Signal Processing 110 (2015) 123–131

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Online gesture-based interaction with visual oriental characters based on manifold learning

Yi Wang [a], Juncheng Liu [b], Xin Fan [a,*], Xiangjian He [c], Qi Jia [a], Renjie Gao [a]

[a] School of Software, Dalian University of Technology, Dalian 116620, China
[b] School of Electronics Engineering and Computer Science, Peking University, Peking 100000, China
[c] School of Computing and Communications, University of Technology, Sydney, Australia

ARTICLE INFO

Article history:
Received 28 April 2014
Received in revised form
23 August 2014
Accepted 29 August 2014
Available online 16 September 2014

Keywords:
Human–computer interface
Visual oriental characters
Feature descriptors
Graph-optimized locality preserving projections

ABSTRACT

Online gesture-based interaction with characters has become a more natural and informative human–computer interface with the popularity of new interactive devices (e.g., Kinect and Leap Motion). In this paper, a new feature descriptor named Segmented Directed-edge Vector (*SDV*) is proposed. This simple and yet quite effective descriptor is able to capture the characteristics of visual oriental characters. Moreover, we explicitly build the mappings from *SDVs* to features in a subspace by a modified Locality Preserving Projections (LPP) method with stroke class constraints. These mappings can yield meaningful subspace structures for larger character sets. Extensive experiments on the online interactive system demonstrate the robustness of our method to various issues in gesture-based character's input, such as unnatural breaks, overlapped or distorted radicals, and unconscious or quivering trajectories. Our system can still achieve accurate recognition when accumulative errors occur with complex characters.

1. Introduction

One of the most attractive means of natural human computer interaction (HCI) is gesture-based character input and recognition (GCIR) that provides more interactive controls [1]. GCIR has great potentials in real applications, such as disable assistance, interactive computer games, smart appliances and writing trainings. In the field of computer vision (CV), GCIR can be regarded as the merge of character recognition and gesture recognition. Researchers have developed numerous approaches in both areas [1–5], but their feature extractors and classifier learning algorithms may be inapplicable to GCIR. On one hand, various 'static' features, such as gradient features, Gabor features and SIFT, are widely used for handwritten character recognition [6–8]. Nevertheless, radicals or strokes of a character are likely to be overlapped or distorted in the context of GCIR that is different from the traditional way of writing, does not have a writing plane (paper or touch screen) and does not have any demarcation for writing. On the other hand, classical classifiers including Hidden Markov Models (HMM) [9] and Conditional Random Field (CRF) [10] provide an elegant framework to label gestures. Unfortunately, these methods can hardly render the real-time performance for larger label sets, when there are, for example, thousands of or more gestures (characters). GCIR also demands an incremental learning ability that embraces new characters and writing styles in an online fashion. Furthermore, learning strategies in GCIR have to be robust to outliers brought by unconscious quivering trajectories and accumulative errors.

In this paper, we put forwards a novel online interactive character's input and recognition method by gestures based on manifold learning. We utilize the directional and ordered information of gestures in GCIR to design a new descriptor that has a good discriminative ability in recognition.

* Corresponding author. Tel.: +86 411 87571638.
E-mail address: xin.fan@ieee.org (X. Fan).

http://dx.doi.org/10.1016/j.sigpro.2014.08.042
0165-1684/

We study subspace learning techniques and learn explicit mappings to recover intrinsic configurations for numerous characters, and obtain real-time and high recognition rate. The main contributions of this paper are given below:

(1) We propose a new stroke-level feature representation for dynamic gestures, named Segmented Directed-edge Vector (*SDV*). *SDV* is invariant to scale variations, while combining both geometrical and dynamical information of trajectories. *SDVs* have a low repetition rate on a large character set, and have a reasonable tolerance of writing fault.
(2) We use *SDVs* and a modified Locality Preserving Projections (LPP) [11,12] to obtain robust and efficient recognition results on large character sets. Stroke class-specific constraints are incorporated in order to generate more meaningful subspace structures and explicit mapping functions for classification. We are able to achieve real-time recognition by applying a parallel searching technique.
(3) We implement the proposed practical gestures-based character input and recognition (GCIR) system using depth images from monocular videos captured by a Kinect sensor and the features obtained in (2) above. This system can tackle unnatural breaks, overlapped or distorted radicals, and unconscious or quivering trajectories in real applications. Moreover, we build a semantic associative database that can accelerate the speed of the system further.

2. Related work

In recent years, studies in the computer vision area on motion recognition and shape reconstruction have witnessed a growing interest in subspace analysis and manifold learning techniques [13–15]. Given a set of high-dimensional data points, manifold learning techniques aim at discovering the geometric properties in a data space, including the Euclidean embedding, intrinsic dimensionality, connected components and homology. Manifold learning techniques can be classified into linear (MDS [16], LDA and PCA [17], etc.) and non-linear (ISOMap [18], LLE [19], LE [20], RML [21], LTSA [22], etc.) methods, which have been applied to face and gait recognition and show impressive performance.

Locality Preserving Projections (LPP) [11] attempt to find optimal linear approximations to the Eigen-functions of the Laplace Beltrami operator on a manifold. This technique seeks to preserve the intrinsic geometry of data and the local structures. In contrast to other manifold learning algorithms, LPP possess a remarkable advantage that can generate an explicit mapping. Furthermore, compared with HMM or CRF methods, LPP are defined everywhere of training data and may be simply applied to any new data. Many improvements on LPP have emerged in the previous few years. Yen et al. [23] proposed an orthogonal neighborhood preserving discriminant analysis (ONPDA) method, which effectively combines the characteristics of LDA and LPP. Wong and Zhao [24] proposed two feature extraction algorithms derived from LPP, i.e., the supervised optimal locality preserving projection (SOLPP) algorithm and the normalized Laplacian-based supervised optimal locality preserving projection (NL-SOLPP). These algorithms use both local information and class information to model the similarity of data.

