the visual feature of shape points instead of regarding the shape as an image or temporal sequences of geometric primitives. We also present two feature representations to train an SVM classifier. Experiments show that in the NicIcon database fine results are achieved, including high accuracy rate and real-time effects. This method is insensitive to stroke order and number. In future, we will do more experiments using larger databases with more classes, and introduce more feature to enhance accuracy rate.

References

1. Hammond,T., Davis,R.: LADDER, a sketching language for user interface developers. *Computer & Graphics*,29(4):518-532(2005)
2. Lee,W., Kara,L.B., Stahovich,T.F.: An Efficient Graph-Based Recognition for Hand-Drawn Symbols. *Computers & Graphics* 31, 554-567(2007)
3. Ouyang,T.Y., Davis,R.: A visual approach to sketch symbol recognition.Proceedings of the 21st International Joint Conference on Artificial Intelligence(IJCAI-09) (2009)
4. Almazan,J., Fornes,A., Valveny,E.: A non-rigid appearance model for shape description and recognition. *Pattern Recognition*, 45,3105-3113(2012)
5. Escalera,S., Fornes,A., et al.: Blurred shape model for binary and grey-level symbol recognition. *Pattern Recognition Letters*,30,1424-1433(2009)
6. Eitz,M., Hays,J., Alexa,M.: How do humans sketch object?. *ACM Transactions on Graphics, Proc. SIGGRAPH* (2012)
7. Willems,D., Niels,R., Gerven,M.V., et al.: Iconic and multi-stroke gesture recognition. *Pattern Recognition*, 42,3303-3312(2009)
8. Delaye,A., Anquetil,E.: HBF49 feature set: A first unified baseline for online symbol recognition. *Pattern Recognition*, 46, 117-130(2013)
9. Fujisawa,H.: Forty years of research in character and document recognition—an industrial perspective. *Pattern Recognition* 41,2435-2446(2008)
10. Liu,C.L., Nakashima,K., Saho,H., et al.: Handwritten digit recognition: investigation of normalization and feature extraction techniques. *Pattern Recognition*, 37,265-279(2004)
11. Niels,R., Willems,D., Vuurpijl,L.: The NicIcon database of handwritten icons. 11th International Conference on frontiers in Handwriting Recognition (ICFHR 2008), Montreal, Canada, August 19-21(2008)