SIZE AND ORIENTATION NORMALIZATION OF ON-LINE HANDWRITING USING HOUGH TRANSFORM

Amy S. Rosenthal *Jianying Hu* *Michael K. Brown*
Bell Laboratories, Lucent Technologies
700 Mountain Avenue
Murray Hill, NJ 07974, USA
{s_amy, jianhu, mkb}@research.bell-labs.com

ABSTRACT

We introduce a new method for size and orientation normalization of unconstrained handwritten words based on the Hough transform. A modified Hough transform is applied to extremum points along the y coordinate to extract parallel lines corresponding to the boundary lines separating different vertical zones of the handwritten word. One dimensional Gaussian smoothing with variable variance is applied in the Hough space to alleviate the problems caused by the large variation in natural handwriting and the sparseness of extremum points. The method has been tested with and incorporated into an HMM based writer-independent, unconstrained on-line handwriting recognition system and a 25% error rate reduction has been achieved.

1. INTRODUCTION

Preprocessing of on-line handwriting can be classified into two types: noise reduction and normalization. Noise reduction attempts to reduce imperfections caused mainly by hardware limits of electronic tablets, through operations such as smoothing, wild point reduction, hook removal, etc. Normalization refers to the reduction of geometric variations introduced by the writer and typically includes orientation normalization (baseline correction), size normalization (core height correction) and deskewing (slant correction) [1]. For unconstrained handwritten words, the normalization of orientation and size are by far the most difficult tasks among the ones mentioned above, and will be the main focus of this paper.

To facilitate further discussion, we first define four boundary lines for a handwritten word, which include: the base line (joining the bottom of small lower-case letters such as "a"); the core line (joining the top of small lower-case letters); the ascender line (joining the top of letters with ascenders such as "l"), and the descender line (joining the bottom of letters with descenders such as "g"). Orientation normalization involves estimating the orientation of the word, usually through base line detection, and rotating it to the horizontal level, while size normalization involves estimating the core height (the distance between the base line and core line) of the word and rescaling it to a standard length. These two tasks are closely related: a good estimate of core height relies on accurate estimation of the orientation. Both tasks become very difficult for unconstrained handwritten words, where the boundary lines are often not well defined, as shown in Fig. 1.

Various techniques have been developed in the past to tackle this difficult problem. The main approaches include the histogram based methods [2, 3, 1, 4], linear regression based methods [5], and model based methods [6]. In this paper we describe a new approach based on the Hough transform. First, local maximum and minimum points along the y coordinate are extracted, then a modified Hough transform is applied to extract parallel lines corresponding to the boundary lines. In order to handle the problems caused by the large variation in natural handwriting coupled with the sparse nature of extremum points in each word sample, one-dimensional Gaussian smoothing with adjustable variance is applied in the Hough space, followed by parameter refinement using linear regression. Compared with previous techniques, the Hough transform based method has the advantage that it simultaneously provides the optimal estimates of both the orientation and the core height of an input word. The method has been tested with our HMM based writer-independent, unconstrained on-line handwriting recognition system [7] and a 25% error rate reduction has been achieved.

Figure 1. A handwritten word whose boundary lines are not well defined.

In the next section we give a brief description of the modified Hough transform. In section 3 we describe in detail how to apply Hough transform to detect boundary lines of handwritten words. Experimental results are provided in section 4, and we conclude in section 5.

2. MODIFIED HOUGH TRANSFORM

The Hough transform [8] is now a well known method for identifying patterns in the presence of noise. Unlike regression or Bayesian methods, the Hough transform will completely ignore pattern outliers.

The most commonly treated problem in the literature is the detection of a straight line in a noisy image, however, in principle any pattern can be transformed. In its simplest form we consider the transformation of the linear equation

$$y = mx + b \tag{1}$$

from the *pattern* space (x, y) to the Hough *parameter* space (m, b). For each point (x, y) in the pattern space, we gather evidence for a particular line (or set of lines) in the parameter space by quantizing the parameter space into an array of accumulator bins and incrementing bins for all values of (m, b) that satisfy (1). The resulting accumulation of counts is representative of an estimate of probabilities of the line parameters taking on the corresponding values. Thus, a single point in pattern space transforms to a line in parameter space. A set of points on a single line in pattern space transforms to a pencil of lines intersecting (approximately) at a single point that represents the most likely value of (m, b) for the line in pattern space.