The construction of a neighbor weight graph is the key to subspace learning algorithms [11–15]. Recent studies show the influence of the graph construction on clustering measures and resultant subspace representation. Traditional construction methods using k-nearest neighbors typically lead to an unbalanced graph and thus unfavorable performance. Zhang et al. [12] developed an unsupervised Graph-optimized Locality Preserving Projections (GoLPP), which incorporated graph construction into the LPP objective function, and thus obtained a joint learning framework for graph construction and projection optimization. GoLPP produce a changeable graph instead of a fixed one in LPP. The graph is gradually updated in an iterative process, and naturally takes transformed data. However, the adjacent graph of GoLPP is initialized by the traditional k-nearest neighbor method without class information. Therefore, the initialization may take some heterogeneous samples for optimization. Yu et al. proposed a hypergraph-base manifold learning in [25]. By varying the neighborhood size, they generated a set of hyperedges for each image and its visual neighbors. The joint learning of the image labels and hyperedge weights automatically modulates the effect of hyperedges. In [26], they adopted sparse representation to select a few neighbors of each data point that span a low-dimensional affine subspace passing near that point. After that, the whole alignment strategy is utilized to build the manifold.

3. Framework of the proposed approach

In this paper, we focus on developing a gesture based and real-time recognition method with interactive input on large visual oriental character sets. We name the trajectory of a control hand from the beginning to the end in the process of writing a character, including the transition edges between strokes, as a Visual Oriental Character (VOC). Oriental characters refer to stroke and structure based characters, such as the characters in Chinese, Japanese and Korean. Examples for structured oriental characters are shown in Fig. 1. Besides, English letters and Arabic numerals can also be written in this style using gestures.

Our approach consists of four steps, i.e., control hand recognition and tracking, directed-edges quantified and optimization, feature extraction and selection, and large dataset training and classification, as shown in Fig. 2. In the off-line pre-processing stage, *SDVs* are computed for VOCs in training sets. Then, a subspace learning algorithm is adopted to learn explicit mappings of features from high-dimensional feature spaces to features in low-dimensional embedding spaces. Moreover, a semantic associative database is built to accelerate the speed of input. In the on-line processing stage, we segment the control hand from depth images obtained from monocular videos captured by a Kinect sensor. And a stroke tracking and optimization method generates *SDVs* for input VOCs. Finally, the recognition among embedding spaces can be achieved by explicit mapping functions and the k-nearest neighbor searching.

4. Feature extraction

4.1. Directed-edges quantified and optimization

Hand detection and tracking are important steps prior to feature extraction that influences the recognition accurate rate. As the Kinect sensor combined with an infrared sensor is capable of adding depth information to 2D images, it is more reliable to segment hands from various complex backgrounds. Besides, the Kinect for Windows SDK supports human skeleton with joint recognition and tracking, so it is easy to get the coordinates of an operator's hand end in depth space. In our work, we take the operator's right hand end as his or her valid control hand.

A directed-edge is defined as the vector between two turning valid control points, and its direction is consistent with the writing order. Direction-edges are often quantified approximately into discrete directions. Different from the uniform eight-direction quantifying [4], we take into account the characteristics of gesture based input which is controlled by human's arm linkage mechanism. A test of drawing fifty Chinese characters ‘in the air’ by ten people is conducted. The results show that the edges of horizontal and vertical directions are easily drawn, but those edges in slanting directions are not. Therefore, we enlarge the quantified range of the even directions to $\pm$arctan(1/3) and that of the odd directions to $\pm(\pi/4$-arctan(1/3)). In order to make trembling lines or curves be standard directed-edges (see some examples in Fig. 3), we adopt the abstraction-integrate algorithm based on polygon approximation proposed in [27]. For some characters with curves, more than one style can be added to improve the robustness of the recognition.

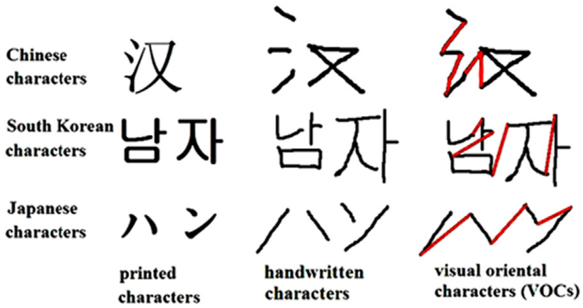

Fig. 1. Examples of three types of oriental characters with their VOCs.

4.2. Segmented directed-edge vector

Simply, we can take an eight dimensional directed-edge vector $DV=\{d_1, d_2, d_3, \ldots, d_8\}$ as the feature for a VOC, where each entry corresponds to a direction and its value is the total number of directed-edges performed in that direction. *DVs* have both statistic and ordinal information for VOCs, but lack strong discriminate abilities. For example, two simple different VOCs have the same *DVs* in Fig. 4(a). Furthermore, as VOCs become complex, the probabilities of the directed-edges falling in eight directions are prune to equal values, and thus the repetition rate of VOC sets greatly increases.

We propose a segmented directed-edge vector (*SDV*) as a new presentation for the VOC. *SDV* is defined as:

$$SDV^{sn} = \{DV_1^{sn}, DV_2^{sn}, \ldots, DV_{dn}^{sn}\} = \{V_1, V_2, \ldots, V_{tn}\}, \tag{1}$$

where sn is the segmented size, and there are sn direction-edges assigned to the directional entry i $(i=1,\ldots,dn)$.

