Offline Signature Verification Using Online Handwriting Registration

Yu Qiao, Jianzhuang Liu
Department of Information Engineering
The Chinese University of Hong Kong
qiao@gavo.t.u-tokyo.ac.jp, jzliu@ie.cuhk.edu.hk

Xiaoou Tang
Microsoft Research Asia
Beijing, China
xitang@microsoft.com

Abstract

This paper proposes a novel framework for offline signature verification. Different from previous methods, our approach makes use of online handwriting instead of handwritten images for registration. The online registrations enable robust recovery of the writing trajectory from an input offline signature and thus allow effective shape matching between registration and verification signatures. In addition, we propose several new techniques to improve the performance of the new signature verification system: 1. we formulate and solve the recovery of writing trajectory within the framework of Conditional Random Fields; 2. we propose a new shape descriptor, online context, for aligning signatures; 3. we develop a verification criterion which combines the duration and amplitude variances of handwriting. Experiments on a benchmark database show that the proposed method significantly outperforms the well-known offline signature verification methods and achieve comparable performance with online signature verification methods.

1. Introduction

Signature is a socially accepted authentication method and is widely used as proof of identity in our daily life. Automatic signature verification by computers has received extensive research interests in the field of pattern recognition. Depending on the format of input information, automatic signature verification can be classified into two categories: online signature verification [6, 13, 8, 12, 5] and offline signature verification [7, 23, 15]. In the former case, a hand pad together with an instructed pen [6, 13, 8, 5] or a video camera [12] is used to obtain the online information of pen tip (position, speed, and pressure). Therefore the input is a sequence of features. In the latter case, the input is a two-dimensional signature image captured by a scanner or other imaging device. Online signature verification has been shown to achieve much higher verification rate than offline verification [6, 13, 8, 12, 5, 7, 23, 15]. The state of the art of online verification achieves equal error rates (EERs) ranging from 2% to 5% [6, 13, 8, 12, 5], while the EERs of offline verification are still as high as 10%-30% [12, 7, 23]. This difference is largely due to the availability of dynamic information in online system [5, 15]. Roughly speaking, the matching and annotation problems for 2D images are more difficult and time consuming than those for 1D sequences. Although online verification outperforms the offline one, its use of special devices for recording the pen-tip trajectory increases its system cost and brings constraints on its applications. In some situations, such as check transaction and document verification, offline signature is obligatory. This paper focuses on offline signature verification, and our objective is to discriminate between genuine signatures and skilled forgeries which are written by careful imitation.

Various features have been proposed for signature verification tasks. These features can be roughly divided into two types [13, 5]: 1) global features which are extracted from the whole signature, including block codes [7, 23], Wavelet and Fourier series [13], etc.; 2) local features which are calculated to describe the geometrical and topological characteristics of local segments, such as position, tangent direction, and curvature [13, 8, 5, 12]. The global features can be extracted easily and are robust to noise. But they only deliver limited information for signature verification [13, 8]. On the other hand, local features provide rich descriptions of writing shapes and are powerful for discriminating writers, but the extraction of reliable local features is still a hard problem.

The local features based approaches are more popular in online verification than in the offline one. This is because it is much easier to calculate local shape features and to find their corresponding relations in 1D sequences than in 2D images. This fact inspires us to consider recovering writing trajectories from offline signature images. Then local features can be calculated and aligned more efficiently and effectively by using recovered trajectories. Similar idea had been adopted by Lee and Pan [15], where they proposed local tracing algorithms to find the dynamic information of signatures. However, as pointed out in [14, 22], the local

1-4244-1180-7/07/$25.00 ©2007 IEEE

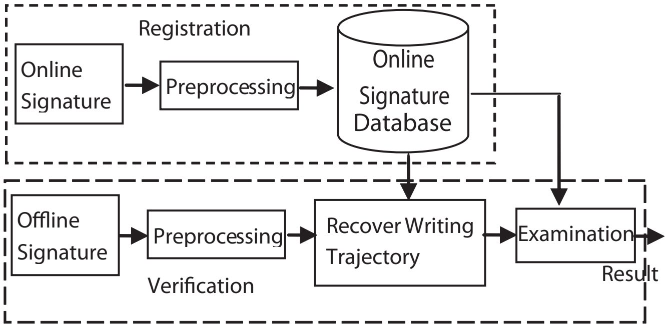
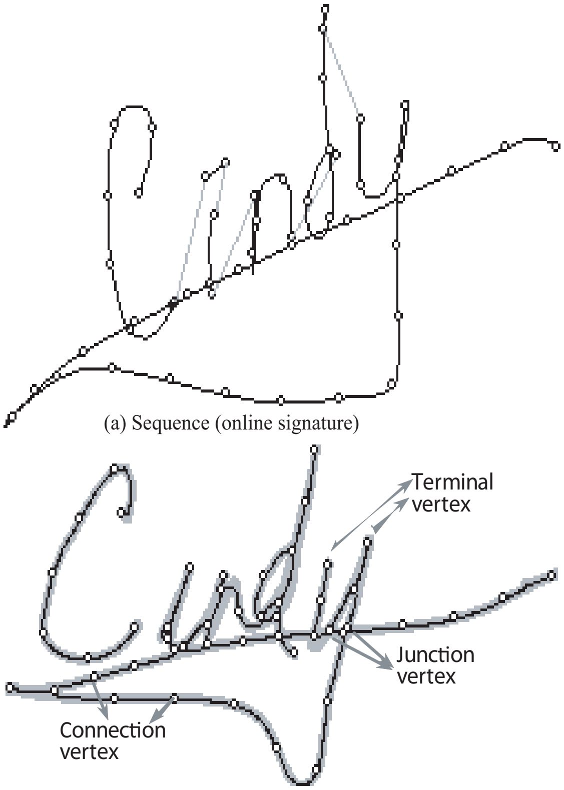


