Dynamic Signature Verification using Local and Global Features

Charles E. Pippin
cepippin@cc.gatech.edu

July 2004

Abstract

Handwritten Signature verification is a biometric technique that is useful because signatures are in many practices accepted as a means of identity verification. This work approaches signature verification using seprate filters with different approaches. In the first, global features of the signature, such as average velocity are considered using a Euclidian distance. In the second filter, local features are considered. Strokes are segmented using the minima of the velocity and encoded before comparing them using dynamic time warping and signer-specific thresholds.

1 Introduction

Handwritten signature verification has many practical applications including use in financial transactions, providing electronic signatures for documents, and in providing additional security measures for computer system authentication. Signature verification also has the advantage that is culturally more accepted and less intrusive than other biometric techniques, such as fingerprinting and iris scanning. These characteristics make signatures more easily collectable and accepted as a means of identity verification [1]. Given a MasterCard estimate of over $450 million per year in credit card fraud, proof of identity at the point of purchase is but one very plausible application for verification technology [1]. Indeed, signature hardware tablets are becoming more pervasive in daily life. According to NCR, over 270,000 signature capture tablets were sold to retailers in 2001 alone [2].

It is suggested that an individual's signature is likely to be unique because it is regarded as being ballistic in nature, created by low level motor control functions in the human nervous system [3]. While it may be a simple matter to forge the shape of a signature after practice, forging these dynamic characteristics becomes more difficult. With current hardware it is possible to capture many such dynamic features of an individual's signature, including velocity, pen-tip pressure and pen tilt, in addition to shape characteristics.

Inherently, an individual's signature varies over time, and even between consecutive instances. Therefore, any technique must allow for some intra-class variance while screening inter-class variance [4]. In this work, two verification filters are presented, each employing different techniques common in the literature. The first filter extracts high-level global features of a signature and compares these against stored signature templates using KNN classification. The second filter uses velocity based stroke segmentation to encode the signature as a series of strokes and then uses dynamic time warping to find the closest matches between test and template signatures.

It is hoped that these techniques, when used in concert provide an accurate mechanism for testing the identity of a signer. Moreover, each has its own merits when used alone. It can be seen through the experimental results contained in this paper, the first filter provides an accuracy of 91% and the second an accuracy of 77%.

2 Related Work

Various methods have been reported for online verification, and a common approach is to segment a signature into strokes before processing. Schomaker and Teulings apply velocity based stroke segmentation taking advantage of the ballistic nature of handwriting [4]. Lee et al. segment strokes using geometric extrema [5].

Once a signature has been segmented into strokes, a popular technique for representing features of the stroke is to use a character-based directional encoding scheme [5,9], with the entire signature represented as a string. The string can then be compared to others using time warping methods to match the strings[5]. The time warping technique is applied in other work as well [6, 7] and a treatment of the classic algorithm is given in [8].

In collecting dynamic features, much work has been done using velocity, acceleration, and pen pressure [4]. Pen tilt is another dynamic characteristic that can be useful as a feature [10].

3 Extracting Global Features

Considering only global features of a signature has advantages that it is simple to compute and addresses privacy concerns because it does not need to retain the original signature once the features have been extracted. This makes it ideal as an inexpensive technique that can be used to catch a majority of forgeries, without risk to privacy. In fact, we shall see that with a small number of global features, this technique can classify signatures with approximately 89% accuracy.

Another strength of this approach is that as an individual's signature changes over time, each signature need only be added to the reference database, and newer signatures will naturally be closer to more recent reference signatures.

Table 1: Global Features for Comparison

Feature	Description
Average Pressure	The average pen-tip pressure over the entire signature
Pen Tilt	The average tilt of the pen while writing over the entire signature
Average Velocity	The average x, y velocity over all sample points
Number of Pen Ups	The number of times the pen was lifted over the entire signature
Number of Strokes	The number of velocity based strokes over the entire signature

The global features extracted are listed in Table 1, and compose the feature vector. In using these features, we also gain the advantage that little pre-processing needs to be performed on the raw signature data. As such, these features could be extracted in real time as the signature is captured, eliminating the need to store the raw data at all, if this is the only technique to be used.

A feature vector likewise represents each signature in the enrolled signature database. When a new signature is presented for verification, this is treated as a two-class classification problem, the genuine signatures of the signer and all others. While there are many valid statistical and learning based approaches for such classification, the use of the K-nearest-neighbor algorithm was selected in keeping with the theme of a simplistic approach for this first filter. In this manner, each feature vector is then plotted and the test signature picks those that are closest in Euclidian distance. If a significant number of the closest signatures (as determined empirically) match the claimed class of the signer, then the signature is accepted as valid.

