MULTIPLE FEATURE INTEGRATION FOR WRITER VERIFICATION

Cha, S-H.; Shrihari, S.

Published in:
EPRINTS-BOOK-TITLE

MULTIPLE FEATURE INTEGRATION FOR WRITER VERIFICATION

SUNG-HYUK CHA AND SARGUR N. SRIHARI
Center of Excellence for Document Analysis and Recognition
State University of New York at Buffalo, 14228, U.S.A.
E-mail: {scha,srihari}@cedar.buffalo.edu

Given two handwritten documents, the *writer verification* problem is to determine whether the two documents were written by the same person. It is tackled by extracting various features and classifying the patterns into their classes. Features are diverse in type while techniques in *pattern recognition* typically require that features be homogeneous. The solution proposed overcomes both the non-homogeneity of features and the intractability of infinite number of writers by a *dichotomy transformation*. In this model, the distance between each homogeneous feature type is used. We integrate several distance measures for many feature types: element, histogram, string, convex hull, etc into one useful for *writer verification*. Experimental results with 1,000 writers with three sample documents per writer, using only 12 feature distances, results in 97% accuracy.

1 Introduction

Features encountered in various pattern recognition problems can be diverse in type. Both continuous and non-continuous features have been studied widely in *pattern recognition* [1], *machine learning* and *feature selection* [2] areas. In Liu and Motoda's version of the hierarchy of feature types [2], only elementary feature types were considered: discrete ordinal and nominal, continuous, and complex. Features observed in real application such as the *writer identification* have much more complicated feature types than these elementary feature types. Various types of features are shown in Fig. 1 and we integrate them into one useful for the *writer verification* problem.

The *writer verification* is a process to compare questioned handwriting with samples of handwriting obtained from known sources for the purposes of determining authorship or non-authorship. In other words, it is the examination of the design, shape and structure of handwriting to determine authorship of given handwriting samples. This problem plays an important investigative and forensic role in many types of crime. Document examiners or handwriting analysis practitioners find important features to characterize individual handwriting as features are consistent with writers in normal undisguised handwriting [3]. Authorship may be determined due to the hypothesis that people's handwritings are as distinctly different from one another as their

In: L.R.B. Schomaker and L.G. Vuurpijl (Eds.), Proceedings of the Seventh International Workshop on Frontiers in Handwriting Recognition, September 11-13 2000, Amsterdam, ISBN 90-76942-01-3, Nijmegen: International Unipen Foundation, pp 333-342

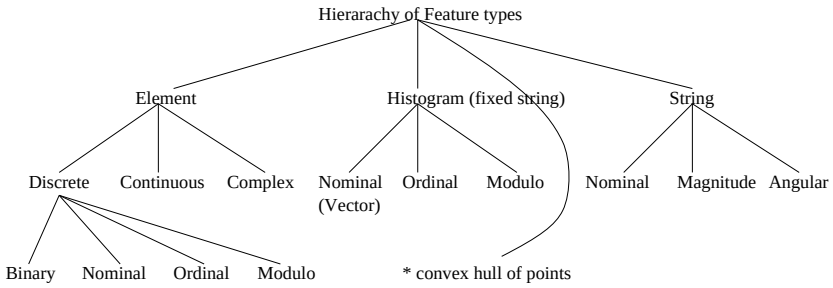


Figure 1. *Transformation from Feature domain to Feature distance domain*

individual natures, as their own finger prints.

The writer identification can be viewed as a U.S. population category classification problem, so called *polychotomizer*. As the number of classes is enormously large and almost infinite, this problem is seemingly insurmountable. Thus, we use a simple *dichotomizer* model that is a classifier that places a pattern in one of only two categories. In order to identify the writer of a questioned document, we model the problem as a two class classification problem: authorship or non-authorship. Given two handwriting samples, the distance between two documents is first computed. This distance value is used as data to be classified as positive (authorship, inner-variation, within author or identity) or negative (non-authorship, intra-variation, between different authors or non-identity). We use within author distance and between authors distance throughout the rest of this paper. Also, we use subscriptions of the positive ($\oplus$) and negative ($\ominus$) symbols as the nomenclature for all variables of within author distance and between authors distance, respectively.

