Character Recognition Using Mathematical Morphology

Tapas Kanungo and Robert M. Haralick

Intelligent Systems Laboratory
Electrical Engineering Department, FT-10
University of Washington
Seattle WA 98195
U.S.A.

September 14, 1990

Abstract

We describe our partially completed work on a set of morphological algorithms for recognizing handwritten numerals. The basic morphological operations – dilation, erosion, opening and closing – are used to extract characteristic features of these characters and check for their appropriate topological configuration. The paper describes the key shape characterization for two groups containing a total of the six numerals on which we have so far worked and give the morphological algorithms for their recognition.

1 Introduction

The problem of character recognition has attracted many researchers in the past. However, there has been a renewed interest in this problem in the recent years. Reasons for this interest range from the technological improvements in hardware required for user friendly interfaces to impetus provided by application areas such as postal mail sorting. Numerous commercially available character recognition systems with varied performance characteristics exist today in the market. These character recognition systems use varied algorithms and work on varied type of input data. A comprehensive survey of these systems, techniques and data sets can be found in the recent survey article by Tappert, Suen and Wakahara [TSW90].

It is evident that the hand written characters are shapes with high variability. Often the written character might be ambiguous but the context usually eliminates the ambiguity. Many techniques have been proposed for character and numeral classification. These techniques can be classified into two categories: on-line techniques and off-line techniques. In the on-line case, the recognition has to be performed while the characters are being written. This implies that the recognition has the dynamic information about the character – the chronological order of strokes that were made to compose the character. In the off-line case, no such dynamic information is available and the recognition is attempted after the writing is over. In this paper we will be addressing the off-line approach only.

Some of the known off-line techniques use feature analysis, analysis by synthesis, and pairwise detection to classify characters. In feature analysis, a set of features are used to represent a character. The input image is processed and if the set of features detected match the set of features representing a particular character, we say that the character is detected. In analysis by synthesis the characters are represented by modelling their generation using a set of primitive curves and rules. These rules are simple rules for connecting the primitives together. The input character is broken down into strokes and each of them is represented using the generation model. This representation is then matched to the prototype characters. In pairwise detection, special algorithms are developed for pairs of characters that are highly likely to be confused. For example, O and C, V and Y.

Mathematical morphology is an important tool that can be used to process images based on the shape information. Not much literature exists on methods that exploit this technology for character recognition. Mitchell *et al.*[MG89] use mathematical morphology to extract features and then have a high level module to get topological information about these features and match it with a database of stored prototypes. In this paper we describe our initial work on algorithms to recognize hand written numerals using mathematical morphology. We use the basic morphological operators – dilation, erosion, opening and closing – to extract various features and check for various topological configurations. In section 2 we give the basic definitions and notation used in this paper. We describe the basic algorithm in section 3 and give the implementation details in section 4. We give the result in section 5 and in the same section discuss the implication of the results. Finally, we conclude by summerizing our results in section 6.

2 Definitions and Notation

In this section we define all the necessary terms and give the notations used in this paper. The definitions of the basic morphological operations based on the tutorial by Haralick *et al.* [HSZ87] have been restated below for easy reference.

Dilation is the morphological transformation which combines two sets using vector addition of set elements. If A and B are sets in $\mathbf{Z}^2$, the dilation of A by B is the set of all possible vector sums of pairs of elements, one coming from A and one coming from B.

Definition 2.1 The *dilation* of A by B is denoted by $A \oplus B$ and is defined by

$$A \oplus B = \{c \in \mathbf{Z}^2 \mid c = a + b \text{ for some } a \in A \text{ and } b \in B\}.$$

Erosion is the morphological dual of dilation. If A and B are sets in $\mathbf{Z} \times \mathbf{Z}$, then the erosion of A by B is the set of all elements of x for which $x + b \in A$ for every $b \in B$.

Definition 2.2 The *erosion* of A by B is denoted by $A \ominus B$ and is defined as

$$A \ominus B = \{x \in \mathbf{Z}^2 \mid x + b \in A \text{ for every } b \in B\}.$$

Opening an image with a disk structuring element smooths the contour, breaks narrow isthmuses, and eliminates small islands and sharp peaks or capes.