One difficulty with this method arises when m become infinite. To handle this problem, other transformation functions are used such as a circular mapping or a sinusoidal mapping [9]. We use the sinusoidal Hough transform from (x, y) to (ρ, θ):

$$\rho = x \cos\theta - y \sin\theta. \quad (2)$$

where a line through point (x, y) is specified by the angle θ and length ρ of its normal from the origin. Equation (2) can also be written as an integration kernel

$$h(x, y, \rho, \theta) = I(x, y)\delta[x \cos\theta + y \sin\theta - \rho]. \quad (3)$$

where I is the image magnitude in the pattern space and *delta* is the Dirac delta function that is normally integrated into a sinusoidal string of accumulator bins. Typically I is binary valued taking on unit value at points on image lines. An accumulator bin thus receives a unit count where singularities appear in h.

Practical considerations prevent this method from being applied to arbitrary pattern functions. The most common difficulty is the size of the accumulator array can be very large when there are many parameters and/or when high resolution on one or more of the parameter values is required.

There is also a necessary compromise between parameter resolution and noise sensitivity. While it is desirable to estimate parameters with high precision, thus dictating a high resolution accumulator, care must also be taken that noise does not so widely scatter the accumulated counts in the neighborhood of the most likely parameter values that no single bin dominates in the neighborhood. This latter difficulty is compounded when the input pattern is sparsely sampled, yielding few counts in the parameter space, as is the case in our current work. Hence it is also desirable to lower the resolution to obtain higher accumulator counts.

One method for addressing this tradeoff is to take a probabilistic view of the pattern space and correspondingly the parameter space. To motivate and ground this probabilistic notion in the real world we can assume that each measurement in the Euclidean pattern space is independently noisy and each pattern point can be represented by an independent identically distributed probability density function. For optical patterns we can also point out that the optical transfer function, when applied to each pattern point, yields a point spread function that has similar smoothing properties. The physical justification is not very important, and we will depart from this justification later in choosing other smoothing functions, but the resulting effect on the transformation process is important because it allows us to handle noise at higher resolution.

Replacing the image function $I(x, y)$ in (3) with a radial basis function yields non-isotropic smoothing in the parameter space since the Hough transformation is nonlinear. The effect of this modification is more easily seen by rewriting the integration kernel (3):

$$h(x, y, \rho, \theta) = I(x, y)\delta[(x^2 + y^2)^{\frac{1}{2}} cos(\theta - tan^{-1}(\frac{y}{x})) - \rho], \quad (4)$$

which is easily obtained after some algebraic manipulation, to show the non-linear affect of (x, y) on (ρ, θ). The absolute and relative values of x and y affect both the magnitude and phase of the cosine component. Thus, a radial basis smoothing function is transformed into an approximately elliptical function with major axis alternately oriented along the ρ axis and θ axis throughout the parameter space.

While in principle we can compute this modified transform, the computation is expensive. We can also approximate the effect while saving considerable computation by applying a one-dimensional smoothing operation in the parameter space. We chose to implement this smoothing operation by applying Gaussian smoothing in the ρ dimension only. In our implementation the normally integer valued accumulator bins are replaced by real valued bins and the transformed function is smoothed before incrementing the bin values. Various methods of determining the variance of the Gaussian have been tested and are discussed in detail in the next section.

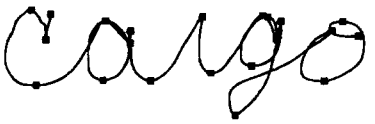


Figure 2. Sample containing backtrack strokes.

3. BOUNDARY LINE EXTRACTION USING HOUGH TRANSFORM

The modified Hough Transform is applied to extract boundary lines among points of y maxima and minima of a given handwritten word. First, a search is performed through the sample points to find consecutive pairs of y maxima and minima. These extrema are selectively chosen to reduce the number of "spurious points". For example, pairs making up small peaks are considered noise and are not included in the final set of extrema; pairs which form relatively long horizontal segments are also ignored as they are usually ligatures between letters and do not contribute significant information about the boundary lines.

Another type of spurious extrema are formed by backtracking strokes in cursive handwriting. For example, the sample "cargo" shown in Figure 2 contains backtracks in the letters "c", "a" and "g" which create minimum points forming a false base line. A heuristic was developed to isolate these types of minimum points so they can be removed from the data set. The strokes between a maximum and its corresponding minimum are considered one segment, as are the points between that minimum and the following maximum. If the curvature of the first segment is positive and that of the second segment is negative, and both curvatures are significant, the segments are searched for a cusp. If a cusp exists in the region of the minimum point, a backtrack is assumed to exist and the misleading minimum point is removed from the data. Both maximum points are retained as they lend support to each other for locating the appropriate core line.