In the feature distance domain (dichotomizer model), all feature distance types are nothing but scalar values and homogeneous regardless of their feature types. Hence, multiple type features are integrated into the feature distance scalar values to solve the writer identification problem. Clearly, the performance depends largely on the distance measure for each homogeneous feature. In this paper, we briefly introduce previously defined various distance measures and their algorithms for many feature types: element, convex hull [4], histogram [5,6], string [7,8].

The rest of this paper is organized as follows. In section 2, the transformation of the discrete ordinal feature domain to feature distance domain is described. Section 3 discusses computing the distance between a set of multi-dimensional points utilizing convex hulls. Section 4 and 5 deal with

the distance between histograms and strings, respectively. Finally, section 7 concludes this work.

2 Dichotomy Transformation and a ANN Dichotomizer

The full description and analysis of the *dichotomy transformation* can be found in [9]. Suppose there are three writers, $\{W_1, W_2, W_3\}$. Each writer provides three documents and two scalar value features extracted per document. The

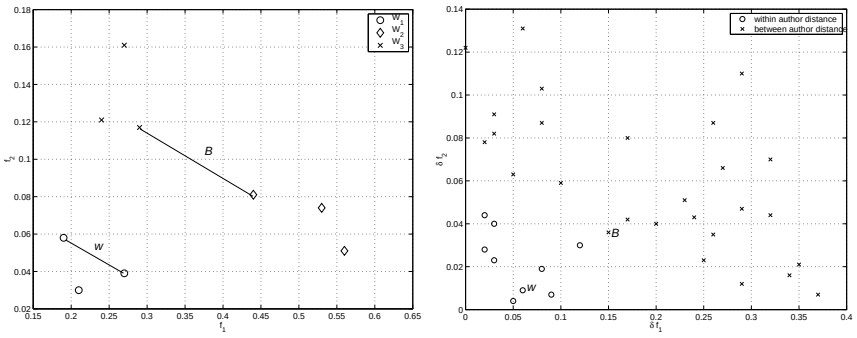


Figure 2. *Transformation from Feature domain to Feature distance domain*

left plot in Fig. 2 shows the plot of documents for every writer. If the number of writers is finite and small, conventional clustering techniques will cluster all documents written by the same author. However, when the number of classes is too large or infinite, clustering techniques are of no use. Instead, we take the distance between writings by the same writer and categorize it as a *within author distance* denoted by $x_{\oplus}$. The sample of *between author distance* is, on the other hand, obtained by measuring the distance between two different person's handwritings and is denoted by $x_{\ominus}$. Let d_{ij} denote i'th writer's j'th document.

$$x_{\oplus} = \delta(d_{ij} - d_{ik}) \text{ where } i = 1 \text{ to } n, j, k = 1 \text{ to } m \text{ and } j \neq k \quad (1)$$

$$x_{\ominus} = \delta(d_{ij} - d_{kl}) \text{ where } i, k = 1 \text{ to } n, i \neq k \text{ and } j, l = 1 \text{ to } m \quad (2)$$

where n is the number of writers, m is the number of documents per person, δ is the distance between two documents. The right-side plot in Fig. 2 represents the transformed plot. The feature space domain is transformed to the feature distance space domain. There are only two categories: *within author distance* and *between author distance*.

Figure 3 depicts the whole process of the *writer verification* using the *dichotomy transformation*. Let f_i^j be the i'th feature of j'th document. First, features are extracted from both document x and y:

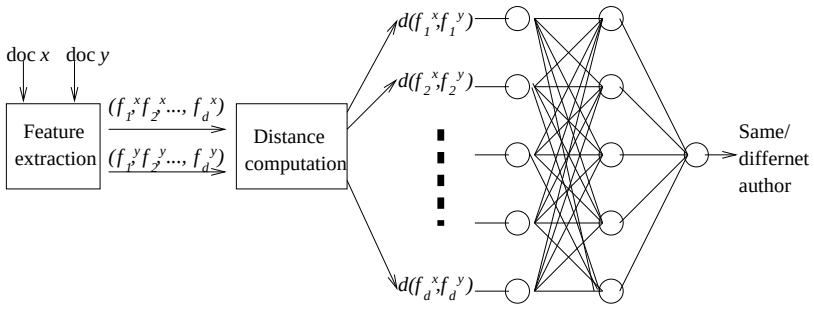


Figure 3. Writer Verification Process and dichotomy transformation

$\{f_1^x, f_2^x, \cdots, f_d^x\}$, $\{f_1^y, \cdots, f_d^y\}$. And then, each feature distance is computed: $\{\delta(f_1^x, f_1^y), \delta(f_2^x, f_2^y), \cdots, \delta(f_d^x, f_d^y)\}$. The *dichotomizer* takes this feature distance vector as an input and outputs the authorship. The following sections discuss various types of features and their distance measures that are used in the *dichtomizer*.

3 Convex hull of points

In this section, the distance between two sets of multi-dimensional points is presented. First, the convex hull of the set of points is computed and then the average distance between the convex hull and all points in the *questioned document* is computed.

Consider one query document and two reference documents (A & B) written by two different writers and one of them is the writer of the query document as shown in Fig. 4. The query document has 5 "W" characters. The document A contains 12 "W" characters and the document B contains 11 "W"'s. Three ordinal features are extracted per character. They are the ratios of height of the peak, the width of the valleys and height/width. In the document level, a set of ordinal three-dimensional points is the feature.

Fig. 5 illustrates the geometrical relationship between the convex hulls of the query document and two documents (A & B). Obviously, the writer of the document B is the writer of the query document. The convex hull of the set of points represents one's handwriting style. The further rationale for

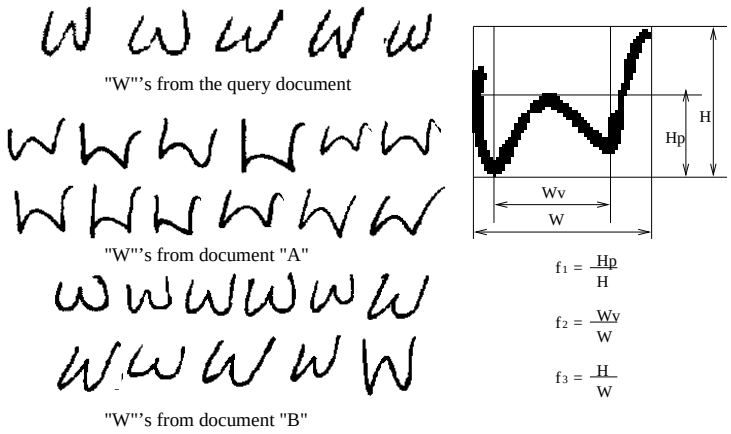


Figure 4. *Sample "W" characters and features*

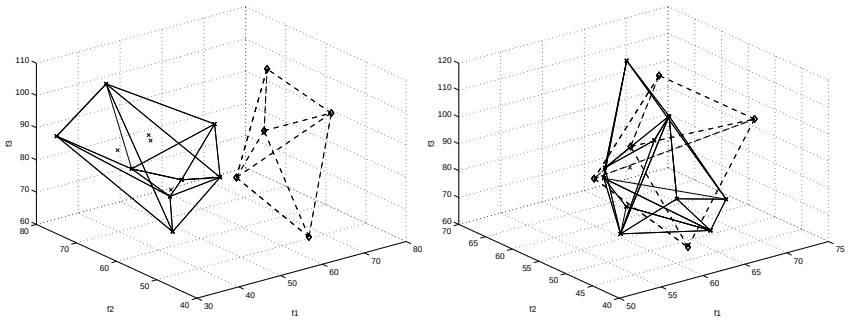


Figure 5. *3d representation of the authorship for document A and B*

computing the distance between the convex hull and all points of the query document is fully discussed in [4].