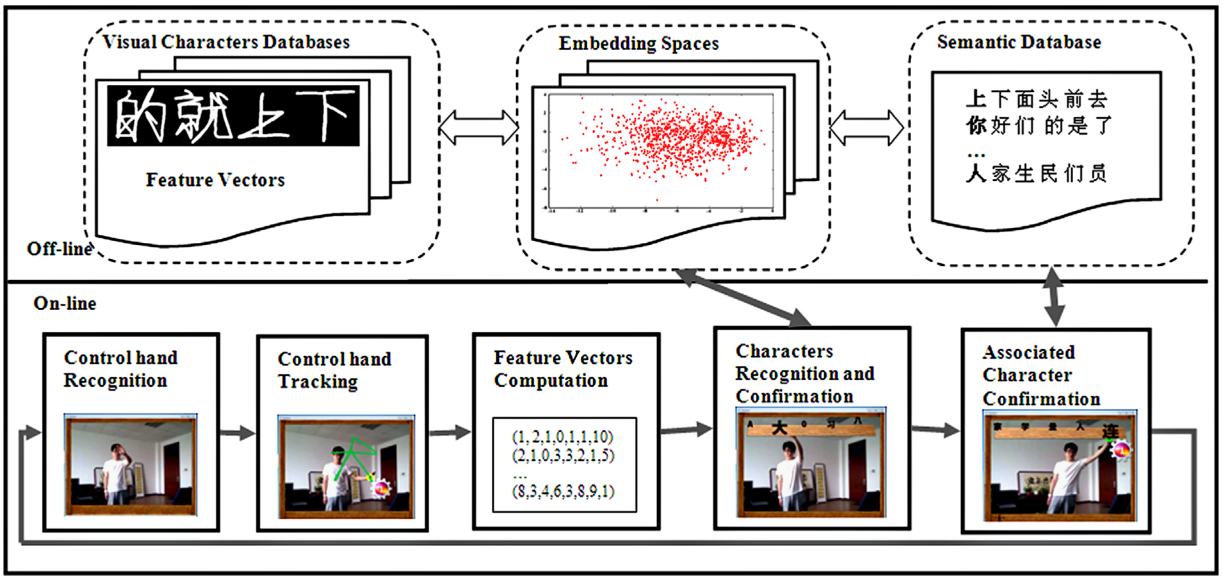

Fig. 2. The framework of gesture-based character input and recognition through manifold learning.

sn Influences the discriminant ability. *dn* is determined by *sn* and represents the total number of directed-edges in a VOC. *tn* is the total number of components in a *SDV*. Both *sn* and *tn* are artificially set beforehand. *tn* ($tn \geq dn \times 8$) is always larger than the total number of strokes of the most complex VOC in a training set. Furthermore, we make zero padding in the short vector.

SDVs are simple and yet effective representations for VOCs, and have the merits as follows:

(1) *SDVs* have low repetition rates even when the VOC sets are large. To our best knowledge, there is no available pubic database for VOCs so far, and some of the existing online character databases such as Kuchibue and Nakayosi (for Japanese), SCUT-COUCH 2009 (for Chinese), and CASIA-OLHWDB (for Chinese) do not contain the trajectory data between strokes [4], so we have to build a new VOC database. It is reported that one thousand Chinese characters are enough for daily communication in an article [28], so we build a new feature database VC_0, in which every SDV is set to have *tn* dimensions.

(2) *SDVs* have the ability to make similar characters have similar geometric and statistic features with a good choice of *sn*. If *sn* is set to be too small, *tn* increases. A larger *sn* corresponds to a lower discrimination. If *sn* equals to the total number of direction-edges of a VOC, *SDVs* degenerate to *DVs*. For example, in Fig. 4(a), *SDVs* of two different simple characters are the same with a large *ns*, but different with $sn=5$ in Fig. 4(b). Other examples for two complex characters are shown in Fig. 5. There are three settings of *sn*, i.e., the total number of direction-edges of *DVs*, $sn=10$, and $sn=5$, and their *SDVs* are illustrated in Fig. 5(a and g), Fig. 5(b, c) and (h, i), and Fig. 5(d, e, f) and (j, k, l) respectively. We can see from these figures that the first eight components of their *SDVs* are of the same geometry when $ns=5$, reflecting the same parts of the two characters. This property is quite important to preserve the neighborhood structure in the manifold learning steps as discussed in the next section. And with some experiments, we found that $ns=5$ is the best choice for training set VC_0. The repetition rate of *SDVs* (with $ns=5$) is 0.8% in contrast to 2.5% of DVs.

(3) *SDVs* have fault tolerance for wrong VOCs. *SDVs* can reduce calculation errors in eight directions to some extent. A directed-edge written in a wrong direction only influences at most eight components and leaves out the others.

(4) *SDVs* are scale invariant for VOCs. We develop *SDVs* upon the statistics for directed-edges that are independent of the length of a single directed-edge.

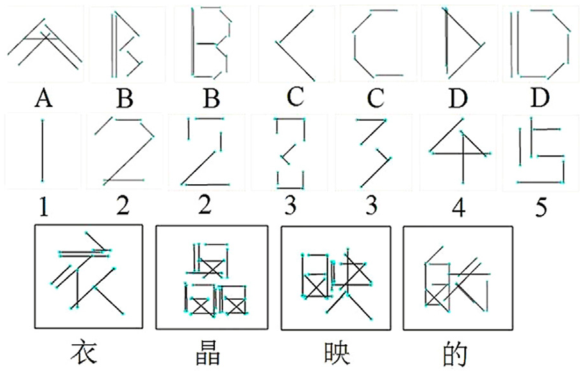

Fig. 3. Some examples for quantified VOCs generated by directed-edges quantified and optimization algorithm in training sets. For clear demonstration, we make every vertex of segments some pixels apart in this graphic representation.

