Recognizing hand-printed digits with a distance quasi-metric

Thomas E. Portegys
Lucent Technologies, Naperville, Illinois, USA

Author Information:

Name: Thomas E. Portegys
Address: 846 Emerald St. Naperville, IL 60540
E-mail: portegys@lucent.com
Tel: 630-713-4171
FAX: 630-224-1288

Running head:

A distance quasi-metric.

Recognizing hand-printed digits with a distance quasi-metric

Abstract

A distance quasi-metric for pattern recognition is presented. The “quasi” modifier distinguishes the metric from “true” distance metrics which obey a set of standard constraints. By relaxing one of the constraints, and coupling it with a fast multi-dimensional search technique, the metric demonstrates improved accuracy and efficiency compared to other metrics in recognizing hand-written digit samples. A high-level design of a fast optical comparator for computing the distance in $O(\sqrt{n})$ is also presented.

Introduction

A distance quasi-metric for pattern recognition is presented. The “quasi” modifier distinguishes the metric from “true” distance metrics which obey a set of standard constraints. By relaxing one of the constraints, the metric demonstrates improved search accuracy and efficiency compared to other popular metrics. It has been coupled with a fast multi-dimensional search technique [16] to recognize Optical Character Recognition (OCR) digit samples derived from National Institute of Standards and Technologies (NIST) data [15]. The metric is applicable not only to OCR tasks, but to pattern recognition tasks in general. A high-level design of a fast optical comparator for computing the distance in $O(\sqrt{n})$ is also presented.

The task of recognizing hand-written characters is an active area of research for both neural networks [9,12,13] and memory-based nearest neighbor schemes [4,6,7,14]. Both approaches achieve commendable accuracy [10], but each suffers deficiencies: large numbers of patterns slow the recognition time for nearest neighbor schemes, and neural networks become increasingly difficult to train. A distance metric is crucial to the search for nearest neighbor patterns in multi-dimensional pattern spaces, since it serves not only to determine the classification of a test pattern, but also to guide the search engine efficiently.

Metric description

Let *X* and *Y* be patterns, where a pattern is a spacial configuration of binary-valued pixels. Pixels of unit value are defined as *feature* pixels, and pixels of zero value are defined as *background* pixels. Consider only the feature pixels in each pattern. Let each y_i pixel in *Y* map to the pixel in *X* which is least distant from it, and *pixdist* be the function which computes this Euclidean pixel-to-pixel distance (the computation of *pixdist* for both feature and background pixels in a pattern is called a *distance transform* [2,3]). Similarly, let each pixel x_i in *X* map to those in *Y*. For the *n* pixels in *X* and *m* pixels in *Y*, the distance between *X* and *Y*, *dist(X,Y)* is defined as:

$$dist(X, Y) = dmap(Y \to X) + dmap(X \to Y)$$

$$dmap(Y \to X) = \left(\sum_{i=1}^{m} pixdist(y_i, X) \right) / m$$

$$dmap(X \to Y) = \left(\sum_{i=1}^{n} pixdist(x_i, Y) \right) / n$$

The two *dmap* terms are mean distances of the directed mappings of one pattern to the other. The first term may envisioned as the “fit” of the “hand” of *Y* to the “glove” of *X*, and the second as the fit of *X* to *Y*. Thus henceforth it will be referred to as the glove metric. As special cases, if both patterns have no feature pixels, then *dist(X,Y)=0*, and if one has no feature pixels, then *dist(X,Y)* equals a positive constant value. Taking the mean value serves to make the distance independent of the ‘size’ (number of feature pixels) of the patterns. Otherwise, the distance between similar small patterns would be less than that between large patterns of equivalent similarity.

The glove metric is based on the intuitive notion that a pattern is not just an abstract vector of pixel values, but is a configuration of feature and background pixel values. This view suggests that comparing patterns is a matter of matching the feature pixels in one pattern to those in the other. In OCR and other domains of pattern recognition, this view has produced favorable results. Another approach along these lines involves the use deformable templates [11], in which a maximal overlay of feature pixels is achieved by transforming the geometric configuration of the patterns using parameterized methods. The extent of overlay and magnitude of the transformation determine the distance between the patterns.