4 Stroke Segmentation and Local Features

The second filter seeks to take advantage of the dynamic characteristics of the signature more effectively, and perform comparisons of local features. The approach taken is to segment the signature into a series of strokes, encode each of the strokes, and find a close matching between the segments. Test signatures can then be compared against template signatures, and if the match is below a signer's specific threshold, the signature is accepted.

Before local features can be compared, sample signatures are preprocessed. Each signature is rotated about the y-axis so that it is aligned horizontally. Secondly, each signature is normalized to a standard width. Because the tablet can record signature information when the pen is not fully touching the tablet, only points with positive pressure are considered. After this stage, the signature consists of a sequence of normalized points.

4.1 Velocity Based Stroke Segmentation

A difficulty in signature verification is in finding the best level of granularity for comparison. Because of the variation between signatures for a single individual, a point-by-point comparison is not ideal. Furthermore, a global comparison will inherently overlook local features. A common approach is the use of the minima of the velocity [4]. In using the velocity information as plotted against the overall signature, two consecutive minima define the endpoints of a stroke. This dependence on velocity rather than shape seeks to take advantage of the ballistic nature of signatures. It would be very difficult in practice for a forger to copy the shape and the dynamic characteristics of a signature [4,11]. By segmenting strokes using the velocity information, the signature is then broken down into more easily analyzed sub-components, an example is shown in Figure 1.

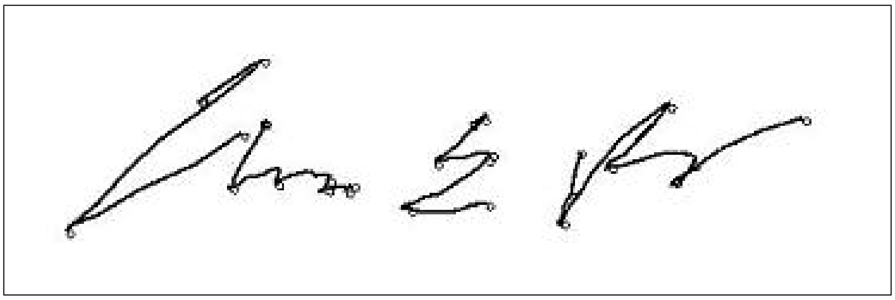

Figure 1: Velocity Based Strokes

4.2 Stroke Encoding

Once the signature has been segmented, each stroke is encoded as a letter, with the entire signature consisting of a string of characters. Here, the approach is most similar to that of [9], in which the origin of each stroke is placed at the origin in the x, y plane, and the endpoint of the stroke ending in one of the four quadrants. Each quadrant is assigned a single letter, (A, B, C, D) as shown in 2. A signature is then represented by a string of these letters, representing a sequence of strokes, such as ”CBACACAACABAD”.

In determining the distance between two such encoded strokes, we first consider whether the encoded stroke letter is the same (i.e. ’A’ vs. ’B’). If they are not the same, then a constant penalty is imposed, with adjacent quadrants receiving half the penalty of opposite quadrants. The comparison between strokes therefore considers stroke direction (via encoding) and dynamic velocity information (via creation of stroke segments.)

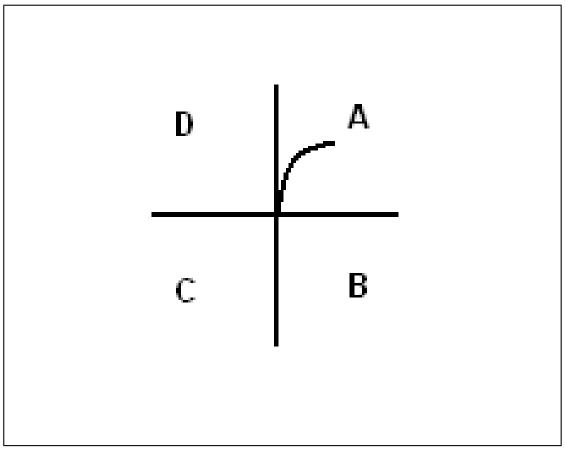

Figure 2: Stroke Encoding

4.3 Dynamic Time Warping

Once the strokes have been encoded, we are faced with the problem of aligning two such encoded strings, and determining a measure of their similarity. If we consider these sequences as a time series, we can employ dynamic time warping as a technique that can be used to find an alignment that minimizes the distances by ”warping” the axis of one or more of the strings to find a better alignment between points. The classic time warping algorithm is presented in [8], and an example of a warping alignment between two signals is shown in Figure 3.