Definition 2.3 The *opening* of a set B by a structuring element K is denoted by $B \circ K$ and is defined as

$$B \circ K = (B \ominus K) \oplus K.$$

The morphological operation *closing* smooths the contours in an image, fuses narrow breaks and long thin gulfs, eliminates small holes, and fill gaps in the contours.

Definition 2.4 The closing of a set B by a set K is denoted by $B \bullet K$ and is defined as

$$B \bullet K = (B \oplus K) \ominus K.$$

3 The Algorithm

Each handwritten numeral has some characteristic shape feature that distinguishes it from other numerals. Some of the features might be present in a subset of characters and not present in the rest. For example, the numerals 1, 4, 7 and 9 have a near vertical stem that differentiates them from 0, 2 , 3 5, 6 and 8. But only 4 and 9 have a "blob" towards the upper left part of the stem. Thus, this feature distinguishes 4 and 9 from 1 and 7. Further, the "blob" of 4 usually has a concavity facing north where as the "blob" of 9 does not have such a concavity. Thus, 9 and 4 can be distinguished by using this concavity feature.

Our approach to the recognition of handwritten numerals is based on a morphological decision tree. At each node of the tree a sequence of operations is performed which distinguishes one group of numerals from another. We have just begun the task of developing this decision tree and in this section we describe what we have completed for distinguishing the numerals in each of the groups { 4, 6, 9 } and { 2, 3, 5 }. No attempt of character size normalization has been used. This may be necessary to add in at a later time. The work has been mainly concerned with understanding the shape characteristics of the numerals in morphological terms and has not reached the stage where we have begun evaluation. The methodology we are following is to do all our design work on the training set and when the design has been completed, we will then evaluate on a test set which has not been worked with during the design phase. So at this point we are not able to discuss meaningful error rate characteristics except what we have on the training set.

3.1 Morphological Sequences for Extracting Features

In this section we describe the morphological sequences for extracting features that characterize shapes of characters. Some of the primitive features we have looked at are are stems, blobs, vertical bars, and concavities.

3.1.1 Blobs

Blobs are formed in the image when loops or a configuration of pen strokes get connected into an "ink blot" when we close the image with a big structuring element. Thus if the image of the character is I, then the blob can be formed by the operation:

$$I_{blob} = I \bullet K$$

where K is a structuring element big enough to connect the strokes. For example, if we have the image of numeral nine, the loop of the nine can be morphologically closed by a structuring element bigger that the diameter of the loop. The loop will thus get filled and form a blob.

3.1.2 Stems

Stems are the ascending or descending lines in a character. It is the part of the line that is not also a part of a blob. For example, the numeral nine has a loop and a near vertical line at its right. The part of the line that extends below the loop is the stem. Similarly, the numeral six has a stem that extend above a loop.

Stems can be extracted by subtracting the blob portion from the character. Care must be taken while doing the closing for connecting up the blob. It must be done in a manner so that the concavity between the stem and the blob does not get filled. In the case of the numeral nine we used the following sequence to get the stems:

$$I_a = I \bullet K_{wedge} \quad (1)$$
$$I_b = I_a \circ K_{disk} \quad (2)$$
$$I_c = I_b \oplus K_{square} \quad (3)$$
$$I_{stem} = I_c^C \cap I \; . \quad (4)$$

Here the structuring element K_{wedge} is a wedge that fits into the cavity between the stem and lower part of the loop but does not fit into the gap in the north east corner of the loop. The structuring element K_{disk} is a disk with diameter bigger than the stem thickness. The structuring element K_{square} is a square with side equal to the diameter of the disk used in the opening. The origin of the square should be at the bottom one third of the height of the square so that the small tails that sometimes appear in the numeral 6 does not confuse the algorithm and thereby make a 6 get detected as 9.

3.1.3 Vertical Bars

Vertical bars are detected by opening the image with one pixel wide lines oriented at near vertical angles and then taking a union of the results. In our experiments we used lines of length 21 pixels and orientations ranging from 50 degrees to 110 degrees, measured counterclockwise from the horizontal axis, in steps of 10 degrees. The difference between vertical bars and stems is that stems are the portions of the vertical bars that extend above or below the blobs.

$$I_{VertBars} = (I \circ K_{\theta_1}) \cup (I \circ K_{\theta_2}) \cup \cdots \cup (I \circ K_{\theta_n})$$

where K_{θ_i} are the line structuring elements at various near vertical orientation.