Figure 1. System diagram.

tracing methods are sensitive to noise and writing variance and it is difficult to design tracing algorithms which can be applied to variant writing styles. In fact, direct recovery of writing trajectory is still an open problem in handwriting research [18, 14, 22]. To circumvent this difficulty, we propose a novel approach that uses online signatures in the registration phase. We develop this approach based on the observation that registration needs to be done only once and it has to be done in person in-situ (such as in a bank) where an online device is easily available. In the verification phase, the procedure is exactly the same as an offline system thus is convenient to use. In our algorithm, we take advantage of the online registration data to recover writing trajectory of an offline input signature image and make the verification decision based on the recovered trajectory. The system diagram is shown in Fig.1.

In the context of signature verification, tested signatures whether genuine or skillfully forged, usually have similar shapes with the registration ones. (If a tested signature has a shape very different from the registration, it can be easily identified as a forgery.) This similarity allows us to recover the writing trajectory by using online signatures as examples. In this paper, we model signatures using Conditional Random Fields (CRFs) and reduce the recovery of writing trajectory to a CRFs inference problem.

Different from online signature verification, our problem is based on the recovered trajectories which do not have dynamic writing information such as speed, pressure, and orientation. To compensate the loss of the dynamic information, we introduce a new descriptor, online context, which summarizes shape information by vector histograms and is robust to local deformation. We develop a new verification criterion that combines two types of variances in handwriting duration and amplitude. Experimental results show that our method achieves EERs which are lower than those of the previous offline verification methods and are comparable to the online methods.



(a) Sequence (online signature)

(b) Graph (Offline sigmature)

Figure 2. Examples of sequence and graph. Circles represent points/vertices and black lines represent edges. Gray lines in (a) correspond to pen movements out of paper. Gray area in (b) represent original image.

2. Recovery of writing trajectories

The recovery of a writing trajectory from an offline signature image is to find a trajectory within the offline image which is most similar to the trajectory of the online registrations. There are two basic problems, how to evaluate the similarity and how to find the best path? This paper answers these two questions within the framework of conditional random fields. The similarity is defined through the state and transition functions of CRFs, and the problem to find the best trajectory is reduced to the inference problem of CRFs.

2.1. Representation of online/offline signatures

The online signature consists of a set of consecutive strokes, denoted by $S = s_1, s_2, ..., s_m$ where m is the number of strokes. We sample points with equal space along each stroke. Then we connect all the strokes to get a single sequence of points denoted by $L = p_1, p_2, ..., p_n$ (Fig.2a). For each two consecutive points p_i and p_{i+1} in L, we add a directed edge e_i: $p_i \rightarrow p_{i+1}$ between them.

The offline signature image is modeled by graph $G =$

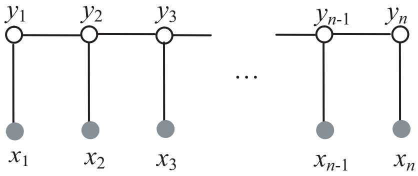


Figure 3. Chain structured conditional random field.

(E, V) (Fig. 2b) where $V = \{v_i\}$ denotes a set of vertices and $E = \{e_k\}$ denotes a set of edges. G is constructed from the skeleton of input image. Each vertex v_i corresponds to a point sampled from the skeleton. There are three kinds of vertex: *terminal vertex* which is a pixel with degree one, *junction vertex* which is a pixel or a cluster of connected pixels with degree three or more, and *connection vertex*, which is a two-degree pixel sampled along a segment of skeleton between two terminal or junction vertices. If there exists a segment of skeleton between v_i and v_j, two directed edges e_{i1}: $v_i \to v_j$ and e_{i2}: $v_j \to v_i$ are added into E. For edge e, we define its continuous set $N(e)$ as a set of edges which can be traced continuously next to e. $N(e)$ includes the edges whose start vertex is the end vertex of e and also the edges which may start a new stroke after e.

