Supervised Classification of Early Perceptual Structure in Dot Patterns*

Mihran Tuceryan and *Anil K. Jain*
Dept. of Computer Science
Michigan State University
East Lansing, MI 48824

Narendra Ahuja
Beckman Institute
University of Illinois at Urbana-Champaign
Urbana, IL 61801

Abstract

A supervised algorithm for computing perceptual groupings in dot patterns is presented. The algorithm uses shape features of the polygons in the Voronoi tessellation of the input pattern. The training patterns identified by humans are used to obtain an initial non-contextual classification which is then refined by a probabilistic relaxation labeling.

1 INTRODUCTION

This paper is concerned with grouping of simple image plane entities – points in a dot pattern. The goal is to develop a set of rules as well as a computational process that makes use of the rules for identifying groupings of dots.

To define a computational approach to perceptual grouping, it is necessary to specify what precisely is the desired output of the grouping process given the input dot pattern. This was discussed in detail in [2]. The possible perceived labels assigned to dots were *INTERIOR, BORDER, CURVE* and *ISOLATED*. The assignment of these labels was accomplished by a set of independent modules each of which had its own expertise in identifying some aspect of this grouping.

In this paper we study the module that identifies *INTERIOR* and *BORDER* dots in a given pattern. We use human expertise to label the dots as interior and border in a small training set of dot patterns, and use this information for feature selection and supervised classification. As a result of this supervised training, we have been able to systematically select the relevant features to be used in this grouping module.

2 VORONOI NEIGHBORHOOD

The "neighbors" of a dot has been defined in many ways in the literature: circular neighborhood; k-nearest neighbors and various extensions to near neighbors; the minimum spanning tree; the relative neighborhood graph and the Gabriel graph; and the Voronoi tessellation [1]. We use the Voronoi tessellation as the definition of the "neighborhood" of a point.

The Voronoi tessellation of a set of points S in a plane is a partition of the plane (R^2) into regions such that the region assigned to a point P consists of all points in R^2 which are closer to P than to any other point in S. This results in a polygonal region assigned to each point called the *Voronoi polygon* [3]. Two points are said to be *Voronoi neighbors* if the Voronoi polygons enclosing them share a common edge.

Features based on the shapes of the Voronoi polygons indicate whether a dot is *INTERIOR* or *BORDER*. A total of 23 features are initially used to characterize the geometric properties of the Voronoi polygons. This number is later reduced as a result of a feature selection process. We now summarize the features extracted from the Voronoi polygons.

(a) *Moments of area:* Let (x_i, y_i) be the coordinates of dot i and let R be the Voronoi polygon containing dot i. Then the $(p+q)$th moments of area of R is defined as $m_{pq} = \iint_R (x - x_i)^p (y - y_i)^q \, dx \, dy$. We use the first six moments of area as well as their translation, scale, and rotation invariant features as shape features for the Voronoi polygons. A total of sixteen such features are used.

(b) *Compactness:* Let α_k, $k = 1, \cdots, l$, be the angles subtended on a dot by its successive neighbors. The compactness or regularity of the Voronoi polygon is defined as *compactness* $= (\alpha_{max} - \alpha_{avg})/\pi$, where $\alpha_{avg} = \sum_{k=1}^{l} (\alpha_k - \alpha_{max})/(l-1)$ and $\alpha_{max} = \max_k \alpha_k$. The interior cells are "compact."

(c) *Elongation:* The elongation is computed by first identifying an ellipse (with major axis, a, and minor axis, b) which has the same second-order moments as the Voronoi polygon. The elongation of the Voronoi polygon is then computed as $\sqrt{1-(b/a)^2}$. We also obtain the direction of the minor axis of the ellipse.

(d) *Eccentricity:* The eccentricity measure is a scaled vector indicating how much and in which direction a dot is off the center of gravity of its Voronoi polygon.

*This research was supported in part by from Air Force Office of Scientific Research grant no. AFOSR 90-0061 and NSF grant no. CDA 8806599.

0-8186-2915-0/92 $3.00 © 1992 IEEE

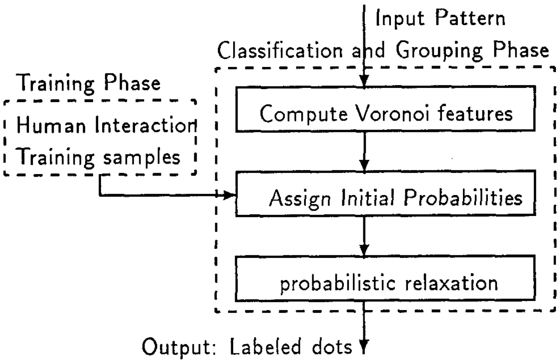

Figure 1: The steps of the algorithm to compute perceptual grouping.

