Tools for Automatic Recognition of Character Strings in Maps

Line Eikvil Kjersti Aas Marit Holden
Norwegian Computing Center, P.O. Box 114 Blindern, N-0314 Oslo, Norway
e-mail: Line.Eikvil@nr.no Tel: (+47) 22 85 25 00 Fax: (+47) 22 69 76 60

Abstract. *This paper describes tools for character string recognition on maps. Single character recognition is performed using elliptical Fourier descriptors applying a statistical classifier. The recognized characters are grouped into strings, and the syntax of these strings are then analysed to detect and correct errors. As training of the classifier is essential, tools for manual and automatic training and updating are included.*

1 Introduction

The use of geographical information systems is increasing, and for such systems efficient acquisition of cartographic data is crucial. Often the information is contained in paper-based maps. Manual digitizing of maps is a costly and tedious process, and automation is therefore desirable. Automatic map conversion includes both extraction of lines and symbol recognition. The system described here performs both tasks, but this paper focuses on the recognition phase.

Symbol recognition in maps poses a more complex problem from that of traditional OCR. The symbols may be intermixed with graphics, printed at varying angles with several fonts or handwritten. We have used a feature based approach, applying a statistical classifier. In [1] methods based on transformations or series expansions are said to be robust to rotation, style variation, and distortions. We have therefore used features based on the Fourier expansion [2] [3] which have shown good results [4].

In a complete recognition system efficient tools for training are necessary. In this system we have included tools for both training and automatic updating of the statistical descriptions. Using these tools it is possible to obtain good class descriptions from an initial manually obtained training set of minimal size.

The recognition of character strings is performed stepwise. First the single characters are classified based on class descriptions obtained during a training phase. Then, the symbols are grouped to obtain informative strings. Finally, syntax analysis is used to check the strings against a grammar defining legal numbers and words.

2 Recognition of single raster symbols

From the binary image of the map, the symbols are first separated from other structures (sec. 2.1). Next, the characteristic features are extracted (sec. 2.2). Based on the features the symbols are classified (sec. 2.4), and at the same time the rotation of the single symbols can be determined (sec. 2.3). The statistical approach also requires a training phase where the extracted features are used to obtain a statistical description for each class (sec. 2.5).

2.1 Segmentation

During segmentation, the connected components of foreground pixels are extracted. From the contour of these connected components, certain parameters are analysed to determine whether this may be a symbol candidate. The technique is simple, but problems may occur if symbols touch or if they are fragmented. The problem of symbols touching lines may be eliminated by performing line extraction prior to recognition.

2.2 Feature extraction

The features are based on the Fourier expansion of the contour of the symbols, and can easily be made invariant to scale, shift and rotation. The coefficients of each term n of the Fourier expansion will be denoted (a_n, b_n, c_n, d_n). Two methods for deriving features from the Fourier coeffecients are presented below. The features are robust to noise and style variations, but can be sensitive to deformations of the contours.

Lin & Hwang's method. Lin and Hwang [3] presented the following set of descriptors :

$$\begin{aligned} I_n &= a_n^2 + b_n^2 + c_n^2 + d_n^2 \\ J_n &= a_n d_n - c_n b_n \\ K_{mn} &= (a_m^2 + b_m^2)(a_n^2 + b_n^2) + (c_m^2 + d_m^2)(c_n^2 + d_n^2) + 2(a_m c_m + b_m d_m)(a_n c_n + b_n d_n) \end{aligned} \tag{1}$$

These descriptors are independent of rotation, and can be made independent of scale by diving the I- and J-terms by J_1 and the K-terms by J_1^2. K_{mn} defines the relationship between the ellipse resulting from term m and n. We have used K_{mn} with $m = 1$. K_{mn} is always positive, and to be able to separate a symbol from its reflection, K_{1n} should have different sign dependent of the sign of θ_{1n}. (θ_{1n} is the difference in angle between the 1'st and n'th ellipse). This is obtained by computing a sign function:

$$\text{SIGN} = (a_n c_n + b_n d_n)(c_1^2 + d_1^2 - a_1^2 - b_1^2) + (a_1 c_1 + b_1 d_1)(c_n^2 + d_n^2 - a_n^2 - b_n^2) \tag{2}$$

The sign of K_{1n} is determined by SIGN. If the ellipse corresponding to the first or the nth term of the expansion is circular, the expression above is zero and K_{1n} is undefined.