For each directed edge e, we calculate its three features (θ, a, b), where θ is its directed angle, a and b represent its normalized positions. These features will be used to evaluate the similarity cost between online and offline signatures in the next.

Given online registration signature L and offline signature G, we want to find the corresponding relations between the elements in L and the elements in G. As the writing order of the edges in L is known, we can trace the edges in G according to the orders of their corresponding edges in L, thus recover the writing trajectory. Next, we will formulate the problem within the framework of conditional random fields.

2.2. Conditional random fields

Conditional Random Fields (CRFs) are non-generative graphical models first proposed by Lafferty et al. in [9]. CRFs define a conditional probability distribution of label set $Y = \{y_i\}_{i=1}^n$ given observed feature set $X = \{x_i\}_{i=1}^n$. We use a first order chain structured CRF as shown in Fig.3.

Compared with HMM, CRFs relax the independent assumption of states by capturing dependencies among observations. CRFs can avoid the label bias problem [9] exhibited by HMMs and MEMMs. CRFs need a small number of training samples and allow flexible features, as it need not to specify a complete distribution for explaining observations. These facts make it suitable for our task.

Conditioned on X, random variable y_i obeys the Markov property i.e., $p(y_i|X, y_j(j \neq i)) = p(y_i|X, y_{i-1}, y_{i+1})$. According to the Hammersley-Clifford theorem [4], the conditional probability can be written into the following form [9]:

$$p_\Lambda(Y|X) = \frac{1}{Z(X)} \exp(\sum_{i,k} \lambda_k f_k(y_{i-1}, y_i, X, i) + \sum_{i,k} \mu_k g_k(y_i, X, i)), \quad (1)$$

where $\Lambda = \{\lambda_k, \mu_k\}$ represents the set of parameters and $Z(X)$ is a normalized factor. $f_k(y_i, y_{i-1}, X, i)$ is a transition function [1] at positions $i-1$ and i; $g_k(y_i, X, i)$ is a state function at position i.

A straightforward approach is to assume that CRF has the same topology as G and to use the vertices of G as observations and the indices of the points in L as labels. However, this approach has several disadvantages: 1) G may include double traced lines, so one observation may correspond to more than one state; 2) G may have complex topology which will increase the computational complexity; 3) it is difficult to account for between-edge relations if we use vertices/points as states, since a function of Eq.(1) includes at most two states.

Due to these considerations, we use a chain structured CRF (Fig. 3), where the online signature is regarded as its observation, each observation x_i represents an edge in L and label y corresponds to the index of directed edge e_y in G.

In this way, functions g_k represent the directional and positional differences which are used to punish the shape difference between X and Y. Functions f_k represent the smoothness which are used to evaluate the continuity of edges, thus to make the recovered writing path smooth.

2.3. Parameters estimation for CRFs

The online registration signatures are used to train the CRFs. We calculate the matching cost for each two signatures by using online context based dynamic time warping (DTW). Then the set of training signatures is divided into one or more clusters, where the pairwise matching cost in each cluster must be less than a threshold. Let $S_1, S_2, ..., S_M$ denote a cluster. In the next, we will train a CRF for each cluster. Define the mean signature S_m as the one which has the minimum DTW matching cost with the remaining signatures. We divide S_m into n edges with equal length and then divide other signatures according to their aliments to S_m. In this way, all the signatures are divided into n edges. For each edge, we calculate its three features.

[1] f_k and g_k are called ”features” in [9], however, the term ”feature” has a meaning of the properties of edges in this paper. To avoid confusion, we use term ”function”.

Let X denote the observed feature sequence of S_m, and $Y^{(j)}$ denote the indexes of the edges in $S_j (j \neq m)$. (X, $Y^{(j)}$) represent a training pair for CRFs. Totally, the number of training samples is $N = M - 1$. Note that observation X is fixed during the training phase.

Given training samples $D = \{(X, Y^{(j)})\}_{j=1}^{N}$, we need to estimate the parameters $\Lambda = \{\lambda_k, \mu_k\}$. Since state function g_k can be seen as a special form of f_k, henceforth we will omit the discussion on g_k and μ_k for simplicity. The learning of CRFs aims to find the parameters which maximize the conditional log-likelihood:

$$L(\Lambda) = \sum_{Y} \log p_{\Lambda}(Y|X). \tag{2}$$

To reduce overfitting, we use a Gaussian prior $p(\Lambda)$ with a diagonal covariance matrix for regulation. Similar prior is adopted in Bayesian-CRF [19]. Thus Eq. (2) becomes

$$L(\Lambda) = \sum_{Y} \log p_{\Lambda}(Y|X) - \sum_{k} \frac{\lambda_k^2}{2\sigma_k^2} + \text{const}, \tag{3}$$

where σ_k^2 is the variance of λ_k, which can be approximated by the reciprocal of the variance of f_k in experiments.