Related research applies time warping to find a similar alignment between stroke endpoints in a signature, and then uses differences between points in that alignment to train a back propagation neural network on the average distances between local features of those points [5]. In a similar vein, this work seeks to find a minimal alignment between stroke segments but employs a signer specific threshold to determine a match. The signer specific threshold is calculated as follows: for each signer, pair wise comparisons are made for all enrollment signatures, and the nth closest distance for that signer defines the threshold. The value to use for n is found empirically. This allows for some intra-class variation while at the same time tailoring the threshold to the differences in each signer.

The dynamic time warping algorithm proceeds in two stages. First an n x m distance matrix is created, and second, a shortest path alignment between opposite corners of the matrix is found. To construct the matrix, one signature (or rather the encoding of a signature) is placed across the top and the other signature across the left side of the matrix. This matrix will represent the distance for every point in one signature to every point in the other signature. The shortest path through this matrix then corresponds to an ideal alignment between points in the signatures. Intuitively, the distance of this path becomes a measurement of similarity between two signatures; the shorter this distance, the more similar two signatures are, and

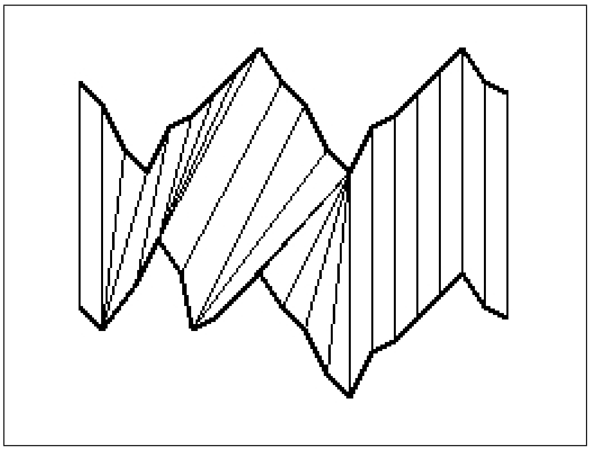

Figure 3: Using Dynamic Time Warping, we can determine the distance between two similar sequences. In this trivial example, the sequences differ slightly in the y-axis and the time axis. The DTW algorithm attempts to find a mapping between differences in the sequence. For our purposes, we consider an alignment of stroke encodings in a sequence.

for identical signatures this distance will be zero.

To calculate the shortest path, dynamic programming is used find the path recursively. The shortest path from a given cell is simply the distance in the current cell along with the smallest distance of a neighboring cell. More formally, this is described by Equation 1, as presented in [8].

$$\gamma(i,j) \quad = \quad d(q_i, c_j) + min\{\gamma(i-1,j), \gamma(i-1,j-1), \gamma(i,j-1)\} \qquad (1)$$

Dynamic programming allows us to efficiently calculate these distances recursively by calculating them in advance and caching the cumulative results. We start by populating the topmost row and leftmost column with the cumulative distances from each cell back to the starting cell (the top-left cell.) From there, we can populate each column, moving from left to right, one cell at a time. The distance at each cell is simply the distance between the (i, j) points at that cell and the minimum of the distance of adjacent cells (i-1, j), (i, j-1), (i-1, j-1). The task of finding the shortest path is reduced to backtracking through this matrix, always selecting the minimum distance among the adjacent cells.

Verification of a new signature with this technique compares the new signature as described above against all stored template signatures. If the test signature's distance from the closest template signature is less than the signer specific threshold, the signature is accepted as valid.

5 Experiments

5.1 Database

In signature verification literature, the most common types of forgeries considered are random forgeries, in which any signature may be substituted as a forgery, simple forgeries, where the forger does not attempt to trace a genuine signature and skilled forgeries [4]. Skilled forgeries are those in which a forger has had time to study and practice a subject's signature and are therefore more difficult to detect. This study performs experiments using skilled forgeries that attempt to pass as genuine.