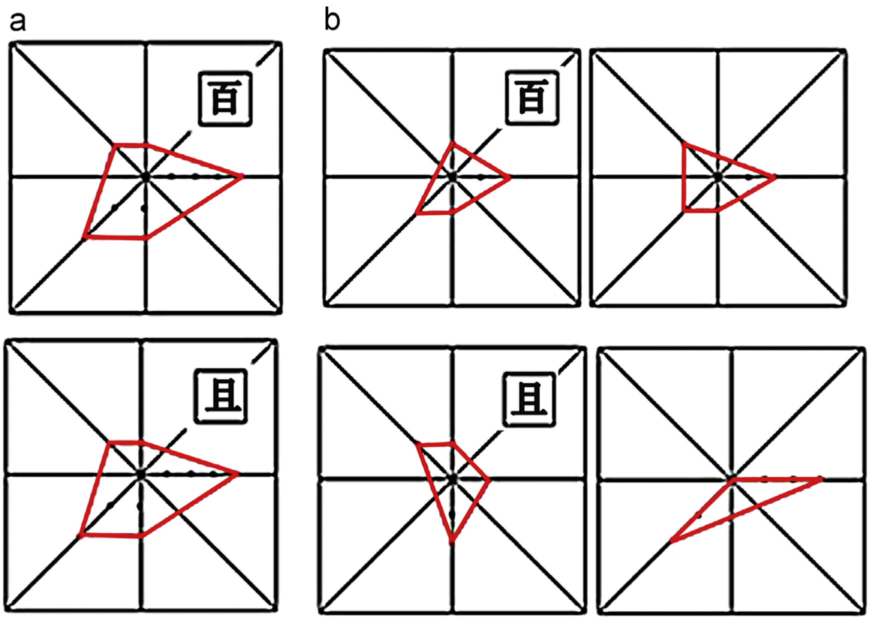

Fig. 4. Schematic *SDVs* for two simple Chinese characters: (a) *SDVs* are same with $ns=10$; (b) *SDVs* are different with $ns=5$.

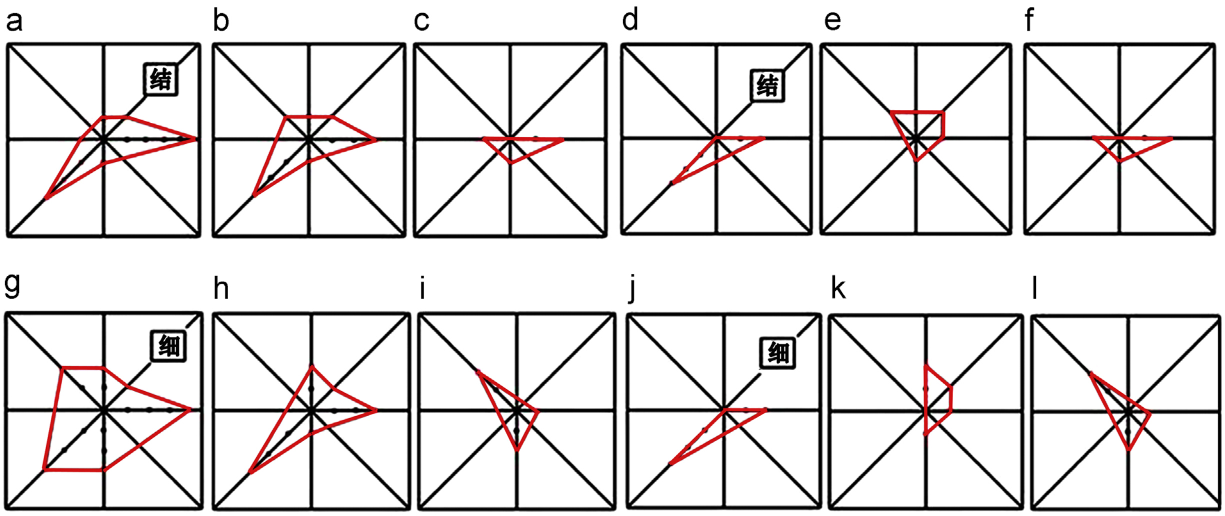

Fig. 5. Schematic *DVs* and *SDVs* of two VOCs. (a, g): *DVs*; (b, c, h, i): *SDVs* with $ns=10$; (d, e, f, j, k, l): *SDVs* with $ns=5$.

5. Learning embedding spaces

In this section, we use *SDVs* and subspace learning methods to obtain robust and efficient recognition for a large VOC dataset. The algorithmic procedure of our C-GoLPP (classified GoLPP) based algorithm for visual character manifold learning is stated as follows.

5.1. Constructing initial fuzzy adjacency graph

Here, we incorporate radicals and the total number of strokes as category constraints since radicals and strokes are typically used for index of words in a dictionary of oriental characters (such as Chinese characters). Characters belonging to same radical class are partly similar, and similar characters probably have similar numbers of strokes. Therefore, in our method, characters are firstly classified by radicals and listed according to the total number of strokes in every radical class. Then, characters having the same stroke number form a sub-class in a radical class. These category constraints are stored in a fuzzy membership degree matrix U such as the one used in FCM [29]. For example, a matrix with four adjacent classes is shown in Table 1, where adjacent neighbors are specified in advance. The adjacent graph for GoLPP [11] made by U will work during all iterations.

In order to generate the similarity of two feature vectors, the difference between two *SDVs* is considered, so is the difference of the total numbers of directed-edges (denoted by Ns) as shown in (2). The parameter a ($a \le 1$) determines the weights of these two types of differences. This similarity includes the global geometry of a character, so it is robust to the stroke sequence order varying with users. Q_{ij} in (3) represents the unified possibility that VOC_i belongs to class j.

$$P_{ij} = \exp(-(\alpha||SDV_i - SDV_j||^2 + (1-\alpha)|Ns_i - Ns_j|)/\sigma_1) \quad (2)$$

$$Q_{ij} = \sum_{k=1}^{Nc} (u_{ik} - u_{jk})^2 \quad (3)$$

Table 1
An example for fuzzy membership degree matrix U.