3 COMPUTING THE PERCEPTUAL GROUPINGS

The flowchart of the algorithm for computing the interior and border labels to be assigned to each dot is given in Figure 1. We now describe the details of each step.

3.1 Training Phase

The training phase involves (a) collecting a set of example dot patterns in which each dot is labeled as *BORDER* or *INTERIOR*, and (b) making use of the training set in (a) for effectively assigning the perceptual role of each dot in test patterns. The collection of training samples involved showing a set of six representative dot patterns to the human trainer and recording the perceptual role assigned to each dot in the pattern. The total number of dots in this set was 217 labeled *BORDER* and 210 labeled *INTERIOR*.

The second phase of the training involved analyzing the contribution of each feature of the Voronoi cell in the classification process. A sequential forward feature selection algorithm was used. This algorithm gives an estimate of the error probability for each subset of features to be used. We took the first n features such that the error probability estimate of the set was under 0.3. This resulted in the selection of six features ($n = 6$). These top six features were: eccentricity magnitude, x direction of eccentricity vector, y direction of eccentricity vector, the elongation magnitude, the x direction of minor axis, the y direction of minor axis.

The initial classification of points into perceptual categories was done by using the nearest neighbor classification method in the feature space. Furthermore, a probability for each class was computed which was then fed to the relaxation labeling as the initial probabilities. We describe below the details of the initial probability computation and the relaxation labeling.

3.2 Relaxation Labeling

Relaxation labeling is defined by a collection of objects a_i, $i = 1, \cdots, n$, and a set of labels λ_j, $j = 1, \cdots, m$. Each object has a set of probabilities $p_i(\lambda_j)$, assigned to it that represent the likelihood of object a_i having label λ_j, where $\sum_j p_i(\lambda_j) = 1$. Each object is assigned a set of initial probabilities, $p_i^{(0)}(\lambda_j)$. Then, the probabilities are updated iteratively as follows [4]:

$$p_i^{(k+1)}(\lambda) = \frac{p_i^{(k)}(\lambda)[1 + q_i^{(k)}(\lambda)]}{\sum_\lambda p_i^{(k)}(\lambda)[1 + q_i^{(k)}(\lambda)]} \qquad (1)$$

where

$$q_i^{(k)}(\lambda) = \sum_{j \in N(i)} d_{ij} \sum_{\lambda'} r_{ij}(\lambda, \lambda') p_j(\lambda') \qquad (2)$$

The term $r_{ij}(\lambda, \lambda')$ in the Equation (2) represents the compatibility of labels λ and λ' on objects a_i and a_j, respectively. $N(i)$ is the set of neighbors for object a_i. The d_{ij}'s are weights associated with the interaction of the pair of objects a_i and a_j, where $\sum_j d_{ij} = 1$, in order to keep $0 \le p_i \le 1$. Thus $q_i^{(k)}(\lambda)$ represents the support given to the label λ at the object a_i by all its neighbors, a_j.

In our formulation in this paper, the objects a_i are the dots, the labels λ_j are either interior ($\lambda_1 = I$ for *INTERIOR*) or border in one of eight directions $0°, 45°, \cdots, 315°$ ($\lambda_2, \cdots, \lambda_9$). We have set $d_{ij} = 1/n_i$, where n_i is the number of neighbors of dot i.

3.3 Initial Probability Assignments

The initial probabilities are assigned in two steps. First, points are classified as non-directional *BORDER* or *INTERIOR* based on their Voronoi features and the training samples using the nearest neighbor classifier. Then the non-directional border classification is further refined to estimate the initial probabilities for the directional border labels.

We now describe the computation of the non-directional border classification. Let $\Omega = \{\omega_1, \omega_2\} = \{B, I\}$ be the set of class labels (*B: BORDER, I: INTERIOR*). Let $\{(\mathbf{x}_1, \theta_1), (\mathbf{x}_2, \theta_2), \cdots, (\mathbf{x}_n, \theta_n)\}$ be the set of training features. Here $\mathbf{x}_j$ is the six-dimensional feature vector for dot j and $\theta_j \in \Omega$ is the class label for $\mathbf{x}_j$ specified by the human trainer. Let $\mathbf{x}_i$ be the feature vector computed for dot i in a test pattern. The probability that dot i has the class label ω_k is given by the expression:

$$p_{NN,i}(\omega_k) = 1 - \frac{\min_{\theta_j = \omega_k} |\mathbf{x}_i - \mathbf{x}_j|}{\sum_{k=1,2} \min_{\theta_j = \omega_k} |\mathbf{x}_i - \mathbf{x}_j|} \qquad (3)$$