Kuhl & Giardina's method. Kuhl and Giardina [2] derive features independent of the starting point on the contour as follows:

$$\begin{bmatrix} a_n^* & c_n^* \\ b_n^* & d_n^* \end{bmatrix} = \begin{bmatrix} \cos n\theta_1 & \sin n\theta_1 \\ -\sin n\theta_1 & \cos n\theta_1 \end{bmatrix} \begin{bmatrix} a_n & c_n \\ b_n & d_n \end{bmatrix} \qquad \theta_1 = \tfrac{1}{2} \arctan[\tfrac{2(a_1 b_1 + c_1 d_1)}{a_1^2 + c_1^2 - b_1^2 - d_1^2}] \tag{3}$$

The descriptors, a_n^*, b_n^*, c_n^* and d_n^*, can be made independent of rotation by:

$$\begin{bmatrix} a_n^{**} & b_n^{**} \\ c_n^{**} & d_n^{**} \end{bmatrix} = \begin{bmatrix} \cos \psi_1 & \sin \psi_1 \\ -\sin \psi_1 & \cos \psi_1 \end{bmatrix} \begin{bmatrix} a_n^* & b_n^* \\ c_n^* & d_n^* \end{bmatrix} \qquad \psi_1 = \arctan \left[\frac{c_1^*}{a_1^*} \right] \tag{4}$$

ψ_1, is the angle of the semimajor axis of the 1'st ellipse. These descriptors can be made independent of scale by dividing each term by the magnitude of the semimajor axis.

The equation for ψ_1 has two equivalent solutions, which give rise to different values. To uniquely determine ψ_1, we require it to always be positive. If it is negative, we add π radians to the original θ_1 and repeat the computations above. This method may fail for symbols where the angle of the semimajor axis varies around 0 or π radians.

If the first ellipse is circular, the equation for θ_1 cannot be solved, and an alternative approach for determination of ψ_1 suggested in [2] is used. However, this approach will fail if the symbol itself is circular. Moreover, in case of handprinted characters, the first ellipse may be circular for some symbols and not for other symbols of the same class, giving a non-uniform determination of ψ_1.

2.3 Computation of rotation angle

On maps there is usually not a general orientation for the symbols and text strings, and the orientation must be found separately for each symbol. We have here used a method for determining this rotation angle which makes it possible to compute the angle and the descriptors (Kuhl and Giardina's) simultaneously [5]. The rotation angle, ω, of a symbol is here determined as:

$$\omega = \psi - \bar{\psi} \tag{5}$$

where ψ is the orientation of the semimajor axis of the first ellipse and $\bar{\psi}$ is the mean of the axis angle computed for non-rotated symbols of known class during training. ψ is given by equation (4) while $\bar{\psi}$ has to be determined from the training symbols.

The problem of non-uniquely determination of the axis angle, can here be solved for the training case by always choosing the one of the two axis-angles which is closest to that of the previous symbol of each class. The classification may then later be performed for both the two possible axis-angles, choosing the axis angle giving the largest probability. Still, the computation of rotation angle will fail if the first ellipse is circular. However, if the symbol itself is circular, the rotation has no meaning.

2.4 Classification

For the classification we have used Bayes' classifier [6], which assigns a feature vector y to the class c which maximizes the posterior probability:

$$P(c|y) = \frac{\pi_c f_c(y)}{\sum_{k=1}^{C} \pi_k f_k(y)} \tag{6}$$

Here C is the number of classes, π_c is the prior probability density for class c, and $f_c(y)$ is the probability of y given that it belongs to class c. We assume $f_c(y)$ to be a Gaussian distribution:

$$f_c(y) = \frac{1}{(2\pi)^{d/2} \mid \Sigma_c \mid^{1/2}} e^{-\frac{1}{2}(y-\mu_c)' \Sigma_c^{-1}(y-\mu_c)} \tag{7}$$

with covariance matrix Σ_c and mean vector μ_c. d is the size of the feature vector.