Radical class no.	The total number of strokes	SDV_1	SDV_2	…	SDV_N
1	1	0.05	0.25	…	0
1	2	0.2	0.5	…	0
1	3	0.5	0.25	…	0
1	4	0.2	0	…	0.25
1	5	0.05	0	…	0.5
1	–	0	0	…	0
1	Ns_i	0	0	…	0
2	–	0	0	…	0
–	–	0	0	…	0
Nr	–	0	0	…	0

$$Nc = \sum_{k=1}^{Nr} (Nr \times Ns_i) \quad (4)$$

When the fuzzy membership degree matrix is available, similarities between two VOCs in the adjacent matrix S are assigned by

$$S_{ij} = P_{ij} Q_{ij} \quad (5)$$

5.2. Computing explicit mapping functions

Let the set of input instances be $FV = \{Fv_i = SDV_i \in R^d, i = 1, ..., N\}$. Let us also assume that the set of the points in the embedding space corresponding to the Fv_i ($i=1,2,\ldots, N$) is $Y = \{y_i \in R^e, i = 1, ..., N\}$. In the above representations of FV and Y, d ($d = tn$) is the dimensionality of the feature space and e is the dimensionality of the embedding space. The explicit mapping functions are computed in the following steps. For more details, please refer to [12].

Step 1: Calculate the explicit mapping matrix W by solving the following generalized eigenvalue problem in (6) and (7).

$$FvLFv^T w = \lambda FvDFv^T w \quad (6)$$

$$Fv_i \rightarrow y_i = W^T Fv_i \tag{7}$$

where D is a diagonal matrix whose entries are column sums of S. $L = D - S$ is a Laplacian matrix. Both matrices $FvLFv^T$ and $FvDFv^T$ are symmetric and positive semi-definite.

Step 2: Update adjacent matrix as shown in (8), where η is a trade-off parameter.

$$S_{ij} = \frac{\exp(-\|W^T Fv_i - W^T Fv_j\|^2/\eta)}{\sum_{j=1}^{n} \exp(-\|W^T Fv_i - W^T Fv_j\|^2/\eta)} \tag{8}$$

Step 3: Calculate the value of objective function $J(W, S)$ as shown in (9).

$$\min_{W, S_{ij}} J(W,S) = tr[(W^T FvFv^T W)^{-1} W^T FvLFv^T W] + \eta \sum_{i,j=1} S_{ij} \ln S_{ij}$$

$$s.t. \sum_{j=1}^{n} S_{ij} = 1, \quad (i = 1, ..., n), \qquad S_{ij} \geq 0 \tag{9}$$

Step 4: Literately conduct Step 2 and Step 3, until the difference between two adjacent values of $J(W,S)$ is less than the threshold value. Meanwhile, an explicit mapping function (7) is computed.

5.3. Subspace learning for training data

We use a writing pad to generate a large number of characters from which we calculate the *SDVs* of Zhengkai-font and standard stroke order for offline training. Fig. 6 illustrates the embedding spaces for a large Chinese character dataset VC_0 by C-GoLPP and GoLPP respectively. Then, we select 140 Chinese VOCs to compare C-GoLPP and GoLPP on the performance of subspace learning for a smaller data set as shown in the Fig. 7. Figs. 6 and 7 show that both methods can reveal the mode of training data when the size of data set is small. Similar characters in the high dimensional data space also locate closely and present regular structures in the 2D embedding space. When the data set becomes large, our method performs more stable so that all the characters are evidently clustered according to their categories and stroke numbers.

5.4. Subspace learning with combined features

For offline handwritten character recognition, image based offline features (e.g. Gabor, Gradient, SIFT, Character-SIFT, etc.) are used in manifold learning methods [30–31]. They are discriminative when working with deformable handwritten characters, but these features can hardly work well for VOCs. VOCs exhibit more complicated shapes with additional trajectories between valid strokes as seen in Fig. 8. Meanwhile, users cannot always have a good control of their hands when writing in the air unlike writing on paper. Consequently, sever self-occlusion or overlapping

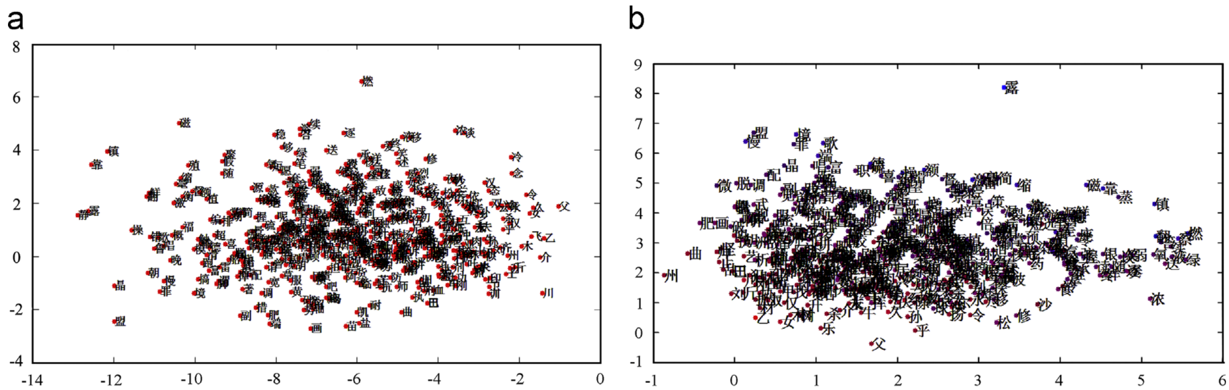

Fig. 6. The embedding spaces for dataset VC_0 by our C-GoLPP in (a) and GoLPP in (b).