It can be shown that the above likelihood function is convex on Λ, which guarantees the convergence to a global optimization [12]. The Generalized Iterative Scaling (GIS) [9, 2] algorithm has been adopted to calculate the optimal parameters. GIS algorithm iteratively increases $L(\Lambda)$ by updating the parameters as: $\lambda_k \leftarrow \lambda_k + \delta_k$. Due to the use of Gaussian prior, our training algorithm is different from that in [9]. The details are as the following:

1. Initialize the parameters $\Lambda^{(0)}$.
   For each function f_k, calculate its mean:

   $$\bar{f}_k = \frac{1}{N} \sum_{k} \sum_{i} f_k(y_{i-1}, y_i, X, i). \tag{4}$$

   For each training pair (X, Y), calculate:

   $$T(X, Y) = \frac{1}{N} \sum_{Y} \sum_{i} f_k(y_{i-1}, y_i, X, i). \tag{5}$$

2. Repeat the following steps until convergence. In the t-th step (t=1,2,...),
   2.1 Calculate $\delta_k^{(t)}$ by solving the equations:

   $$\bar{f}_k = \frac{\lambda_k^{(t)} + \delta_k^{(t)}}{\sigma_k^2} + \sum_{Y} p_{\Lambda^{(t)}}(Y|X) \sum_{i} f_k(y_{i-1}, y_i, X, i) \exp\{\delta_k^{(t)} T(X, Y)\}. \tag{6}$$

   2.2 Update $\Lambda^{(t+1)}$ by $\lambda_k^{(t+1)} = \lambda_k^{(t)} + \delta_k^{(t+1)}$.

It can be shown that the roots $\{\delta_k^{(t)}\}$ of Eq. (6) always lead to the increase of the likelihood. The derivations of this equation can be found in [20].

2.4. Inference by dynamic programming

The recovery of writing order from an offline signature image is formulated as the inference problem of CRFs. For an input signature image, we build its graph model G and calculate the features of its edges. Feature sequence X used in the training is also used as the observation during inference. Y denotes a set of indices of the edges in G. By solving the inference problem of the CRF, we can determine the corresponding relations between the edges in X and the edges in G, that is, to determine the writing order of the edges in G. Formally, the inference problem is denoted by

$$Y* = \arg\max_{Y} p_{\Lambda}(Y|X), \tag{7}$$

subjected to

$$e_{y_i} \in \{N(e_{y_{i-1}}) \cup e_{y_{i-1}}\}. \tag{8}$$

The above constraint is used to ensure that the edges can be traced continuously along the recovered order. As $Z(X)$ is a constant, Eq. (7) is equal to

$$\max_{y_1, y_2, \ldots, y_n} \{\sum_{i,k} \lambda_k f_k(y_{i-1}, y_i, X, i) + \sum_{i,k} \mu_k g_k(y_i, X, i)\}. \tag{9}$$

The above form reminds us that dynamic programming (or Viterbi decoding) can be used to optimize it. Define the following function $F(y, i)$:
If $i = 0$, $F(y, 0) = \sum_{k} \mu_k g_k(y, X, 0)$;
else if $i > 0$,

$$F(y, i) = \max_{y'} \{F(y', i-1) + \sum_{k} \lambda_k f_k(y', y, X, i)\} + \sum_{k} \mu_k g_k(y, X, i). \tag{10}$$

We can iteratively update $F(y, i)$ using Eq. (10). Finally, $F(y, N)$ will be the solution for Eq. (10). Let $Y = y_1, y_2, ..., y_n$ denote the optimal configuration of Eq. (10). The recovered writing trajectory is $e_{y_1} \rightarrow e_{y_2}, ..., \rightarrow e_{y_n}$. To speed up the process, we identify the candidate start and end points for each stroke at first and then do bidirectional search from these candidates [21]. The ECR (Edge Continuity Relation) analysis method proposed in [22] is used to simplify graph model G.

A problem of the above method is that some edges of G may not be traced. To overcome this, we use a refining algorithm to insert the un-traced edges into the recovered

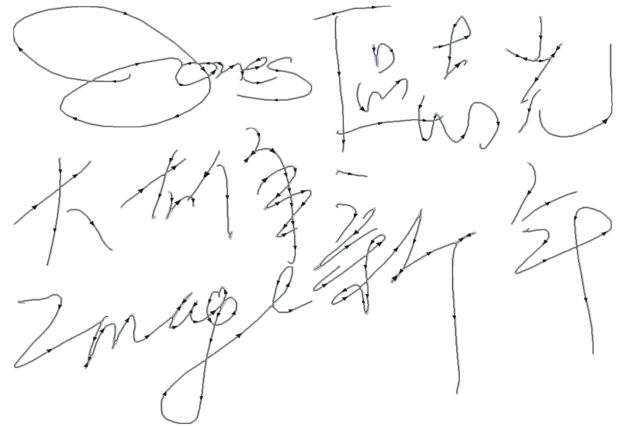


Figure 4. Examples of recovery results. Writing trajectories are shown as the arrows.