If $P(c|y)$ is not comparatively large for any class, the symbol is classified as **doubt**. If $f_c(y)$ is very small for all classes, the symbol is classified as **outlier**. Doubt may occur for very similar classes, while outliers may occur for non-symbols wrongly accepted as symbol candidates.

Figure 2.4 shows the result when Kuhl and Giardina's rotation invariant descriptors are applied to a part of a naval chart with handwritten depths. Ten features from the first three terms of the Fourier expansion were used. The sign @ indicates symbols classified as outliers. Numbers which were connected to lines or other symbols were not passed to the recognition stage. Also, severly fragmented symbols were lost during segmentation. These problems may be avoided by using a more sophisticated binarization or segmentation technique. Symbols connected to depth curves could have been avoided by performing linefollowing with removal of the underlying raster prior to recognition. Of the symbols passed to the recognition stage, 92.5% were correctly classified. A further inspection of the misclassified symbols (see fig. 2.4) revealed that the digits' contours were all severly distorted by noise.

2.5 Training and updating

The Bayes' classifier for Gaussian classes is specified completely by the mean vectors and covariance matrices, $\mu_1, \Sigma_1, \ldots, \mu_C, \Sigma_C$, which can be estimated through training.

Traditional training Here, the situation is that a training set $y = \{y_j^c; j = 1, \ldots, m_c, c = 1, \ldots, C\}$ is available, where the feature vectors y_j^c are obtained from symbols manually labelled with the correct class. By maximizing the likelihood function the maximum likelihood estimates $\hat{\mu}_1, ..., \hat{\mu}_C$ and $\hat{\Sigma}_1, ...\hat{\Sigma}_C$ can be obtained from the training set.

The performance of the Bayes' classifier improves with more training data, but the manual labelling process is timeconsuming, and the operator is not necessarily unbiased when selecting the symbols. However, it is usually inexpensive to get hold of a large number of unlabelled feature vectors. The next section considers the potential of using unknown symbols for updating the maximum likelihood estimates, $\hat{\mu_1}, .., \hat{\mu_c}$ and $\hat{\Sigma_1}, .., \hat{\Sigma_c}$.