These probabilities are used along with the local geometric distribution of the neighbors of a point to estimate the initial probabilities of each of the nine labels. This was computed by observing that for a border label with a given orientation, the interior points would be collected on one side. The label probabilities for relaxation labeling is given by the following expressions:

$$p_i^{(0)}(I) = \frac{1}{9} p_{NN,i}(I) \qquad (4)$$

$$p_i^{(0)}(b_k) = p_{NN,i}(B).$$

$$\left[\sum_{j\in N(i),0\leq\theta\leq\pi}\left[\frac{1}{2}(1-\sin\theta)\right]p_{NN,j}(I)\right] \quad (5)$$

Here the term $\frac{1}{2}(1-\sin\theta)$ is a weight which is maximum when θ is $\pi/2$ on the right hand side of the border. The border label b_k will have a large value if the near-neighbor based border probability $p_{NN}(B)$ is high and all the neighbors on its right hand side have high probabilities of being *INTERIOR*.

Finally, these label probabilities are normalized so that the sum $\sum_\lambda p_i^{(0)}(\lambda)=1$.

3.4 Computation of Compatibility Coefficients

The reason for using the relaxation labeling is so that contextual information can be included into the classification process and that certain global constraints can be enforced. This is accomplished in the relaxation labeling process through the interaction of neighboring dots. This interaction is defined by specifying the compatibility coefficients. We now describe the computation of the compatibility coefficients.

The global constraints enforced in this problem are: (a) Borders are smooth, and (b) the likelihood that a point has *INTERIOR* label is higher if the neighboring points also have *INTERIOR* labels. In order to define the border smoothness constraint, the border label is broken into eight discrete directions: $0°, 45°, \cdots, 315°$. Therefore, each dot can have one of nine labels assigned to it. Based on these, the following are the compatibility coefficients:

$$r_{ij}(I,I) = \left[\frac{P_i(I)+P_j(I)}{2}\right] \quad (6)$$

$$r_{ij}(I,b_k) = \left[\frac{P_i(I)+P_j(B)}{2}\right]\left[\frac{1+\sin\theta}{2}\right] \quad (7)$$

$$r_{ij}(b_k,I) = \left[\frac{P_i(B)+P_j(I)}{2}\right]\left[\frac{1-\sin\theta}{2}\right] \quad (8)$$

$$\begin{aligned} r_{ij}(b_k,b_l) = {} & \frac{1}{8}\left[P_i(I)+P_j(B)\right]\cdot \\ & [2+\cos(\alpha_k-\alpha_l)\cdot \\ & (\cos 2(\alpha_k-\theta_{ij}) \\ & +\cos 2(\alpha_l+\theta_{ij}))] \quad (9) \end{aligned}$$

The $(1+\sin\theta)$ and $(1-\sin\theta)$ terms in Equations (8) and (9) enforce the right hand rule mentioned before. That is, a border label has interior labeled dots on its right hand side. The expression $[2+\cos(\alpha_k-\alpha_l)(\cos 2(\alpha_k-\theta_{ij})+\cos 2(\alpha_l+\theta_{ij}))]$ in Equation (9) is the term enforcing the smoothness along the cluster borders.

4 EXPERIMENTAL RESULTS

We have run our supervised perceptual grouping algorithm on two types of dot patterns. The first was the set of patterns which were used to select training samples. The second set of patterns were completely new dot patterns which were not used as training examples at all. Figure 2(a) shows various example dot patterns. Only the example in (i) is one of the training samples. All the remaining four are new test patterns. Figure 2(b) shows the non-contextual classification produced by the nearest neighbor method using the training data, and (c) shows the output of the relaxation labeling.

The algorithm produces acceptable results in most of the dot patterns. There are some dot patterns which do not result in good groupings (example (iv)). The main reason for this is that in this pattern the features do not have sufficient information to distinguish the border and interior points of the outer sparse cluster. This results in poor initial classification which is very difficult to correct later with the relaxation labeling.

We can also see that the relaxation labeling is cleaning up some of the misclassifications produced by the nearest neighbor classifier. For example, in example (ii) there are many dots in the interiors of the clusters which are classified as border dots. These are cleaned up and classified as interior dots as a result of the relaxation labeling.

5 CONCLUSION

This paper presents two extensions to the perceptual grouping algorithm presented in [2]. The first is the use of human training to identify the various perceptual roles that a dot can play. The second is the use of systematic feature selection techniques to identify the most relevant features in the computation of perceptual grouping.

As a result of this systematic feature selection process, we have shown that the following six features are the most useful: (a) eccentricity magnitude, (b) x direction and (c) y direction of the eccentricity vector, (d) the elongation magnitude, (e) the x direction and (f) the y direction of the major axis of the cell.