Example

Consider the one-dimensional patterns *X* and *Y* in Figure 1:

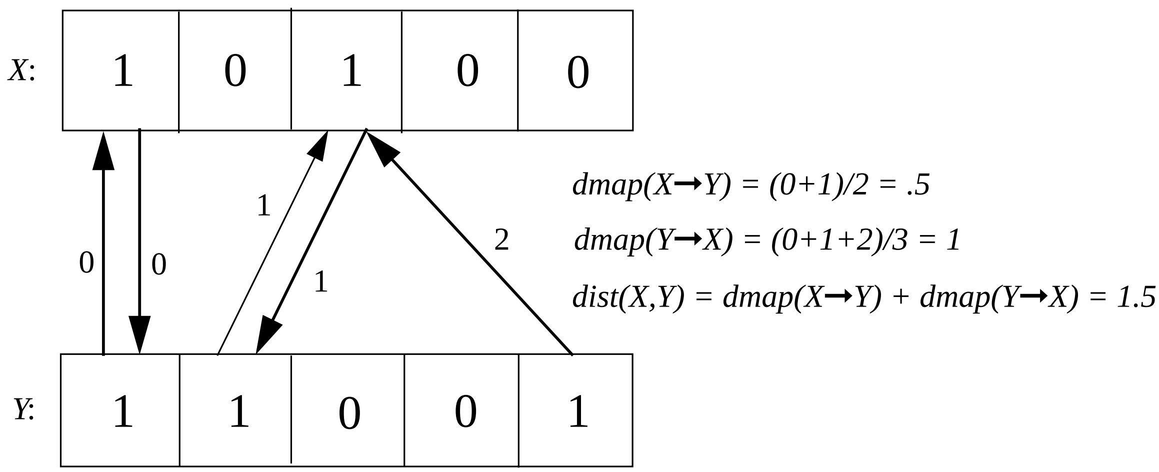

Figure 1 - Distance Example

Analysis

A distance metric satisfies the following relations:

$$dist(X, Y) \geq 0 \qquad (1)$$

$$dist(X, Y) = 0 \Leftrightarrow (X = Y) \qquad (2)$$

$$dist(X, Y) = dist(Y, X) \qquad (3)$$

$$dist(X, Y) \leq dist(X, Z) + dist(Z, Y) \qquad (4)$$

It is easy to show that relations (1)-(3) hold for the glove metric. For true metrics, such as the Euclidean, relation (4), which is called the triangle inequality, holds. For the glove quasi-metric, the triangle relation does not hold, as shown in the counter-example in Figure 2, depicting a superpositioning of patterns *X, Y*, and *Z*:

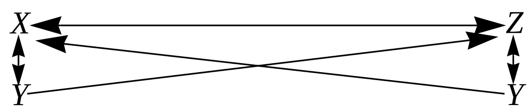

Figure 2 - Relative Distances

If $dmap(X \rightarrow Y)$ and $dmap(Z \rightarrow Y)$ equal 0, then $dist(X,Z) = 2(dist(X,Y)+dist(Z,Y))$. It is tempting to think that this may be the general relationship, but alas, more extreme counter-examples may be found. The triangle relationship is important because it allows a search algorithm to confidently cut off portions of a pattern search space, thus significantly reducing the extent of the search. Conversely, cutting off a search using a metric which does not comply with the triangle relationship may result in overlooking nearest neighbor patterns. For this reason, such a metric may ostensibly seem unworthy of attention. However, at least for OCR patterns, the triangle relationship holds in the preponderant number of cases. Data supporting this claim will be given in the next section.

Results

The glove metric was tested with two other popular distance metrics: the Euclidean [4] and Hausdorff [8]. The Hausdorff distance is simply the maximum of all the *pixdist* distances. For the glove metric test, the search cut off was done as though the triangle inequality held true.

Test procedure:

Each of the metrics under test were plugged into a search engine [16] especially suitable for searching a high-dimensional pattern space (28 x 28 pixels). For each metric, the following procedure was used:

- Build a pattern database using 20,000 randomly ordered NIST training patterns.
- Classify 1,000 NIST test patterns, recording:
  a) The percentage of patterns correctly classified.
  b) The mean number of patterns searched.
  c) The mean number of patterns searched to find the nearest neighbor (NN).

Table 1: Test Results

Metric	Correctly Classified	Mean Searched	Mean to Find NN
Euclidean	95.9%	18764	1180
Hausdorff	96.9%	4558	611
Glove	98.1%	1263	554

The results are shown in Table 1. The glove metric produced both more accurate and efficient results: on average, less than 3% of the 20,000 stored patterns were checked in order to find the least distant. Significantly, a violation of the triangle inequality was detected in fewer than 1% of the inserted patterns, and for these violations, the "long side" exceeded its allowable length by an average of less than 4%.

Optical comparator

The metric calls for computing a set of pixel-to-pixel distances for the *dmap* function. Since OCR patterns may contain hundreds of pixels, this must be done in a fast manner in order to make the metric of practical use. Although algorithmic solutions and specialized digital hardware for fast distance transforms are known [2,3,13], this task is suitable for the inherent parallelism of an optical device [5]. Optical signals can be used to surround the feature pixels in a pattern with distance gradients, forming a *Voroni surface* [8], which are detected by feature pixels in another pattern to yield a directed distance mapping. Such a *dmap(X➞Y)* comparator is shown in Figure 3, comparing two 1-dimensional example patterns.