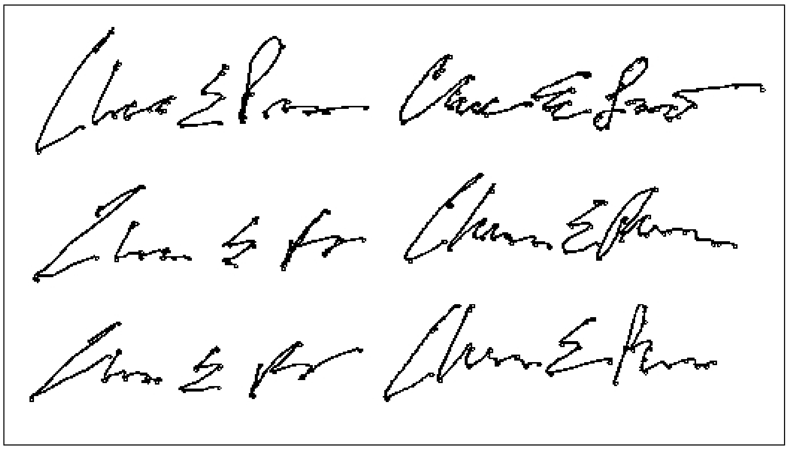

Figure 4: Left: Genuine Signatures. Right: Skilled Forgeries

For this study, all signatures were gathered using a Wacom Intuos2 4x5 digitizing tablet attached to a laptop PC. The capturing software provided visual feedback to the signer and a sheet of paper was placed on top of the tablet to provide a more natural feel while signing. Subjects were asked to sign their name as they normally would, 10 times in succession. For skilled forgeries, subjects were allowed to view another signature and practice it for several minutes before attempting to forge it. Subjects were informed that shape and dynamic characteristics, such as pressure and velocity, would be used in the comparison.

Reference signatures were gathered from 19 test subjects, ranging in age from 20-45. Each subject provided 10 reference signatures and 10 reference words. A subset of the test subjects provided skilled forgeries of other subjects' signatures. This provided a test database of approximately 180 genuine signatures and 73 skilled forgeries of 10 subjects, after errant signatures were removed. Samples of genuine signatures and skilled forgeries are shown in Figure 4.

Two types of error rates are useful in describing accuracy of a signature verification system, as described in [4]. The first is the FAR - False Acceptance Rate (forgeries that were accepted as genuine) and the FRR - False Rejection Rate (gen-

Table 2: Global KNN Filter Results

K	Accuracy	FAR	FRR	Majority Required
4	81.7%	1.2%	17.1%	Yes
7	91.1%	4.3%	4.7%	No

uine signatures that were flagged as forgeries.) The EER - Equal Error Rate is the point at which the FAR and FRR become equal, and is a common measure in the literature.

5.2 Global Filter

To test the performance of the global filter, each reference signature in the database (both genuine and skilled forgeries) was tested using all other signatures to determine if it matched the claimed identity. The K-nearest neighbor algorithm samples the classification of each of the k-closest points to the test instance. The majority classification determines the classification of the new instance. Many different values for k and acceptance thresholds were experimented with, and the best majority classification provides an accuracy of 81%.

However, if we do not require that a majority of the closest signatures agree, only that a single instance of the claimed signature be in the k-nearest set, the best performance found gave an accuracy of 91%. By reducing the sensitivity of the filter in this manner, the EER becomes 4%. These results are show in Table 2.

These results are encouraging given the simplistic nature of this filter and the use of skilled forgeries; however, the addition of new signatures to the reference set could adversely affect the accuracy. For this reason, this filter should be experimented with using a larger dataset. Nevertheless, the inexpensive nature of this filter makes it ideal as a litmus test, for use in determining the most obvious forgeries. Moreover, because the raw signature is not needed for comparison the space requirements for the reference values are minimal.

5.3 Local Filter with Signer Thresholds

To test the local DTW filter, experiments were run with both static thresholds and signer-specific thresholds. During each experiment, the set of all genuine signatures was randomly split into a reference set and a test set, with a 70/30 partition. Each experiment was run 20 times to provide better statistical accuracy. In an experiment, each signature in the test set as well as those skilled forgeries were compared against all reference signatures for the claimed identity. Each comparison returned a distance (as found using the dynamic time warping technique described above) between the test signature and a reference signature for the claimed identity. This test applied the minimal distance principle to the set of distances found: if

Table 3: Local DTW Filter Results

Accuracy	FAR	FRR	Signer Threshold
77.1%	11.9%	10.8%	Yes
72.9%	11.0%	16.1%	No

any of the distances were less than the prescribed threshold, then the test signature passed.

Signer specific thresholds were found for each signer by performing comparisons of all pairwise combinations of that signer's reference signatures. In this set of distances, the nth smallest distance in that set determines the signer threshold. The value for n adjusts the sensitivity level across all signers, and was determined empirically during experimentation such that the FAR and FRR would be approximately equal. The hope is that in using this approach, variability within a set of signatures can be tailored to each signer. As expected, signer specific thresholds performed slightly better than static thresholds, with the averaged result at 77% and an EER around 11

6 Future Directions

A major focus of future work on this project is to better identify which of the global features used in the K-nn comparison are most relevant. The classic K-nn algorithm considers all features in the distance calculation, even though one or more may not be relevant to the classification. By ”tweaking” the weights applied to these features (or removing some of them altogther), it is likely that better results can be obtained. Taking this further, all possible combinations of the 5 selected global features could be experimented with to determine the combination of features that yeilds the best performance.