3.1.4 Concavities

Concavities are regions of the background that are enclosed by the foreground in three of the north, west, south and east directions. The east, west, north and south concavities can be found using the following relations:

$$\text{NORTH-CAVITY} = (I \oplus N) \cap \overline{(I \oplus S)} \cap (I \oplus E) \cap (I \oplus W) \cap \overline{I} \quad (5)$$

$$\begin{aligned}\text{SOUTH-CAVITY} &= \overline{(I \oplus N)} \cap (I \oplus S) \cap (I \oplus E) \cap (I \oplus W) \cap \overline{I} \quad (6)\\ \text{EAST-CAVITY} &= (I \oplus N) \cap (I \oplus S) \cap (I \oplus E) \cap \overline{(I \oplus W)} \cap \overline{I} \quad (7)\\ \text{WEST-CAVITY} &= (I \oplus N) \cap (I \oplus S) \cap \overline{(I \oplus E)} \cap (I \oplus W) \cap \overline{I}\,. \quad (8)\end{aligned}$$

Here E, W, N, and S are line structuring elements. N is a vertical line with the the origin at the bottom, S is also a vertical line structuring element but with the origin at the top of the line. E and W are horizontal line structuring elements. E has its origin at the left end of the line and W has its origin at the right end of the line. Further details can be found in [MG89].

3.2 Group I: {4, 6, and 9}

In this subsection we analyze the characteristic features of the numerals belonging to Group I – 4, 6, and 9 – and how to recognize them. The most salient features of the numerals belonging to Group I are their stems, concavities facing the north direction, and "blobs". To distinguish among them, we notice that:

- Numerals 4, 6, and 9 have stems, vertical bars, blobs, and north concavities.
- Numeral 4 has a north concavity or a blob to the west of its vertical bar.
- Numeral 6 has a blob to the southeast of its stem.
- Numeral 9 has a blob or a north concavity to the northwest of its stem.
- Numeral 6 has a north concavity to the east of its stem.
- Numeral 4 has a north concavity to the west of its stem.

Hence, the following algorithm can be used to recognize a numeral from Group I:

Group I Algorithm:

1. **Find** stems.

2. **Find** blobs.
3. **Find** vertical bars.
4. **Find** north concavities.
5. **Dilate** the stems in the northwest direction.
6. **If** the dilated stems intersect the blobs,
   (a) **If** the vertical bars dilated west intersect a north concavity, recognize the numeral as 4.
   (b) **Else** recognize it as 9.
7. **Else:**
   (a) **If** the vertical bars dilated west intersect a north concavity, recognize the numeral as 4.
   (b) **Else** recognize it as 6.

3.3 Group II: {2, 3 and 5}

The three numerals belonging to Group II have as characteristic features concavities facing to the east and west directions. To distinguish among them, we notice the following facts:

- Numerals 2, 3 and 5 have one east concavity.
- Numerals 2 and 5 have one west concavity.
- Numeral 3 has two west concavities, close to each other, that are located one on top of the other.
- Numeral 2 east concavity is in the southeast direction with respect to its west concavity.
- Numeral 3 east concavity is in the southeast direction with respect to its top west concavity.
- Numeral 3 east concavity is in the northeast direction with respect to its bottom west concavity.

- Numeral 5 east concavity is in the northeast direction with respect to its west concavity.

From the above, given an image with a numeral belonging to the Group II, the numeral can be recognized using the following sequence:

Group II Algorithm:

1. **Find** east concavities.
2. **Find** west concavities.
3. **Dilate** the east concavities in the northwest direction.
4. **If** the dilated east concavities do not intersect a west concavity, recognize the numeral as 5.
5. **Else**:
   (a) **Dilate** the east concavities in the southwest direction.
   (b) **If** the dilated east concavities do no intersect a west concavity, recognize the numeral as 2.
   (c) **Else**, recognize it as 3.