4 Distance between Histograms

A histogram representation of a sample set of a population with respect to a measurement represents the frequency of quantized values of that measurement among the samples. In the earlier work [5], a distance between sets of measurement values as a measure of dissimilarity of two histograms was pro-

posed and three versions of the distance measure, corresponding to whether the type of measurement is nominal, ordinal, and modulo, were given.

$$\text{Distance between Histograms: } D(A,B) = \min_{A,B}\left(\sum_{i,j=0}^{n-1} d(a_i, b_j)\right) \tag{3}$$

where $d(a_i, b_j)$ is defined differently for its measurement. Also, efficient algorithms for computing the distance between two univariate histograms: $\Theta(b)$, $\Theta(b)$ and $O(b^2)$ for each type of measurements, respectively, where b is the number of levels in histograms.

An example of ordinal histogram is the projection of an image or grey level histogram. The distance can be viewed as the minimum amount of movements of cells to transform one histogram $H(X)$ to the target histogram $H(Y)$. The definition for ordinal type histograms [10] is as follows:

$$D[H(X), H(Y)] = \sum_{i=0}^{b-1} \left| \sum_{j=0}^{i} (H_j(X) - H_j(Y)) \right| \tag{4}$$

It is the sum of absolute values of prefix sum of difference for each level.

However, this approach is not suitable for angular type histograms as the first bin and the last bin are neighbors as shown in Fig. 6. The distance

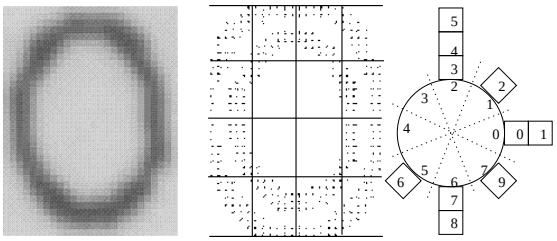


Figure 6. *Sample image, Gradient direction map, and Angular Histograms*

between histograms of angular measurements that is modulo is fully discussed for handwritten character similarity [6].

Gradient direction histograms are computed by the *Sobel* operators: direction $= \tan^{-1} \frac{S_y(i,j)}{S_x(i,j)}$. A sample of the gradient direction maps of a character image is shown in Fig. 6 Sample *"A"* character and the gradient direction histograms are shown in Fig. 7.

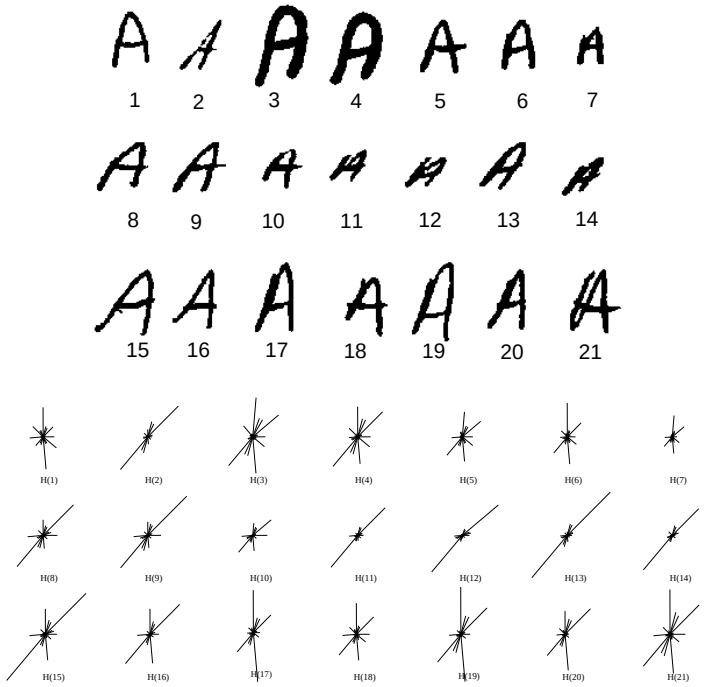


Figure 7. *Sample A's and their Angular Representation of gradient direction histograms*

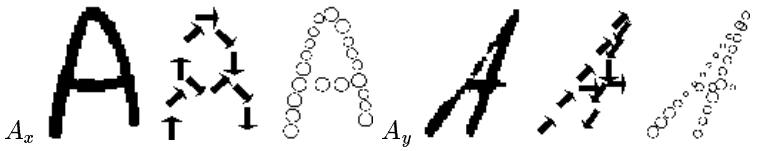
5 Distance between Strings

Characters can be viewed as strings of direction and magnitude. They are denoted as SDSS and SPSS (stroke direction and pressure sequence strings). The sequence of direction is known as *Freeman style code*. Stroke direction and pressure sequence strings of a character were used as character level image signatures for writer identification [7,8].