The approach of using training patterns to characterize the various perceptual roles for the dots can be used in other applications. For example, each of the modules in the algorithm presented in [2] can be implemented using this approach.

References

[1] N. Ahuja. Dot pattern processing using voronoi neighborhoods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-4(3):336–343, May 1982.

[2] N. Ahuja and M. Tuceryan. Extraction of early perceptual structure in dot patterns: Integrating region, boundary, and component gestalt. *Computer Vision, Graphics, and Image Processing*, 48:304 – 356, December 1989.

[3] F. P. Preparata and M. I. Shamos. *Computational Geometry: An Introduction*. Springer-Verlag, 1985.

[4] A. Rosenfeld, R. Hummel, and S. Zucker. Scene labeling by relaxation operations. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-6:420–433, 1976.

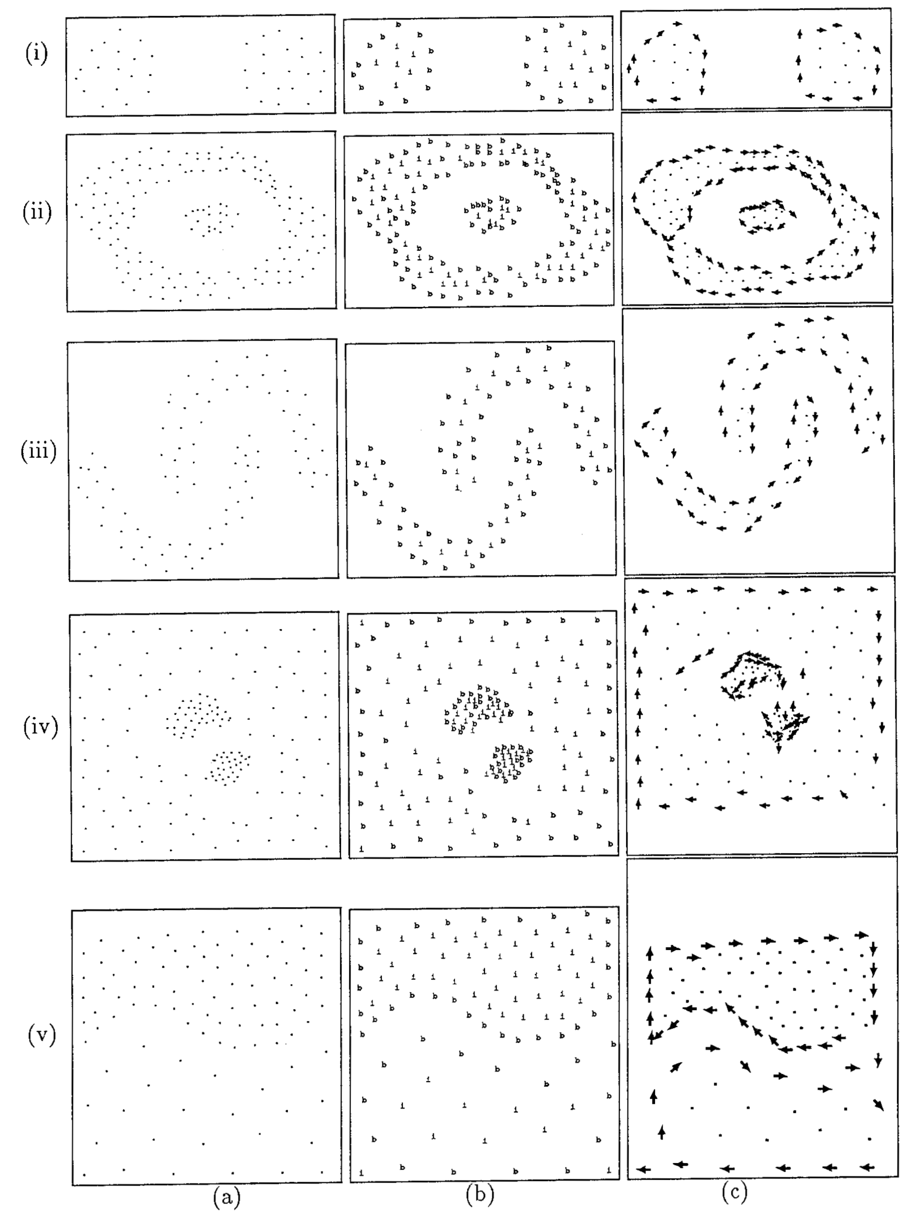

Figure 2: (a) Example dot patterns. (b) Initial nearest neighbor classification of the dots. (c) The result of relaxation labeling.