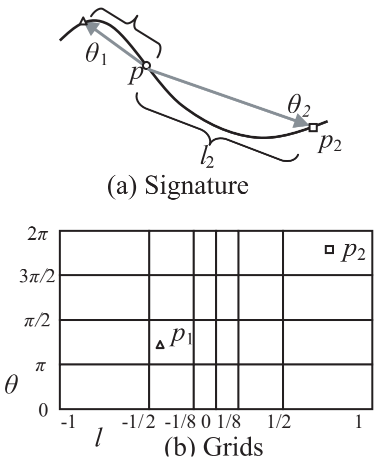
trajectories. The algorithm works in a greedy fashion. In each step, it selects the edge which can be inserted in the smoothest way. If too many edges are still not traced after refinement, the signature must be very different from the online registration and it should be recognized as a forgery. Since there are more than one CRF models for a subject, we use each of them to recover the writing trajectories and the final result is selected as the one with the largest likelihood. Some examples of the recovery are shown in Fig. 4. It can be seen that the proposed method can effectively deal with double traced edges and junctions of multiple strokes, which are regarded as the hard problems in recovery [22]. In our experiments, most failed recovered signatures are forgeries.

3. Signature verification

In this section, we compare the recovered trajectory with the online registrations to reach a verification decision. It seems that the conditional probability defined in Eq.(1) can be used as a criterion for verification. However, Eq. (1) only measures the conditional distribution given one signature example and it doesn't describe the signature distribution of a writer. Moreover, it is just a coarse description and cannot account for the detailed shape information of writing trajectory, which is important for signature verification.

Handwriting is a complex biomechanical process, which includes the movements of fingers, wrist, and forearm. It has been shown that humans generate handwriting through controlling the magnitude and direction of speed [17]. And handwriting exhibits variances in both duration (time) and amplitude (space).

Although the writing trajectory is recovered, our problem is still different from the online signature verification, since the recovered trajectory doesn't have the dynamic information such as speed and pressure. Moreover, the recovered trajectory is within the skeleton calculated by the thinning process. It is well known that thinning algorithms are sensitive to noise and may result in unwanted artifact lines [18, 22]. Thus the recovered trajectory does not always superpose the original writing trajectory. To reduce the unfavorable effects of these problems, we introduce online context based dynamic time warping for aligning the trajectories. And we develop a verification criterion which combines the time and amplitude variances.



(a) Signature

(b) Grids

Figure 5. Online Context.

3.1. Online context

For each point p along the signature, we define the online context to describe its relative shape information along the trajectory. We draw vectors from p to all the other sampling points along the trajectory (Fig. 5a). Each vector $p \rightarrow p_1$ is represented by two parameters (θ, l), where θ is the directional angle, and l is the normalized length of the trajectory between p and p_1 (the term "normalized" means the length is divided by the total length of the signature). These vectors provide rich descriptions of the detailed shape. However, because of the large number of vectors, it is not easy to deal with them directly. Thus, we use a histogram to summarize the distribution of these vectors on the two parameters of θ and l. Specially we introduce the grids whose vertical axis corresponds to θ and whose horizontal axis corresponds to the logarithm [2] of l (Fig. 5b). Then we count the number of vectors falling into each grid. Formally, the online context h_k is defined as

$$h_p(k) = \#\{p_i \neq p : p \rightarrow p_i \in \text{Grid}(k)\}. \qquad (11)$$

Online context is related to the shape context proposed in [1]. Both compute the distribution of the vectors from one point to others for describing shapes. However, we use curve length other than Euclidean distance to construct

[2] The use of logarithm is to make online context more sensitive to the nearby points than to the faraway ones.

grids. Moreover, the corresponding problem is solved by the dynamic time warping, other than the bipartite graph matching in [1]. In our experiments, online context outperforms shape context for signature verification tasks.

For point p_i in one trajectory and point q_j in another trajectory, we use $d(p_i, q_j)$ or simply $d(i, j)$ to denote the matching cost between p_i and q_j. Using χ^2 statistics, we have

$$d(i,j) = \frac{1}{2} \sum_k \frac{(h_{p_i}(k) - h_{q_i}(k))^2}{h_{p_i}(k) + h_{q_i}(k)}. \tag{12}$$

It is easy to see that the online context and the above distance are invariant to translation and scale. Besides, online context has practical significance for our verification task. Skeleton is sensitive to noise and may be deformed in thinning procedure. So it is hard to calculate reliable local features from skeleton directly [18, 22]. The online contexts are global and rich descriptors which are robust to the local perturbations caused by noise and deformation. Besides,researches had shown that fast handwriting, such as signature, is a ballistic movement [16, 18], where the writing shape is formed in a global fashion in human brain. The online contexts provide global description of shapes, which should be closer to the brain representation of handwriting than local features such as directional angles and positions. We believe the online contexts are also effective features for handwriting recognition tasks which is different from verification. But the exploration of this is out of the range of this paper.