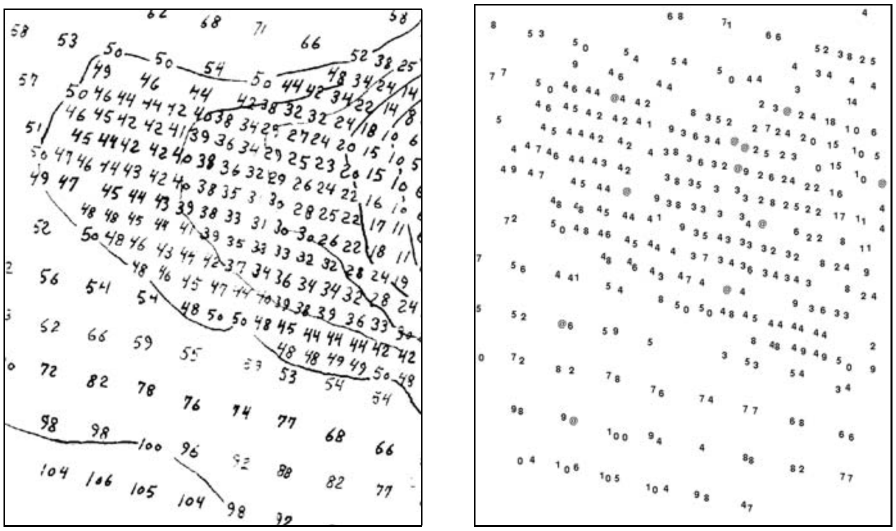

Fig. 1. *Handwritten depths (left) and the result from classification (right).*

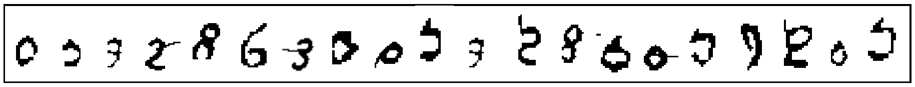

Fig. 2. *Difficult symbols: 0,5,9,2,8,6,3,0,0,5,9,2,8,6,0,3,9,2,0,5.*

Automatic updating of parameter estimates. The maximum likelihood estimates from the previous section may be updated using feature vectors of unclassified symbols. Here we assume that in addition to the feature vectors, $y = \{y_j^c; j = 1, \ldots, m_c; c = 1, \ldots, C\}$, with known class labels, another set of unlabelled feature vectors $x = \{x_i; i = 1, \ldots, N\}$ is available. A feature vector x_i of unknown class is assumed to follow the mixture distribution: $f(x_i) = \pi_1 f_1(x_i; \mu_1, \Sigma_1) + \cdots + \pi_C f_C(x_i; \mu_C, \Sigma_C)$ The simultaneous likelihood of the two sets y and x is given by:

$$\prod_{c=1}^{C}\prod_{j=1}^{m_c} f_c(y_j^c; \mu_c, \Sigma_c) \prod_{i=1}^{N} \left\{ \sum_{c=1}^{C} \pi_c f_c(x_i; \mu_c, \Sigma_c) \right\} \tag{8}$$

Because the class labels of the feature vectors in x are unknown, the maximum likelihood estimates must be determined iteratively. This may be done using the EM-algorithm [7]. The EM equations for the case where maximum likelihood estimates are to be updated from feature vectors with unknown class, may be considered as natural generalisations of the EM equations for the case where feature vectors with known class memberships are available [6]. The necessary equations can be found in [8]. The parameter values are ensured to converge by the general theory of EM-algorithms [9] and the limiting values are the updated estimates π_c^*, μ_c^* and Σ_c^* for $c = 1, ..., C$. Different results using this approach can be found in [10] and [11].

3 Grouping and Syntax Analysis

It is usually desireable to group the single symbols resulting from the recognition, into words and numbers. This is done based on the symbols' location in the image which is defined through the coordinates of their bounding box. The ordering of the symbols is done under the assumption that symbols in a string are always ordered from left to right. For vertically orientated strings, the symbols are assumed to be ordered from top to bottom. The result of the grouping is a set of symbol strings.

For the syntax analysis a grammar is defined based on the syntax directed translation schema (SDTS), described in [12] and used for character recognition in [13]. The grammar consists of syntax rules for a set of string types, where a string is defined by an ordered set of substrings. The rules for a substring define the length and the set of legal basic symbols (alphabet) for the substring. In addition to the alphabet, the possible translations for the symbols are defined. The length of a substring may vary over an interval, specified by a minimum and maximum length. A minimum length of zero indicates that the substring may be skipped. A maximum length of zero, flags that an extra symbol should be inserted.

All symbols are assigned a list of possible symbol classes, sorted on probability, during classification. For each symbol, the probability is found as that of the most probable *legal* class in the list. This means that if the most probable class, is not a legal symbol, it will not be considered. If the symbol class itself is not legal, but has a translation which is legal, the translation is considered. If none of the classes in the list is legal, the probability of the symbol is zero. In this way, the syntax analysis may detect and correct classification errors. In addition, symbols like points and commas which tend to disappear during digitization, can be inserted.

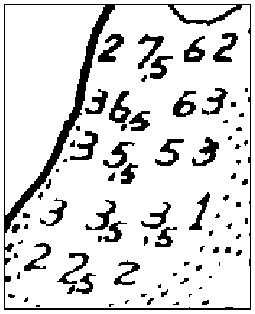

Fig. 3. *A part of a naval chart with handwritten depths.*

Figure 3 shows a part of a naval chart containing handwritten depth values. To avoid selecting the dots indicating shallow waters as symbol candidates, the limits for the smallest symbols were set larger than these dots. However, then the decimal points were lost as well. To be able to correct these errors, we defined a grammar containing definitions of two types of strings; single numbers and decimal numbers. We knew the depths to be below 10 meters and the accuracy of the measurements half a meter. Hence, we defined the decimal numbers to consist of three parts; first a digit between 0 and 9, then a decimal point and finally the decimal '5'. For single numbers the legal classes were the numbers from 1 to 9. Applying these rules, all the numbers were correctly classified and the commas were inserted correctly.