The local, DTW filter could perhaps be improved by incorporating a learner, such as a back propagation neural network into the comparison, rather than relying on a raw distance threshold. Furthermore, incorporating a confidence of classification into each filter would allow them to be easily integrated into a committee of filters with a voting scheme. In addition, these filters need to be tested on another, larger data set for better statistical accuracy.

A third technique for ensuring the identity of a signer would be to generate a new word dynamically from discriminating features found in reference stroke templates. This technique would seek to extracts strokes and letters from words that a writer has provided. The next step would be to determine which strokes and therefore which letters most distinguish a particular signer. Once this is determined, a new word can be generated dynamically, presented for writing, and the result compared against the stored templates. Intuitively, a dynamic word should be very difficult to forge, as even a skilled forger will have no advance knowledge of it. Rather, the forger would be faced with the more difficult problem of mimicking the writing style of a signer.

Lastly, these techniques may also be useful in the context of other hardware. With the recent arrival of new pen technology, such as the io pen by Logitech, handwriting can be captured using electronic pens and special paper (without a tablet.) Signature verification could be useful here as well; a user of the pen might sign their name first in order to determine the ability access to handwritten notes, for instance.

7 Conclusions

The equal error rate of 4% for the global K-nearest neighbor filter compares more favorably to other signature verification methods than does the equal error rate of 11% in the global DTW filter. However, this serves as an example for more discussion on these techniques. Additionally, these individual filters could be combined into a single system, with a weighted voting scheme.

In summary, two techniques, using dynamic global and local features, for online signature verification were described. It was also shown that signer specific thresholds improved the performance of the local filter. Moving forward, further experimentation on a larger dataset should be performed. However, it is expected that with additional experimentation and adjustment of the feature sets, improved results can be obtained.

8 Acknowledgements

The author wishes to thank Dr. Charles Isbell for his guidance and useful comments during all stages of this project as well as for the use of his digitizing tablet hardware. The author is also grateful to those individuals who volunteered their time and sample signatures for use in these experiments.

References

[1] A.K. Jain, S. Pankanti, R. Bolle (Eds.) BIOMETRICS: Personal Identification in Networked Society. Kluwer Academic Publishers, 1999.

[2] Kramer, Robert. NCR Captures Patent for Electronic Signature Technology: NCR News Release, At *http://www.ncr.com/media_information/2003/apr/pr040203.htm.* April 2, 2003

[3] L.R.B. Schomaker, H.-L. Teulings (1990). A Handwriting Recognition System based on the Properties and Architectures of the Human Motor Sys-

tem. Proceedings of the International Workshop on Frontiers in Handwriting Recognition (IWFHR). (pp. 195-211). Montreal: CENPARMI Concordia.

[4] R. Plamondon, G. Lorette. Automatic Signature Verification and Writer Identification - The State of the Art, Pattern Recognition 22 (2) (1989) 107-131.

[5] J.Lee, H. Yoon, J. Soh, B.T. Chun, Y.K. Chung. Using geometric extrema for segment-to-segment characteristics comparison in online signature verification. Pattern Recognition 37 (2004) 93-103.

[6] Anil K. Jain, Freiderike D. Griess, Scott D. Connell. On-line signature verification. Pattern Recognition 35 (2002) 2963-2972.

[7] Kai Huang, Hong Yan. Stability and style variation modeling for on-line signature verification. Pattern Recognition 36 (2003) 2253-2270.

[8] E. Keogh, M. Pazzani. (2001). Derivative Dynamic Time Warping. In First SIAM International Conference on Data Mining (SDM'2001), Chicago, USA.

[9] A. McCabe. Hidden Markov Modelling with Simple Directional Features for Effective and Efficient Handwriting Verification. Accepted in Procedings of the Sixth Pacific Rim International Conference on Artificial Intelligence (PRICAI 2000), Melbourne, 2000.

[10] H. Shimizu, S. Kiyono, T. Motoki, W. Gao. An electrical pen for signature verification using a two-dimensional optical angle sensor. Sensors and Actuators A 111 (2004) 216-221.

[11] G. Gupta and A. McCabe. A review of dynamic handwritten signature verication," James Cook University Computer Science Dept. Technical Article, 1997.