3.2. Alignment by dynamic time warping

One of the key problems for comparing two signatures is to find the correspondent relations between their points. *Dynamic Time Warping* (DTW), also called, dynamic programming matching [24], is widely used to align sequences. DTW searches the optimal correspondence (named warping path) among the elements of two sequences by minimizing the accumulated distance. DTW has shown its effectiveness in both online and offline signature verification [6, 13, 12, 3].

Consider two signatures: $L = (p_1, p_2,..., p_n)$ and $L' = (q_1, q_2,..., q_m)$. An alignment between L and L' is represented by warping path , where T is the length of warping path, and two warping functions: $w_p(i) : [1, T] \rightarrow [1, n]$ and $w_q(i) : [1, T] \rightarrow [1, m]$, mean that the $w_p(i)$-th point in L corresponds to the $w_q(i)$-th element in L'. The matching cost between L and L' along w is defined as:

$$D_w(L, L') = \frac{1}{T} \sum_{i=1}^{T} d(w_p(i), w_q(i)). \tag{13}$$

In our approach, $d(w_p(i), w_q(i))$ is calculated by using online context. The optimal warping path is the one which minimizes the above cost function,

$$w* = \arg\min_w \{D_w(L, L')\}. \tag{14}$$

The optimization problem of DTW can be solved efficiently using dynamic programming [24].

3.3. Verification criteria

We use a nearest neighbor classifier for verification. This is mainly due to the limited number of registrations and the large variance among registrations in signature verification task. To do so, we need a criterion (distance) between an examined signature and a registered signature. A good criterion should be sensitive to the between-writer variance while be robust to the within-writer variance.

It has been shown that human handwriting can be divided into basic segments [8, 16]. The variances of handwriting shape come from the changes of the duration and amplitude of each segment. In a Sense, DTW plays a role to counteract duration variance by warping the signature trajectories. Therefore the matching cost of Eq. (13) provides a good measure of the amplitude variance. But it does not include much information about duration. The duration information is encoded in warping path w. However, it is not a good idea to model w directly. This is because w includes a large number of elements and the length of w is variable. We model the duration in a more direct way by using the trajectory length of each segment. For online signature, we divide the trajectory into short segments by breaking at the points whose speed is near zero. Similar technique had been used in [8]. For offline signature, we can align it to a prototype online signature by DTW and then divide it at the points which are aligned to the breaking points of the online one. Thus for each signature, we can divide it into K segments. Let $z = [l_1, l_2,..., l_K]^T$ denote the lengths of the segments.

Next, we estimate the distribution $p(z)$ from a set of training length vectors $z_1, z_2, ..., z_N$ calculated from the online registration signatures. By assuming z obey the Gaussian distribution, we can calculate $p(z)$ as:

$$p(z) = \exp\{-0.5(z - \bar{z})^T \Sigma^{-1} (z - \bar{z})\} / \{(2\pi)^{K/2} |\Sigma|^{0.5}\}, \tag{15}$$

where $\bar{z}$ is the mean and Σ is the covariance matrix of z.

However K can be larger than the number of training vectors N. So Σ may be singular and does not have an inverse. We adopt a PCA based method proposed by Moghaddam and Pentland [11] to approximate $p(z)$.

Let $\Phi_J = [h_1, h_2, ..., h_J]$ where h_1, h_2,..., h_J are the eigenvectors of Σ associated with the largest J eigenvalues $\zeta_1, \zeta_2, ..., \zeta_J$ ($J \ll K$). For vector z, PCA projects it into a low dimensional space by

$$t = \Phi_J^T (z - \bar{z}), \tag{16}$$

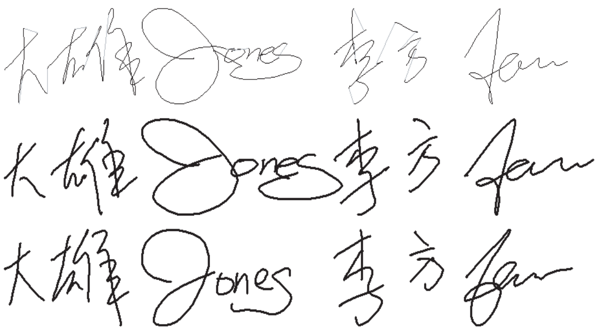


Figure 6. Examples of online signatures and converted images. The first row shows online signatures. Genuine signature images and forgery images are given in the second and the third rows.

where $t = [t_1, t_2, ..., t_J]^T$ is a vector of length J.