The final set of extremum points are called *pattern points*. Since each of the boundary lines is formed by either maximum points or minimum points, the sets of maximum and minimum points are transformed to two separate Hough accumulator arrays (maximum accumulator and minimum accumulator) to avoid unnecessary confusion. The bin sizes for the arrays need to be chosen carefully. On the one hand, they should be fine enough to distinguish the potential ascender line (descender line) from the core line (base line). On the other hand, the bins must also be coarse enough to accurately cluster the data while accommodating noise and natural variation of handwriting. Our experiments show that the choice of bin size for ρ is more crucial than that for θ. Several different methods for computing the bin size for ρ were tested, including varying the bin size according to the length of the word; varying the bin size according to an initial estimate of the core height of the word based on the median of the y differences between maxima and minima; and simply using a constant bin size regardless of the size of the input. Interestingly, the best performance was achieved using a constant bin size which was empirically estimated from a large number of samples.

As explained in the previous section, one-dimensional Gaussian smoothing is applied in the ρ dimension to alleviate problems caused by noise, quantization error and the lack of data points. For each point (x_i, y_i) in the pattern

space, the increment for bin $[\theta_j, \rho_k]$ is computed as:

$$h(x_i, y_i, \theta_j, \rho_k) = e^{-\frac{(\rho_k - \rho_{ij})^2}{\sigma_{ij}^2}} \quad (5)$$

where

$$\rho_{ij} = x_i \cos\theta_j - y_i \sin\theta_j, \quad (6)$$

and σ_{ij} is the variance of the Gaussian kernel applied to point (x_i, y_i) for angle θ_j. We shall call $h(x_i, y_i, \theta_j, \rho_k)$ the *contribution* of point (x_i, y_i) to bin $[\theta_j, \rho_k]$.

When smoothing is applied in the pattern space, the smoothing deviation σ_{ij} varies throughout the Hough parameter space, as discussed in the previous section. It can be estimated in various ways depending on the smoothing model chosen. In the simplest case, σ_{ij} is assigned a constant σ_0 for all i and j. The result is that the sinusoids of the Hough space appear to experience less smoothing in the θ direction near the θ axis ($\rho \approx 0$) because we are smoothing only in the ρ dimension and because of the high slope of the (ρ, θ) function in this region. We call this the *cosine effect*.

In an alternative implementation that addresses the cosine effect, σ_{ij} is computed as:

$$\sigma_{ij} = \sigma_0 (1 + \eta_{ij}^2)^{\frac{1}{2}} \quad (7)$$

where $\eta_{ij} = \frac{\partial \rho}{\partial \theta}|_{i,j} = -x_i \sin\theta_j - y_i \cos\theta_j$. The effect of (7) is to increase the smoothing operator variance in regions where $|\partial\rho/\partial\theta|$ is high. Intuitively, while the transformation is not truly isotropic, the result is to approximate this transformation so that the sinusoid is uniformly dilated in the parameter space.

In yet another implementation, we depart from the notion of physical justification and assume that y_i follows a Gaussian distribution with variance σ_y in the pattern space while x_i remains fixed, thus σ_{ij} is:

$$\sigma_{ij} = \sigma_y \sin\theta_j. \quad (8)$$

This results in larger smoothing variance in regions of the Hough space representing nearly horizontal pattern lines (θ close to $\frac{\pi}{2}$) and small variance in regions representing nearly vertical lines (θ close to 0 or π). This effect, taken together with the sparse nature of our data, results in the enhanced detection of nearly horizontal pattern lines and has the effect of suppressing lines that are nearly vertical, which are rare as boundary lines since most handwritten words have a nearly horizontal orientation. We have experimented with all of the above three methods and the third method demonstrated the best performance.

Aside from accuracy, another concern in performing the Hough transform is how to reduce the amount of computation. In a straightforward implementation of the transform, for each pattern point (x_i, y_i) and each quantized angle θ_j within the range $[0, \pi]$, the center ρ_{ij} is computed using (6), then the contribution $h(x_i, y_i, \theta_j, \rho_k)$ is computed using (5) for each ρ_k inside a window of predetermined length L centered at ρ_{ij}. We shall refer to the computation of the center ρ_{ij} along with the corresponding L contributions as one *transform operation*. Suppose the number of quantization levels for $\{\theta : \theta \in [0, \pi]\}$ is M, and an input sample has N pattern points, then the transform operation is applied $N \times M$ times for this sample. There are two possible methods to reduce the amount of computation.

The first method takes advantage of the fact that the amount of pattern points for most words is small (usually fewer than 30). If instead of examining each quantized angle for each pattern point, we only examine a small number D of quantized angles close to angles of lines formed by pairs of maximum (minimum) points, then the number of the transform operation to be applied becomes $N \times N \times D$. This could lead to significant reduction of computation if $N \times D$ is much smaller than M. However, our experiments indicate that due to the large variation in natural handwriting, this scheme tends to result in heavy compromise in performance with little reduction in computation.