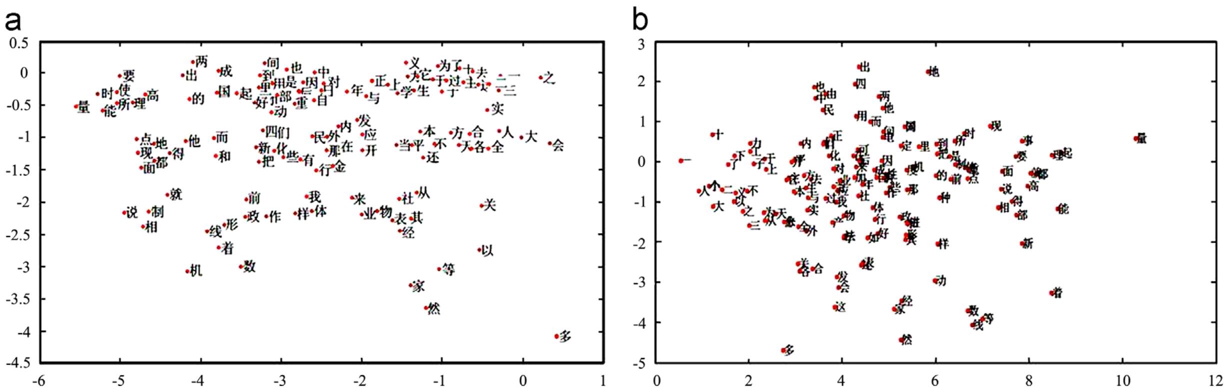

Fig. 7. Embedding spaces for a small dataset by (a) C-GoLPP and (b) GoLPP.

may occur on VOCs, which greatly challenges image-based features. For stroke-based *SDVs*, such overlaps may only influence a small fraction of the feature vectors. More importantly, *SDVs* make use of the sequence or order information of VOCs that is able to discriminate characters sharing similar shapes. In [8], overlapping trajectories and strokes are identified by the directions of fingers. This explicit identification is not always accurate and brings additional computational complexities.

For some cases (such as having no overlaps or writing in wrong orders), image-based feature will do some help, so we also include a set of image-based features denoted by $IV=\{Iv_i\}$, i.e., Hu moments, with a weight f_w as complementary features to *SDVs*. This inclusion is able to improve the recognition accuracy since these image-based features are informative for recognition in normal cases without self-overlaps. The similarity of image-based features is represented in (10), and similarities between two characters in the initial adjacent matrix S are assigned by (11).

$$I_{ij}=\exp(-\|Iv_i-Iv_j\|^2/\sigma_2) \tag{10}$$

$$S_{ij}=f_w P_{ij} Q_{ij}+(1-f_w) I_{ij} \tag{11}$$

5.5. Personalized subspaces

The frequency usage of characters may vary with users, for every person has his/her own daily words. It is useful to bring users' verbal habits into to subspace learning naturally and gradually. The issue is an incremental learning for manifolds which has been widely studied in recent years [32–33]. LPP based manifold learning algorithms obtain a locality preserving subspace for training data [12][33], and it has more capability to absorb new data to some extent by the learnt explicit mappings. As seen in Fig. 9, three new Chinese characters, each surrounded by a box, locate near the similar ones in the subspace. We attempt to update the explicit mappings made by the LPP in order to accommodate new training examples specific to a certain user.

As the amount of data increases, the new mode of dataset should be updated. We could adopt an incremental semi-supervised subspace learning algorithm as proposed in [33] to create new personalized embedding spaces in an online fashion. It is worth noting that our *SDVs* make the updating easier as one can automatically estimate the stroke class from a VOC, for the total number of directed-edges in a VOC is twice of the number of strokes in an oriental character. Thus, we are able to put new *SDVs* to the classes having the same stroke number and update the fuzzy membership degree matrix U and the adjacent matrix S. Fifty Chinese characters are selected randomly to verify the relationships between strokes and directed-edges. The statistics are shown in Fig. 10, where the ratio of the two is about 0.5, especially when characters are getting more complex.

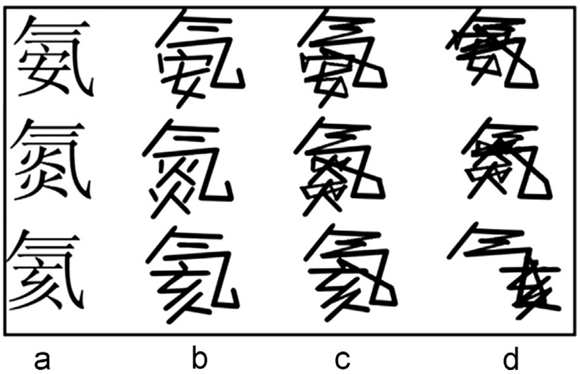

Fig. 8. An example for different style characters: (a) printed; (b) hand-written; (c) VOCs; (d) overlapped VOCs.

6. Experiments and results

6.1. Experimental configurations

We implemented gestures-based character input and recognition (GCIR) system to evaluate our feature representation and learning methods. This system was developed by VS2010, OpenCV2.4.3, QT4.8.3 and Kinect SDK 1.6, running on a laptop running Core 2 Duo CPU at 2.00 GHz, 2G RAM, window7 operating system (32digit). The system interface is divided into three parts (see the online part in Fig. 2): (a) candidates are shown dynamically on the top; (b) the input VOC is shown in the middle region; (c) operator's finally choice demonstrated in the bottom region.

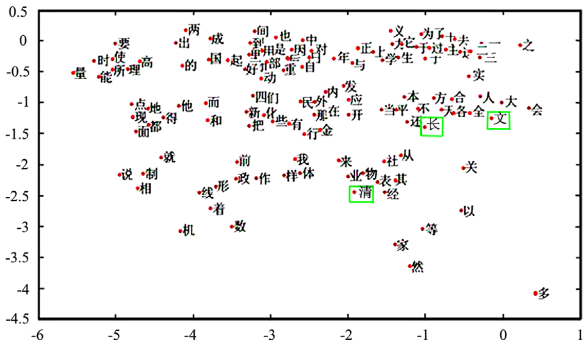

Fig. 9. An example of personalized embedding space by adding three new characters.