By discarding the information on the eigenvectors with small eigenvalues, we can approximate $p(z)$ by

$$p(z) \cong p(t) = \frac{\exp\{-0.5\sum_{j=1}^{J}\frac{t_j^2}{\zeta_j}\}}{(2\pi)^{J/2}\prod_{j=1}^{J}\zeta_j^{1/2}}. \qquad (17)$$

For an online registration L' and a verification signature L, we use $D_{w*}(L, L') - \rho \ln p(z)$ as a criterion for signature verification, where z is a length vector for aligning L to L' and ρ is a weight coefficient.

4. Experiments

We evaluate the performance of our method by using the online signature database for the Signature Verification Competition (SVC) 2004 [26]. This database includes two datasets and each dataset consists of the signatures for 40 subjects[3]. Each subject contributed 20 genuine signatures, and another 20 skilled imitations were written by at least four forgers. So totally there are 3200 signatures. For each subject, we use 10 online genuine signatures for registrations. The remaining data (10 genuine signatures and 20 forgeries for each subject) are converted into the offline images by connecting the points along the writing trajectory with lines. The SVC database includes both English and Chinese signatures. Some examples of online signatures and converted images are shown in Fig. 6. The online registrations are used to training the CRFs for each subject.

Table 1. Comparisons of EERs with offline methods (%).

Data	Project	GSC	ESC	Ours
Set1	25.3	23.3	26.5	**7.3**
Set2	25.1	22.0	27.9	**7.4**

Table 2. Comparisons of EERs with online methods (%).

Data	Position	Angle	SC	OC	Ours
Set1	13.6	6.5	7.2	5.8	**7.3**
Set2	11.9	6.3	4.9	4.6	**7.4**

We implement three offline signature verification methods proposed in [3, 7, 23] for comparisons. In [3], Fang et al. aligned the projection profile of signature images by dynamic time warping and used Mahananobis distance for verification. Kalera et al. [7] extracted binary Gradient Structural and Concavity (GSC) features from signature images and verified signatures by a correlation measure. Sabourin and Genest [23] calculated Extended Shadow Code (ESC) to describe the shapes and used Euclidean distance for verification. We call the three methods as Project, GSC and ESC methods. In our implementation of the Projection method, we select vertical projection for its good performance in Fang's experiments [3]. The GSC and ESC methods calculate pairwise distance between signatures. To be fair, we use nearest neighbor classifiers in their implementation.

Signature verification is a two-class classification problem. We calculate False Rejection Rate (FRR), False Acceptance Rate (FAR) and Equal Error Rate (EER) for evaluation. FRR measures the rates of genuine signatures classified as forgeries, while FAR represents the rates of forgeries recognized as genuine ones. EER is the value when FAR equals to FRR. The results are given in Table 1 and Fig. 8. It can be seen that the proposed method outperforms significantly the three compared methods on both datasets.

We also execute experiments to compare our method with the online signature verification methods. Four online methods are implemented. All of them use DTW for alignment but differ in the ways of calculating the matching cost between points. Specially, the cost is calculated by using normalized position, directional angles [6, 13], shape context (SC) [1] and online context (OC) proposed in Section 3. Table 2 and Fig. 8 show the EERs and ROCs. The results of the proposed method are comparable with online signature verification methods. It can also be seen that online context achieves better performance than the other methods in terms of the EERs.

5. Conclusions

This paper presents a novel method for offline signature verification, which makes use of online handwriting for registrations. We introduce conditional random fields for efficiently matching an offline signature image with online signatures, which leads to the recovery of the writing trajectory from the signature image. We propose online context as detailed descriptors for writing trajectories and use dynamic

[3] In the competition, each dataset includes 100 subjects. But only the data of the first 40 are open to public.

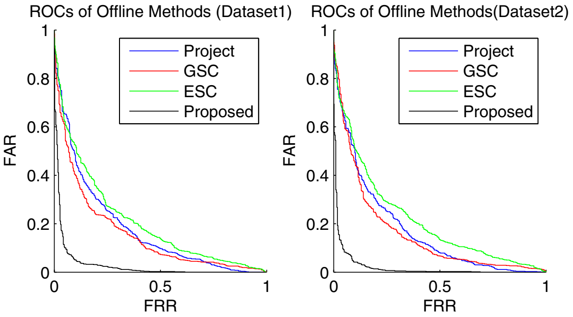


Figure 7. Comparison of average receiving operating curves with offline signature verification methods.