As the conventional definition of edit distance, also known as *Levenshtein distance* is not applicable for angular or magnitude strings in essence, newly defined edit distances were introduced:

$$T[i,j] = \min \begin{cases} T[i-1,j-1] + d(s_{1,i-1}, s_{2,j-1}) & \Leftrightarrow \text{ turn} \\ T[i-1,j] + 1 + d(s_{1,i-1}, s_{2,j-1}) & \Leftrightarrow s_{1,i-1} \text{ is missing} \\ T[i,j-1] + 1 + d(s_{1,i-1}, s_{2,j-1}) & \Leftrightarrow s_{2,j-1} \text{ is missing} \end{cases} \quad (5)$$

Further justification for the usage of the newly defined *turn*, insertion, and



SDSS(A_x) = [↑ ↗ ↑ ↗ → ↘ ↓ ↘ ⇓] +[↘ ↗]
SPSS(A_x) = [◯ ◯ ◯ ∘ ◯ ◯ O ◯ O]) + [◯ ◯]
SDSS(A_y) = [↗ ↗ ↗ ↗ ↗ ↙ ↙ ↓ ↙ ↙] + [→ →]
SPSS(A_y) = [◯ ◯ ∘ ∘ O O O ◯ O O] + [O ∘]

Figure 8. Sample stroke direction and pressure sequence strings for two handwritten "A"'s.

deletion operations instead of the cost-matrix version of *Levenshtein* edit distance [11] can be found in [8].

To measure the distance between two magnitude strings, we use the *Euclidean* distance after normalizing strings into the same length. We utilize the edit path of the stroke direction strings and the interpolated value is inserted in place of the insertion.

6 Experimental Results

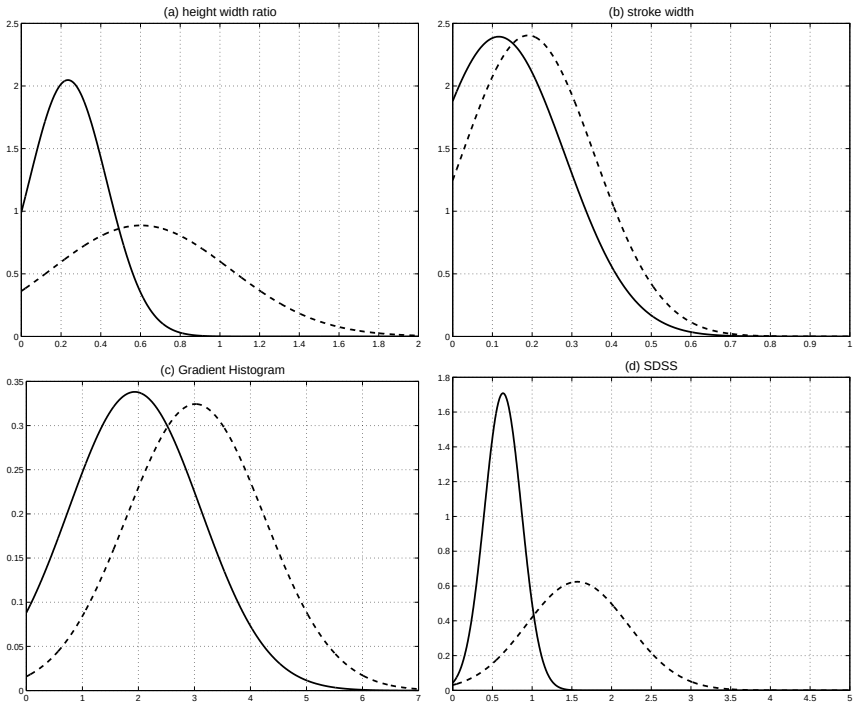
A good descriptive way to represent the relationship between two populations (classes) is calculating overlaps between two distributions. Figure 9 (a)-(d) illustrate the univariate parametric analysis: the two distributions assuming that they are normal. Although this assumption is invalid, we use it to describe the behavior of two population figuratively without loss of generality. The solid and dashed lines indicate the within and between author differences, respectively. The *type I error*, α occurs when the same author's documents are identified as different authors and the *type II error*, β occurs when the two document written by two different writers are identified as the same writer as shown in Figure 9.