4 Implementation Details

The algorithms were tried out on a data base having hand written numerals zero through nine obtained from C. Y. Suen. There were 200 samples of each numeral. These samples were of isolated numerals and were stored in a runlength encoded form. The characters were converted into binary raster scan formats and stored in matrices of size 65×60. The 200 matrices thus obtained were put in one image with 20 matrices in one row and 10 such rows. Thus the final image was of size 650×1200.

The structuring elements required for the morphological operations in various steps of the algorithms were judiciously chosen in order to accomplish the task specification of that step.

The morphological image processing was done using the GIPSY image processing software package. The machine used was Sun 4 running UNIX operating system.

5 Results and Discussion

For the first group of characters we could complete and initial set of experiments. In figure 1 we give the sequence of morphological operations performed on for the group I numerals. The processing shown is for the numeral nine. Figure 1(b) is obtained by closing Figure 1(a) with a wedge pointing south west. On opening Figure 1(b) with a disk of diameter 9 we get Figure 1(c). Notice that there is no stem in the image. Now Figure 1(c) is dilated with a square such that it covers the loop of nine completely. The result is shown in Figure 1(d). In Figure 1(e) we get the stems by intersecting Figure 1(a) with the complement of Figure 1(d). To check if the blob is northwest to the stem, we first dilate the stem with a horizontal line 10 pixels in length and 10 pixels above the origin so that it is shifted up and smeared. This is shown in Figure 1(f). This takes care of the tolerance needed because of the variability in the positions of the stem and the loop. The shifted stem in Figure 1(f) is then intersected with Figure 1(c). A non-empty intersection says that there is a blob to the northwest direction of the stem.

For Group I we get the following result for the training data set: Two hundred samples of 4, 6, and 9 each were considered.

Numeral 4: 186 were recognized correctly as 4. Four were classified wrongly as 6 and ten others were recognized as 9.

Numeral 6: 189 were recognized correctly as 6, and eleven were recognized as 4.

Numeral 9: 191 were recognized correctly as 9, one was recognized as 6 and eight others were recognized as 4.

The group II outputs have not been tabulated at this time.

It is important to note that the recognition procedure uses information in in the foreground as well as in the background. For example, for the numeral three we can either try and detect two semi-circular curves in the foreground or three wedges in the background due to the foreground.

There are two variables in the algorithm: 1) the algorithm itself, i.e., the exact sequence of operations that should be executed to accomplish the detection of a particular numeral. There is no unique algorithm for recognizing a numeral. Nevertheless, two equivalent sequences might differ in the