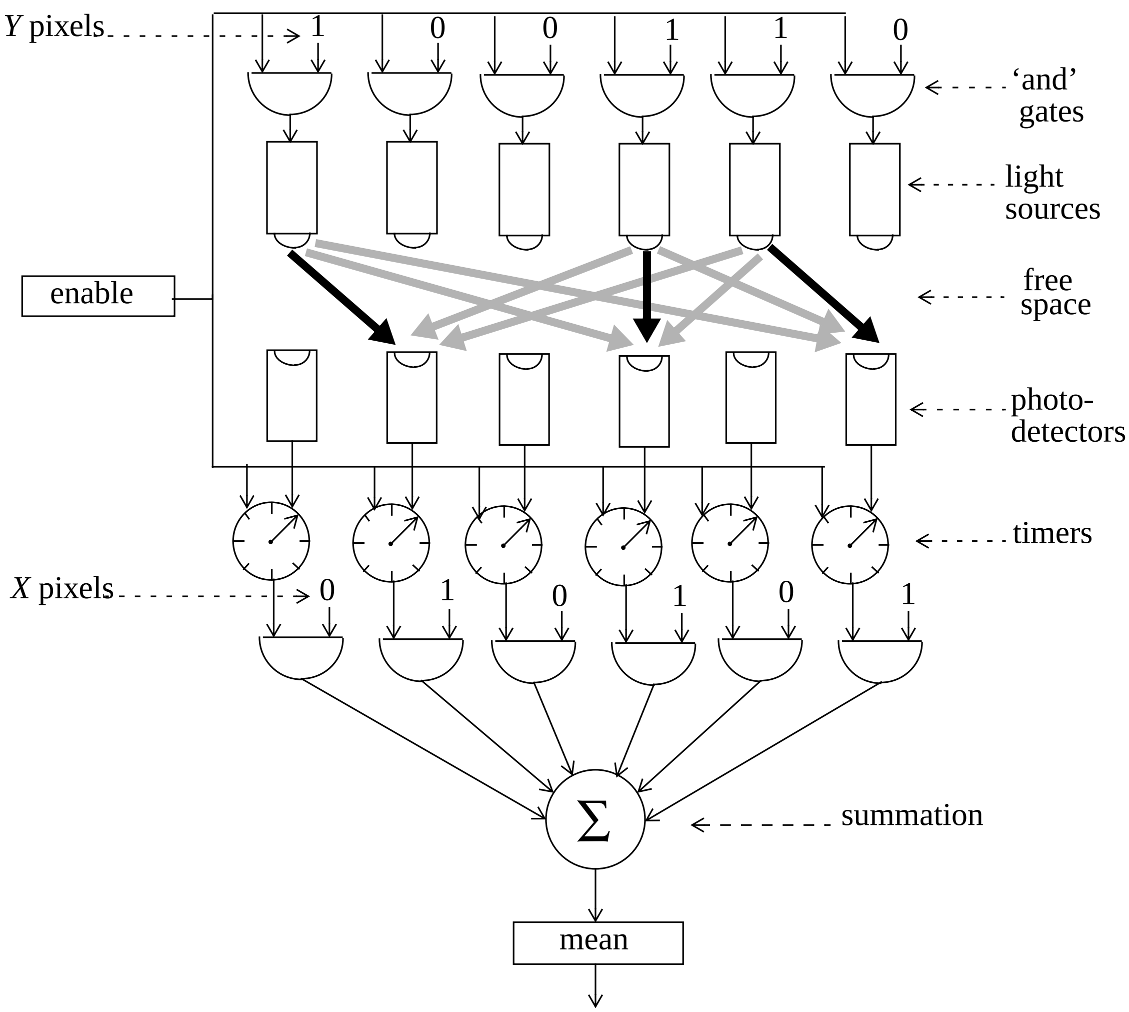

Figure 3 - *dmap(X➞Y)* Comparator

Each feature pixel in *X* must determine the distance to the closest feature pixel in *Y.* To do this, the

Y feature pixels cause the emanations of optical signals which are sensed by photo-detectors. The transit time of the earliest arriving signal is recorded by a timer associated with each detector. This time can be used as a pixel-to-pixel distance. In the example, the earliest arriving signals are denoted by the dark arrows between sources and detectors. The *X* feature pixels enable the output of the timers, which are summed and divided by the number of such pixels to yield the *dmap* output. The entire comparator is triggered by an enable signal, which in this abstraction appears simultaneously at each component (of course, in an actual device this would require careful orchestration). For a timer, the enable signal serves as a reset/start, and the output from its detector stops it.