4 Summary and conclusions

This paper presented different tools for symbol recognition in maps, including single symbol recognition, parameter estimation, grouping and syntax analysis. Symbol candidates are first segmented from the background. Contour based features are extracted and the symbol is classified based on previously obtained class descriptions. Single symbols may be grouped into strings, and a final syntax analysis allows for detection and correction of classification errors.

Combined with the tools for automatic updating of class descriptions, the symbol recognition provides a flexible and powerful tool for map recognition. However, symbols that are severly fragmented or connected to other symbols or linework, are not recognized. The most efficient way of solving this problem is to use more sophisticated methods for binarization, increasing the quality of the binary raster. When this is not possible, methods for separating connected elements must be usd.

The features used in the system are robust to rotation and style variation. However, they are sensitive to broken contours, and they are also unable to distinguish symbols, differing only in the shape of the inner contour. A combination of different features may solve these problems.

Acknowledgements The Norwegian Research Council (NFR) has supported this work.

References

1. R.H. Davis & J. Lyall: *Recognition of Handwritten Characters – a Review.* Image and Vision Computing, Vol. 4, No. 4, pp. 208–218, 1986.
2. F.P. Kuhl & C.R. Giardina: *Elliptic Fourier Features of a Closed Contour.* Computer Graphics and Image Processing, **18**, pp. 236–258, 1981.
3. C-S. Lin & C-L. Hwang: *New Forms of Shape Invariants from Elliptic Fourier Descriptors.* Pattern Recognition, Vol. 20, No. 5, pp. 535–545, 1987.
4. T. Taxt, J.B. Ólafsdóttir & M. Dæhlen: *Recognition of Handwritten Symbols.* Pattern Recognition, Vol. 23, No. 11, pp. 1155–1166, 1990.
5. L. Eikvil & T. Taxt: *Simultaneous Recognition of Class and Rotation Angle of Raster Symbols.* Proceedings SCIA-91, Aalborg, Vol I, pp. 461–468, 1991.
6. R.O. Duda & P.E. Hart: *Pattern Classification and Scene Analysis.* John Wiley & Sons, New York, 1973.
7. A.P. Dempster, N.M. Laird & D.B. Rubin: *Maximum likelihood from incomplete data via the EM algorithm (with discussion).* Journal of Royal Statistical Society, series B, 39, pp. 1–38, 1977.
8. N.L. Hjort: *Notes on the theory of statistical symbol recognition* Technical report no. 778, ISBN 82-539-0265-4, Norwegian Computing Center, 1986.
9. F.J. Wu: *On the convergence of the EM-algorithm.* Ann. Stat., 11, pp. 95–103, 1983.
10. L. Eikvil, M. Holden & G. Storvik: *Methods for Updating of Model Parameters Applied within the Area of Symbol Recognition.* Technical report no. 859, ISBN 82-539-0352-9, Norwegian Computing Center, 1992.
11. G. Storvik, M. Holden & V. Bosnes: *Improving statistical image classification by updating model parameters using unclassified pixels.* Technical report no. 857, ISBN 82-539-0350-2, Norwegian Computing Center, 1992.
12. A.V. Aho & J.D. Ullman: *The Theory of Parsing, Translation and Compiling.* Volume 1, Prentice-Hall, 1972.
13. T. Bjerch, N.L. Hjort, H. Koren & T. Taxt: *New tools in structural and statistical symbol recognition.* Technical report no. 820, ISBN 82-539-0310-3, Norwegian Computing Center, 1990.

This article was processed using the LaTeX macro package with LLNCS style