$$\alpha = Pr(\text{dichotomizer}(d_{ij}, d_{kl}) \geq T | i = k) \quad (6)$$

$$\beta = Pr(\text{dichotomizer}(d_{ij}, d_{kl}) < T | i \neq k) \quad (7)$$

Let $\hat{X}$ denote the distance x position where two distributions intersect. The*type 1 error* is the right side area of positive distributions where the decision bound $T = \hat{X}$.

Albeit errors are very high in the univariate analysis, errors are reduced when we consider the full multivariate analysis of features such as an *aritificial*



	H/W ratio	Str. Width	Gradient Histo	SDSS
α	9.5%	41.6%	31.0%	4.9%
β	40.5%	40.8%	34.2%	19.5%

Figure 9. *Positive and Negative Sample Distributions for each feature*

neural network. Although features are different in type, the distances between each features are nothing but scalar values and one can dichotomize them with a little confusion.

7 Conclusion

In this paper, we categorized feature types beyond the elementary level. They are a set of multi-dimensional points, univariate histograms, and strings. In the dichotomizer model, all distance features are homogeneous and scalar values. We successfully applied the *dichotomy model* to the *writer verification* problem and achieved 97% overall correctness performance.

Acknowledgments

This research has been possible funded by National Institute of Justice (NIJ) in response to the solicitation entitled *Forensic Document Examination Validation Studies*: Award Number 1999-IJ-CX-K010 [12].

References

1. Richard O. Duda, David G. Stork, and Peter E. Hart. *Pattern Classification and Scene Analysis.* John Wiley & Sons, Inc., 2nd edition, 2000.
2. Huan Liu and Hiroshi Motoda. *Feature Selection for Knowledge Discovery and Data Mining.* Kluwer Academic Publishers, 1998.
3. Russell R. Bradford and Ralph B. Bradford. *Introduction to Handwriting Examination and Identification.* Nelson-Hall Publishers: Chicago, 1992.
4. Sung-Hyuk Cha and Sargur N. Srihari. Convex hull discriminant function and its application to writer identification. In *Proceedings of JCIS 2000 CVPRIP*, volume 2, pages 139–142, Febrary 2000.
5. Sung-Hyuk Cha and Sargur N. Srihari. On measuring the distance between histograms. *submitted to Pattern Recognition*, 2000.
6. Sung-Hyuk Cha and Sargur N. Srihari. Distance between histograms of angular measurements and its application to handwritten character similarity. In *Proceedings of 15th ICPR*, pages –. IEEE CS Press, 2000.
7. Sung-Hyuk Cha, Yong-Chul Shin, and Sargur N. Srihari. Approximate character string matching algorithm. In *Proceedings of Fifth International Conference on Document Analysis and Recognition*, pages 53–56. IEEE Computer Society, September 1999.
8. Sung-Hyuk Cha, Yong-Chul Shin, and Sargur N. Srihari. Approximate string matching for stroke direction and pressure sequences. In *Proceedings of SPIE, Document Recognition and Retrieval VII*, volume 3967, pages 2–10, January 2000.
9. Sung-Hyuk Cha and Sargur N. Srihari. Writer identification: Statistical analysis and dichotomizer. In *Proceedings of SS&SPR 2000*, pages –. Springer-Verlag, September 2000.
10. Sung-Hyuk Cha. Efficient algorithms for image template and dictionary matching. *Journal of Mathematical Imaging and Vision*, 12(1):81–90, February 2000.
11. Robert A. Wagner and Michael J. Fischer. The string-to-string correction problem. *Journal of the ACM*, 6(1):168–178, January 1974.
12. Jeremy Travis. Forensic document examination validation studies. Solicitation: http://ncjrs.org/pdffiles/sl297.pdf, October 1998.