A 1-dimensional comparator can be extended to 2 dimensions by conceptualizing the example as a side-view of a 2-dimensional comparator. Furthermore, by clustering the pixels in a circular pattern, it can be seen that the comparator response time is proportional to the transit time of light traveling the diameter of the circle, and thus is $O(\sqrt{n})$.

Although the example device depicts a binary-valued pixel comparator, it can be seen that the signals convey pixel value magnitude as well as distance. The magnitude information could be used in a gray-scale (multi-valued) comparator. Indeed, three gray-scale comparators, each processing a primary color, could work in concert as a color comparator. The use of analog electronic circuitry might also serve to improve performance [1].

Conclusion

The glove distance has been found to improve the accuracy and efficiency of a fast search algorithm for hand-printed digits in comparison to other well-known distance metrics. The distance algorithm is amenable to a fast hardware implementation, and in concert with the search algorithm, might be classified as something of a “brute force” pattern recognizer, relying on speed rather than inherent knowledge about OCR digits. Thus said, it is plausible that using such knowledge, e.g., to do edge enhancement and scaling, would lead to further improvements in speed and accuracy for OCR tasks. In addition, although the search algorithm used here is remarkably insensitive to data set size, the storage requirement could be reduced by not storing training patterns which are nearly identical to already stored ones. This may be accomplished by pre-searching a training pattern before deciding to store it.

References

1 H.S. Baird, H.P. Graf, L.D. Jackel, and W.E. Hubbard, A VLSI architecture for binary image classification, J.C. Simon, ed., *From Pixels to Features*, pp. 275-285, Elsevier Science Publishers B.V. (North-Holland), 1989.

2 G. Borgefors, Distance transformations in digital images, *Computer Vision, Graphics, and Image Processing*, vol. 34, pp. 344-371, 1986.

3 H. Breu, J. Gil, D. Kirkpatrick, and M. Werman, Linear time Euclidean distance transform algorithms, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 5, pp. 529-533, May1995.

4 D. Chaudhuri, C.A. Murthy, and B.B. Chaudhuri, A modified metric to compute distance, *Pattern Recognition*, vol. 25, no. 7, pp. 667-677, 1992.

5 *Computer*, issue featuring optical computing, vol. 31, no. 2, February 1998.

6 B.V. Dasarathy, ed., *Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques*, Los Alamitos, Calif.; IEEE CS Press, 1991.

7 K. Fukunaga, *Introduction to Statistical Pattern Recognition*, 2nd Edition, San Diego, Calif.; Academic Press, 1990.

8 D.P. Huttenlocker, G.A. Klanderman, and W.J. Rucklidge, Comparing Images Using the Hausdorff Distance, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 9, pp. 850-864, September 1993.

9 L.D. Jackel, D. Sharman, C.E. Stenard, B.I. Strom, and D. Zuckert, Optical character recognition for self-service banking, *AT&T Technical Journal*, vol. 74, no. 4. pp. 16-24, July/August 1995.

10 A.K. Jain, J. Mao, and K.M. Mohiuddin, Artificial neural networks: a tutorial, *Computer,* vol. 29, no. 3, pp. 31-44, March 1996.

11 A.K. Jain, Y. Zhong, and S. Lakshmanan, Object matching using deformable templates, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 18, no. 3, pp. 267-277, March 1996.

12 J.D. Keeler and D.E. Rumelhart, A self-organizing integrated segmentation and recognition neural net, J. Moody, S.J. Hanson, and R.P. Lippman, eds., *Advances in Neural Information Processing Systems 4*, pp. 496-503, San Mateo, Calif.: Morgan Kauffman, 1992.

13 G.L. Martin and J.A. Pittman, Recognizing hand-printed letters and digits, D. Touretzky, ed., *Advances in Neural Information Processing Systems 2*, pp. 405-414, San Mateo, Calif.: Morgan Kauffman, 1990.

14 S.A. Nene and S.K. Nayar, A Simple Algorithm for Nearest Neighbor Search in High Dimensions, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 19, no. 9, September 1997.

15 The NIST database can be obtained by writing to: Standard Reference Data, National Institute of Standards and Technology, 221/A323 Gaithersburg, MD 20899 USA and asking for NIST special database 1 (HWDB).

16 T.E. Portegys, A search technique for pattern recognition using relative distances, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 9, pp. 910-914, September 1995.

17 I. Ragnemalm, Neighborhoods for distance transformations using ordered propagation, *CVGIP: Image Understanding,* vol. 56, no. 3, pp. 399-409, November 1992.