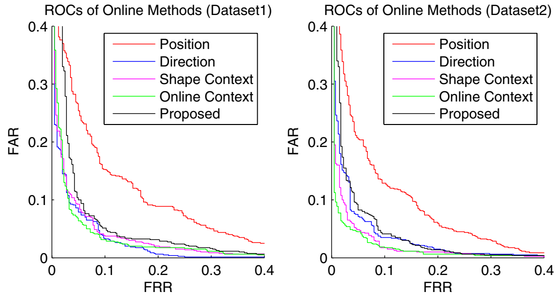


Figure 8. Comparison of average receiving operating curves with online signature verification methods.

time warping to align trajectories. We develop a new verification criterion which combines the duration and amplitude variances of handwriting. Experimental results show that our method achieves an average EER of 7.4% on the SVC database, which is significantly lower than those of the three compared offline methods. Future work includes the improvement of the precision of the recovery and the experiments on larger database with real images collected through the video-based system we recently developed [10, 25].

References

[1] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. PAMI*, 24(4):509–522, 2002.

[2] J. Darroch and D. Ratcliff. Generalized Iterative Scaling for Log-Linear Models. *The Annals of Mathematical Statistics*, 43(5):1470–1480, 1972.

[3] B. Fang, C. Leung, Y. Tang, K. Tse, P. Kwok, and Y. Wong. Off-line signature verification by the tracking of feature and stroke positions. *Pattern Recognition*, 36(1):91–101, 2003.

[4] J. Hammersley and P. Clifford. Markov fields on finite graphs and lattices. *Unpublished manuscript*, 1971.

[5] T. Hastie, E. Kishon, M. Clark, and J. Fan. A model for signature verification. *Proc. SMC*, pages 191–196, 1991.

[6] A. Jain, F. Griess, and S. Connell. On-line signature verification. *Pattern Recognition*, 35(12):2963–2972, 2002.

[7] M. Kalera, S. Srihari, and A. Xu. Offline signature verification and identification using distance statistics, 2004.

[8] R. Kashi, J. Hu, W. Nelson, and W. Turin. A Hidden Markov Model approach to online handwritten signature verification. *IJDAR*, 1(2):102–109, 1998.

[9] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *Proc. 18th ICML*, pages 282–289, 2001.

[10] F. Lin and X. Tang. Dynamic stroke information analysis for video-based handwritten Chinese character recognition. *Proc. ICCV*, pages 695–700, 2003.

[11] B. Moghaddam and A. Pentland. Probabilistic visual learning for object representation. *IEEE Trans. PAMI*, 19(7):696–710, 1997.

[12] M. Munich and P. Perona. Visual identification by signature tracking. *IEEE Trans. PAMI*, 25(2):200–217, 2003.

[13] V. Nalwa. Automatic on-line signature verification. *Proceedings of the IEEE*, 85(2):215–239, 1997.

[14] E. Nel and J. du Preez. Estimating the pen trajectories of static signatures using hidden Markov models. *IEEE Trans. PAMI*, 27(11):1733–1746, 2005.

[15] J. Pan and S. Lee. Offline tracing and representation of signatures. *Proc. CVPR*, pages 679–680, 1991.

[16] R. Plamondon and W. Guerfali. The generation of handwriting with delta-lognormal synergies. *Biological Cybernetics*, 78(2):119–132, 1998.

[17] R. Plamondon and F. Maarse. An evaluation of motor models of handwriting. *IEEE Trans. SMC*, 19(5):1060–1072, 1989.

[18] R. Plamondon and S. Srihari. Online and off-line handwriting recognition: a comprehensive survey. *IEEE Trans. PAMI*, 22(1):63–84, 2000.

[19] Y. Qi, M. Szummer, and T. Minka. Diagram structure recognition by Bayesian conditional random fields. *Proc. CVPR*, 2, 2005.

[20] Y. Qiao. Offline Signature Verification Using Online Handwriting Registration. *Technical Report*. Dept. of IE, The Chinese University of Hong Kong.

[21] Y. Qiao and M. Yasuhara. Recover Writing Trajectory from Multiple Stroked Image Using Bidirectional Dynamic Search. *Proc. ICPR*, pages 970–973, 2006.

[22] Y. Qiao, M. Yasuhara, and M. Nishiara. A Framework Toward Restoration of Writing Order from Single-Stroked Handwriting Image. *IEEE Trans. PAMI*.

[23] R. Sabourin, M. Cheriet, and G. Genest. An extended-shadow-code based approach for off-line signatureverification. *Proc. ICDAR*, pages 1–5, 1993.

[24] H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Trans. ASSP*, 26(1):43–49, 1978.

[25] X. Tang, F. Lin, and J. Liu. Video-based handwritten Chinese character recognition. *IEEE Trans. CSVT*, 15(1):167–174, 2005.

[26] D. Yeung, H. Chang, Y. Xiong, S. George, R. Kashi, T. Matsumoto, and G. Rigoll. SVC2004: First International Signature Verification Competition. *Proc. ICBA*, pages 16–22, 2004.