The second, more effective method is to first obtain a rough estimate of the orientation of the word using a simple and fast procedure (e.g., linear regression), compute the angle $\tilde{\theta}$ of the normal to this orientation, then restrict the range of θ to $[\tilde{\theta} - \phi, \tilde{\theta} + \phi]$, where ϕ is the estimated confidence margin of the initial estimate. This method not only reduces the number of transform operations to be carried out for each pattern point, but also reduces the range of angles to be searched later for the boundary lines. In our experiments using linear regression to provide the initial estimate, little degradation in performance is observed when the value of ϕ is as small as 15 degrees, while the amount of computation is reduced to one-sixth of the original.

After the Hough transform is completed, the two accumulator arrays are searched for parallel lines at each quantized angle as potential boundary lines. Since a word sample always contains the base line and core line, while it may or may not contain the ascender line or descender line, we search for the top two strongest peaks in the ρ histogram corresponding to each angle in each accumulator array. The presence of the ascender (descender) line is determined by comparing the counts at the peaks. If only one peak is found, or the second peak is much weaker (i.e., has much lower count) than the first one, then it is assumed that there is no ascender (descender) line in the sample. If on the other hand, two peaks of comparable strength are found, then it is assumed that the ascender (descender) line is present, and the identity of each of the two parallel lines is determined by their relative position (the line closer to the center of the word is the base line or core line). The *combined strength* of an angle θ_j is defined as the sum of the counts of the highest peaks in the ρ histogram for θ_j in the two accumulator arrays. The angle with maximum combined strength is chosen to be the optimal angle. Fig. 3(a) shows a sample word "bluet", whose core line and ascender line are of comparable strength, and Fig. 3 (b) shows the ρ histograms of the sample at the optimal angle, with two peaks in the maximum accumulator and one peak in the minimum accumulator.

Once the optimal angle and the corresponding bins for the base line and core line are identified in the above manner, one could simply use the θ and ρ values at the center of each bin as parameters for these two lines and normalize the input sample accordingly. However, since both ρ and θ have been rather coarsely quantized, this approach results in poor accuracy. One method to increase the effective resolution is to apply quadratic interpolation to several bins near the chosen bin and search for the precise peak position which may not necessarily be at the center of the chosen bin. We have instead chosen to use a simpler procedure, double linear regression, to refine the parameter estimates.

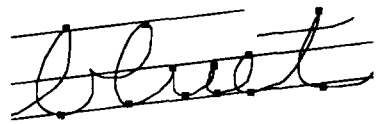
Suppose that $(x_c(i), y_c(i))$ is the set of maximum points with large contributions to the chosen bin for the core line, and $(x_b(j), y_b(j))$ is the set of minimum points with large contributions to the chosen bin for the base line. The following double linear regression procedure is applied to fit a line through each of the two sets of points, with the constraint that the two resulting lines are parallel to each other [5]:

$$y_1 = ax_1 + b_1; \quad y_2 = ax_2 + b_2 \quad (9)$$

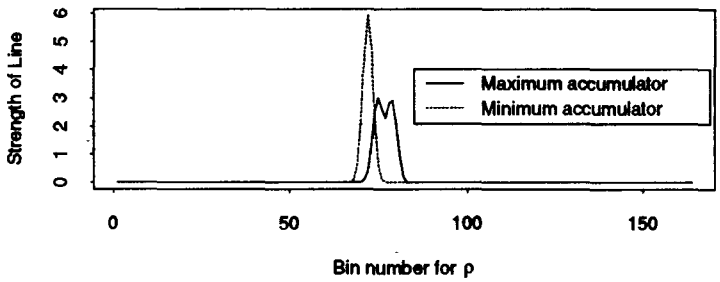
where

$$a = \frac{\overline{x_1} \cdot \overline{y_1} + \overline{x_2} \cdot \overline{y_2} - \overline{x_1 y_1} - \overline{x_2 y_2}}{\overline{x_1}^2 + \overline{x_2}^2 - \overline{x_1^2} - \overline{x_2^2}};$$

$$b_1 = \overline{y_1} - a\overline{x_1};$$



(a)



(b)

Figure 3. A sample word containing equally dominant core and ascender lines, and its ρ histograms at the optimal angle.