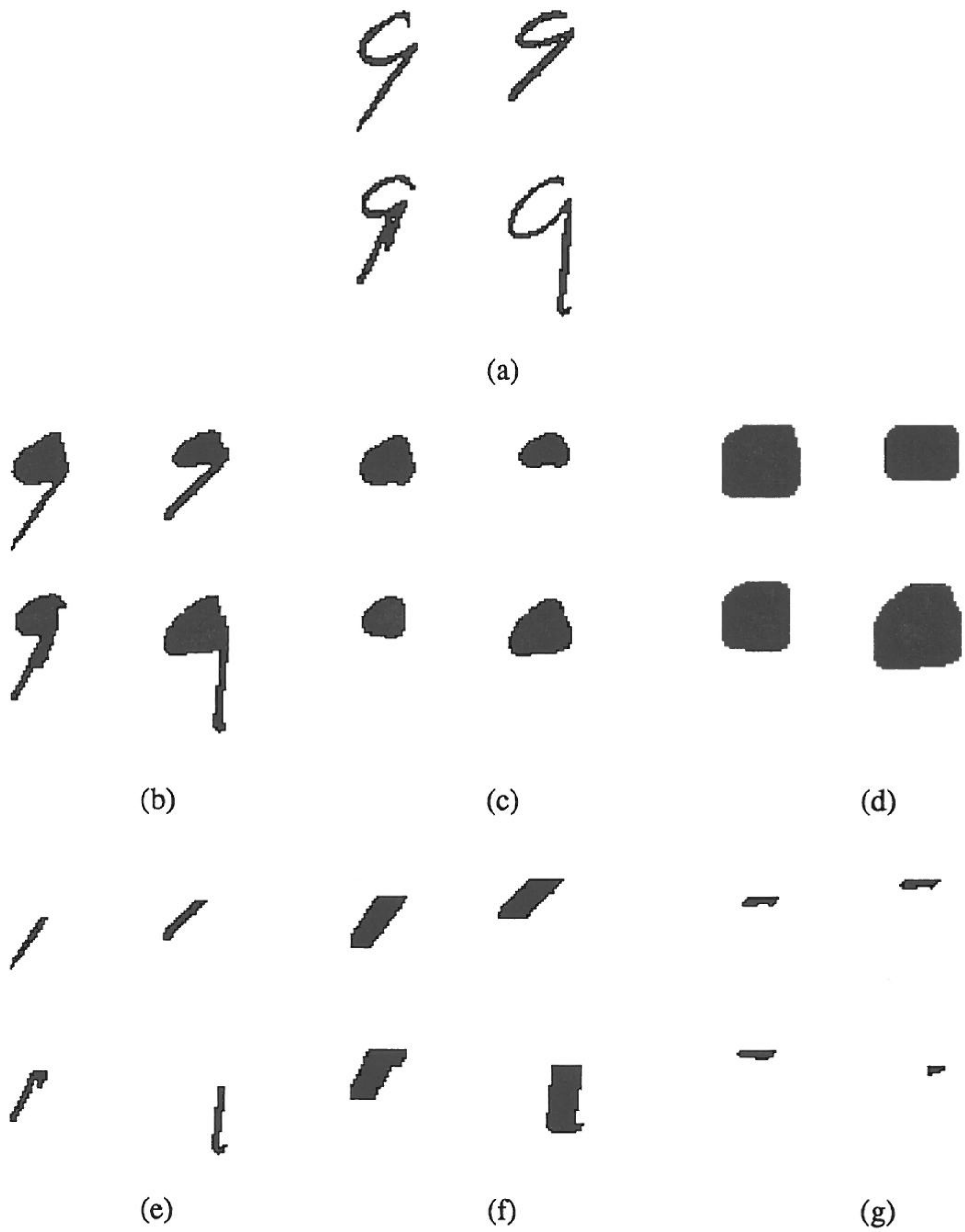

Figure 1: Morphological processing at the node for Group I. Processing is shown for numeral 9.

computation time required by orders of magnitude. A typical example is structuring element decomposition. 2) Once the morphological sequence has been fixed, the size and shape of the structuring elements need to be tuned to achieve better performance. The performance of the algorithm depends greatly upon the choice of the structuring elements. If they are not chosen properly, the results can be very arbitrary and far from what the algorithm is supposed to achieve.

Automatic learning of these algorithms and structuring elements is certainly an important issue which needs to be probed into. Very little work has been done in this direction. Some of the existing literature that address this issue are [Vog88] and [JHS90].

6 Conclusion

We have reported on work in progress for the design of a morphological decision tree solution to the recognition of handwritten numerals. Algorithms for distinguishing between 4, 6, and 9 were described giving an accuracy of just short of 95% on the training set. No results on the testing set can be reported at this time since our methodology prohibits us from applying the algorithm on the test set until our design is complete for all numerals.

References

[HSZ87] R.M. Haralick, S.R. Sternberg, and X. Zhuang. Image analysis using mathematical morphology. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI–9(4):532–550, July 1987.

[JHS90] H. Joo, R.M. Haralick, and L.G. Shapiro. Toward the automatic generation of mathematical morphology procedures using predicate logic. In *Proceedings of the ICCV*, 1990.

[MG89] B.T. Mitchell and A.M. Gillies. A model-based computer vision system for recognizing handwritten zip codes. *Machine Vision Applications*, 2:231–243, 1989.

[TSW90] C.C. Tappert, C.Y. Suen, and T. Wakahara. The state of the art in on-line handwriting recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(4):787–808, August 1990.

[Vog88] R. Vogt. *Automatic generation of morphological set recognition algorithms*. PhD thesis, University of Michigan, 1988.