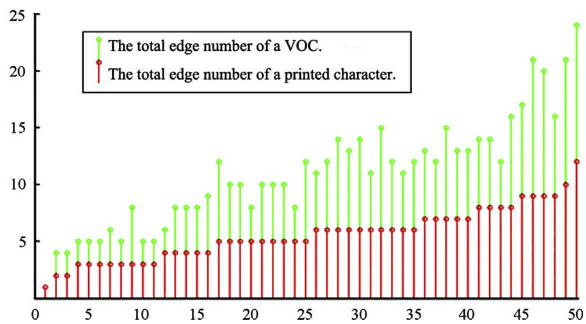

Fig. 10. The comparison of the total number of edges of visual characters and that of their corresponding characters.

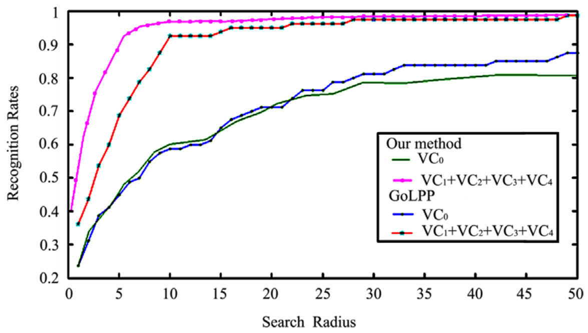

Fig. 11. The comparison of recognition rates between the combined searching on four datasets {VC_1,VC_2,VC_3,VC_4} and searching on a single dataset VC_0.

We define four states of the control hand: (A) start to write a character; (B) writing a character; (C) finish writing a character; and (D) choose a character output in the screen by the recognition algorithm. States (A) and (C) can be confirmed by a pause of more than two seconds for the control hand in the writing region. State (D) can be confirmed by a pause of more than two seconds for the control hand in the candidate characters region. The extraction of the directed edges of a VOC is performed in State (B).

A simple semantic associative database is built to accelerate the speed of input in advance. For every item, the main key indices of characters in the training set, and attributes are the indices of five characters, with which five frequent used two-word phrases can be formed. The users can simultaneously choose the input results and the associative characters in a loop until no associative characters are demonstrated or the user gives a new command.

6.2. Recognition results

The searching radius is critical for recognition accuracy and efficiency as the density of points is not uniform over the embedding spaces. We divided the training data into four subsets {VC_1, VC_2, VC_3, VC_4} according to their daily usage frequencies [28]. This division makes the densities of embedding spaces well-distributed, but also decreases the repetition rates of SDVs in four embedding spaces compared with VC_0 (see Table 2).

In the recognition process, we randomly selected one hundred characters from VC_0 as the testing set and ask five users to write these characters in the standard stroke orders. We project the feature vector of a VOC onto four embedding spaces in parallel by the learnt mapping functions, and sort the neighbors within the searching radius by the similarity metric as the recognition results. Fig. 11 shows the comparison of recognition rates on the four datasets {VC_1, VC_2, VC_3, VC_4} and one larger dataset (VC_0) by using two proposed methods, C-GoLPP and GoLPP. The recognition rate of four datasets combined searching is higher than that in one large dataset for both methods. The system outputs the five most similar candidates at the top region of screen and let users make a final decision. The system can yield a recognition rate as high as 99%.

Our *SDVs* work quite well when characters have the overlapped radicals. We selected fifty complex characters from the dataset VC_1, and generated their *SDVs* by varying the overlapping rate from 10% to 100%. The features for VOCs with self-overlaps below 50% locate quite close to the normal one, and produce a recognition rate above 60% with the Diamond Search in real-time. Besides, we can improve the rate by about 10% with a small amount of human interactions. This experiment validate that SDVs are robust to self-overlap caused by radicals or strokes of characters.

Table 2
The classification of one thousand Chinese characters and the repetition rates of their *SDVs* in five embedding spaces by C-GoLPP and GoLPP.

Data set	Character no.	Frequency of usage (%)	C-GoLPP (%)	GoLPP (%)
VC_0	1–1000	85	1.1	1.5
VC_1	1–140	50	0.1	0.0
VC_2	141–380	20	0.3	0.5
VC_3	381–500	5	0.0	0.0
VC_4	501–1000	10	1.0	1.2

6.3. Analysis

We can see from the above experiments that our subspace learning based on LPP is able to render a simple and fast recognition. The subspace learning provides explicit mapping functions so that we can locate the projected features by an efficient matrix multiplication in the lower dimensional embedding subspace. Our method can also achieve a high recognition rate due to the well-defined learnt structure in the embedding space.

SDVs are established according to the stroke order and may bring inconvenience to those users who cannot write casually. However, this establishment of features follows the oriental culture that well-educated people have to write characters in a certain order. In our GCIR system, exhausted users may give VOCs with more overlapped and distorted radicals, where the image based features are inapplicable. For this case, the SDVs can be assigned a bigger weight, and the system can still perform well.

7. Conclusions

This work focuses on developing a robust, low-cost, stroke order, gesture-based character's interactive input and recognition framework and system for HCI based on segmented directed-edge vectors (*SDVs*) and subspace learning methods. By using Locality Preserving Projections (LPP), several of low-dimensional embedding spaces for large character dataset are learnt respectively according characters using frequencies. Inferring character can be achieved by explicit mappings from a visual input to the embedding spaces. Extensive experimental results demonstrate qualitatively and quantitatively that satisfactory recognition of visual oriental characters can be achieved by our method robustly and efficiently. Our proposed GCIR system can be used as an interface tool for computers and an instruction tool for robots.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China under Grant nos. 61402072, 61033012, 11171052, 61272371, 61003177 and 61328206, the Specialized Research Fund for the Doctoral Program of Higher Education of China (20120041120050), and the program for New Century Excellent Talents (NCET-11-0048).

References

[1] G.R.S. Murthy, R.S. Jadon, A review of vision based hand gestures recognition, Int. J. Inf. Technol. Knowl. Manag. 2 (2) (2009) 405–410.