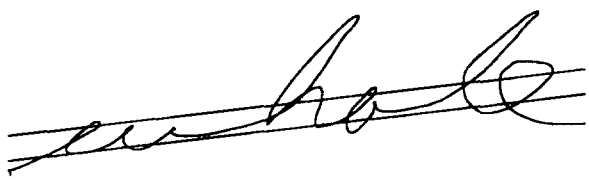


Figure 4. A handwritten word and the estimated base line and core line.

$$b_2 = \overline{y_2} - a\overline{x_2}.$$

Since the position of the extrema could be slightly shifted after orientation correction, further improvement on accuracy is achieved by rotating the sample to horizontal position using the above estimates, recomputing the double linear regression using the new extrema, and iterating through this procedure until the parameters converge (usually after 2 iterations).

Fig. 4 illustrates the estimated base line and core line of the word sample shown in Fig. 1.

4. EXPERIMENTAL RESULTS

The Hough transform based size and orientation normalization algorithm has been tested on an HMM based on-line handwriting recognition system called AEGIS [10], designed for writer-independent recognition of unconstrained handwritten words. Four features were used in the experiments: tangent slope angle, y coordinate, normalized curvature and ratio of tangents. The first two features are directly affected by the normalization process, while the last two are invariant under both scaling and rotation [11]. The decoding of an input sample is carried out in two steps: constrained N best decoding followed by rescoring using segmental features and refined delayed stroke modeling [12].

Experiments were carried out using unconstrained handwritten lower-case word samples from 56 writers. The models were trained using about 5000 samples from 40 writers. The test set contains 125 samples from each of the other 16 writers, totaling 2000 samples. The words were drawn randomly from a 25,000 word dictionary. A lexicon of 1905 words, covering all unique words in the test set, was applied in recognition. The test samples exhibit a wide range of core heights (from 1.2mm to 8.4mm) and orientations (up to 25 degrees away from the horizontal). In computing the modified Hough transform, a constant bin size of 1 degree for θ, and $1.2mm$ for ρ were used . When no size or orientation normalization was applied, we obtained an error rate of 16.4%. After applying Hough transform based size and orientation normalization, the error rate dropped to 12.0%, yielding an over 25% error rate reduction.

5. CONCLUSION

We have described a new on-line handwritten word normalization algorithm base on a modified Hough transform. Compared with previous methods, this new algorithm has the advantage that it simultaneously provides optimal estimates for both the orientation and core height of an input sample. Experiments using this normalization method with an HMM based recognizer yielded substantial reduction in error rates.

REFERENCES

[1] W. Guerfali and R. Plamondon. Normalizing and restoring on-line handwriting. *Pattern Recognition*, 26(3):419–431, 1993.

[2] D. J. Burr. A normalizing transform for cursive script recognition. In *Proc. 6th ICPR*, volume 2, pages 1027–1030, Munich, October 1982.

[3] M. K. Brown and S. Ganapathy. Preprocessing techniques for cursive script recognition. *Pattern Recognition*, 16(5):447–458, November 1983.

[4] H. S. M. Beigi, K. Nathan, G. J. Clary, and J. Subrahmonia. Size normalization in on-line unconstrained handwriting recognition. In *Proc. ICASSP'94*, pages 169–172, Adelaide, Australia, April 1994.

[5] M. Schenkel, I. Guyon, and D. Henderson. On-line cursive script recognition using time delay neural networks and hidden Markov models. In R. Plamondon, editor, *Special Issue of Machine Vision and Applications on Cursive Script Recognition*. Springer Verlag, 1995.

[6] Y. Bengio and Y. LeCun. Word normalization for on-line handwritten word recognition. In *Proc. 12th ICPR*, volume 2, pages 409–413, Jerusalem, October 1994.

[7] J. Hu, M. K. Brown, and W. Turin. HMM based on-line handwriting recognition. *IEEE PAMI*, 18(10):1039–1045, October 1996.

[8] P. V. C. Hough. Method and means for recognizing complex patterns. *U.S. patent 3,069,654*, 1962.

[9] R. O. Duda and P. E. Hart. Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM*, 15(1):11–15, 1972.

[10] J. Hu, M. K. Brown, and W. Turin. Handwriting recognition with hidden Markov models and grammatical constraints. In *Proc. 4th IWFHR*, pages 195–205, Taipei, Taiwan, December 1994.

[11] J. Hu, M. K. Brown, and W. Turin. Invariant features for HMM based handwriting recognition. In *Proceedings ICIAP'95*, pages 588–593, Sanremo, Italy, September 1995.

[12] J. Hu and M. K. Brown. On-line handwriting recognition with constrained n-best decoding. In *Proc. 13th ICPR*, volume C, pages 23–27, Vienna, Austria, August 1996.