[2] J. Yu, Y. Rui, Y.Y. Tang, et al., High-order distance-based multiview stochastic learning in image classification, IEEE Trans. Cybern. PP (99) (2014) 1–8.

[3] S. Carbini, L. Delphin-Poulat, L. Perron, et al., From a wizard of Oz experiment to a real time speech and gesture multimodal interface, Signal Process. 86 (12) (2006) 3559–3577.

[4] C.L. Liu, F. Yin, D.H. Wang, et al., Online and offline handwritten Chinese character recognition: benchmarking on new databases, Pattern Recognit. 46 (1) (2013) 155–162.

[5] M.M. Hasan, P.K. Mshral, Hand gesture modeling and recognition using geometric features: a review, Can. J. Image Process. Comput. Vis. 3 (1) (2012).

[6] M.T. Parvez, S.A. Mahmoud, Offline arabic handwritten text recognition: a survey, ACM Comput. Surveys 45 (2) (2013) 23.

[7] C.L. Liu, H. Fujisawa, Classification and learning for character recognition: comparison of methods and remaining problems, in: Proceedings of the International Workshop on Neural Networks and Learning in Document Analysis and Recognition, 2005.

[8] X. Zhang, Z. Ye, L. Jin, et al., A New Writing Experience: finger Writing in the Air Using a Kinect Sensor, IEEE Multimed. 20 (4) (2013) 85–93.

[9] A. Bharath, S. Madhvanath, HMM-based lexicon-driven and lexicon-free word recognition for online handwritten indic scripts, IEEE Trans. Pattern Anal. Mach. Intell. 34 (4) (2012) 670–682.

[10] H. Chung, H.D. Yang, Conditional random field-based gesture recognition with depth information, Opt. Eng. 52 (1) (2013)017201, http://dxdoi.org/10.1117/1.OE.52.1.017201.

[11] X.F. He, S.C. Yan, Y.X. Hu, and H.J. Zhang, Learning a locality preserving subspace for visual recognition, in: Proceedings of the IEEE International Conference on Computer Vision, 1, 2003, pp. 385–392.

[12] L. Zhang, L. Qiao, S. Chen, Graph-optimized locality preserving projections, Pattern Recognit. 43 (6) (2010) 1993–2002.

[13] J. Yu, D. Liu, D. Tao, H.S. Seah, Complex object correspondence construction in 2D animation, IEEE Trans. Image Process. 20 (11) (2011) 3257–3269.

[14] Z.Y. Zhang, J. Wang, H.Y. Zha, Adaptive manifold learning, IEEE Trans. Pattern Anal. Mach. Intell. 34 (2) (2012) 253–265.

[15] D. Tao, X. Li, X. Wu, et al., Geometric mean for subspace selection, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2) (2009) 260–274.

[16] E. Marie Mugavin, Multidimensional scaling: a brief overview, Nurs. Res. 57 (1) (2008) 64–68.

[17] A.M. Martinez, A.C. Kak, PCA versus LDA, IEEE Trans. Pattern Anal. Mach. Intell. 23 (2) (2001) 228–233.

[18] J.B. Tenenbaum, V. DE Silva, J.C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (5500) (2000) 2319–2323.

[19] S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (2000) 2323–2326.

[20] M. Belkin, P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, Neural Comput. 15 (6) (2003) 1373–1396.

[21] T. Lin, H. Zha, S.U. Lee, Riemannian Manifold Learning for Nonlinear Dimensionality Reduction, Computer Vision – ECCV 2006, Springer, Berlin Heidelberg, 2006, 44–55.

[22] J. Wang, Local Tangent Space Alignment, Geometric Structure of High-Dimensional Data and Dimensionality Reduction, Springer, Berlin Heidelberg, 2011, 221–234.

[23] S. Yen, C. Wu, H. Wang, A. Block-Based, Orthogonal locality preserving projection method for face super-resolution, intelligent information and database systems, Lect. Notes Comput Sci. 7197 (2012) 253–262.

[24] W.K. Wong, H.T. Zhao, Supervised optimal locality preserving projection, Pattern Recognit. 45 (1) (2012) 186–197.

[25] J. Yu, D. Tao, M. Wang, Adaptive hypergraph learning and its application in image classification, IEEE Trans. Image Process. 21 (7) (2012) 3262–3272.

[26] J. Yu, R. Hong, M. Wang, et al., Image clustering based on sparse patch alignment framework, Pattern Recognit. 47 (11) (2014) 3512–3519.

[27] X.Q. Lü, A segment abstraction-integrate algorithm based on polygon approximation and finite state machines, J. Softw. 19 (2008) 52–58.

[28] P.R. China, Ministry of Education and the State Language Work Committee, the Language Specification: Modern Commonly Used Single Characters Norms, Language Press, Beijing, P.R. China, 2009.

[29] H. Izakian, A. Abraham, Fuzzy C-means and fuzzy swarm for fuzzy clustering problem, Expert Syst. Appl. 38 (3) (2011) 1835–1838.

[30] Z. Zhang, L. Jin, K. Ding, et al., Character-SIFT: a novel feature for offline handwritten Chinese character recognition, in: Proceedings of the ICDAR'09 on Document Analysis and Recognition, 2009, pp. 763–767.

[31] D. Tao, L. Liang, L. Jin, et al., Similar handwritten Chinese character recognition using discriminative locality alignment manifold learning, in: Proceedings of the IEEE International Conference on Document Analysis and Recognition (ICDAR), 2011, pp. 1012–1016.

[32] Miao Cheng, et al., Incremental embedding and learning in the local discriminant subspace with application to face recognition, IEEE Trans. Syst. Man Cybern Part C 40 (5) (2010) 580–591.

[33] H.C. Martin, K. Law Anil K, Incremental nonlinear dimensionality reduction by manifold learning, IEEE Trans. Pattern Anal. Mach. Intell. 28 (3) (2006).