LOCATION- AND DENSITY-BASED HIERARCHICAL CLUSTERING USING SIMILARITY ANALYSIS

P. Bajcsy N. Ahuja

Beckman Institute
University of Illinois at Urbana-Champaign
405 N. Mathews Ave., Urbana, IL 61801
Tel: (217)244-4174, (217)333-1837, Fax: (217)244-8371
E-mail: peter@stereo.ai.uiuc.edu, ahuja@vision.ai.uiuc.edu

Abstract

This paper presents a new approach to hierarchical clustering of point patterns. Two algorithms for hierarchical location- and density-based clustering are developed. Each method groups points such that maximum intra-cluster similarity and inter-cluster dissimilarity are achieved for point locations or point separations. Point locations or point separations are denoted as elements. Each method starts with grouping elements into clusters $C_{\mathbf{e_i}}$ for every element $\mathbf{e_i}$ such that all elements in $C_{\mathbf{e_i}}$ are dissimilar to $\mathbf{e_i}$ by no more than an amount θ. A sample mean $\bar{\mu}_{\mathbf{e_i}}$ of all elements in $C_{\mathbf{e_i}}$ is calculated. Clusters are formed by grouping pairs of adjacent elements having similar sample means $\bar{\mu}_{\mathbf{e_i}}$. Clusters with increasing intra-cluster similarity and inter-cluster dissimilarity are identified by increasing θ continuously. Those clusters that remain unchanged over intervals of θ are selected as defining a clustering of the point pattern. The accuracy and computational requirements of the proposed methods are evaluated. Performance of the clustering methods is compared with four other methods. The approach is applied to a two-step texture analysis, where points represent centroid and average color of the regions in image segmentation. In the first step, the centroids are clustered using density-based clustering and the colors using location-based clustering. Then, the resulting texture is identified by combining clustering results and merging the regions represented by points from one final cluster into a textured region.

Index terms: point patterns, hierarchical structure, clustering.

1 Introduction

Clustering explores the inherent tendency of a point pattern to form sets of points (clusters) in multidimensional space. Most of the previous clustering methods assume tacitly that points having similar locations or constant density create a single cluster (location- or density-based clustering). Two ideal cases of these clusters are shown in Figure 1. Location or density becomes a characteristic property of a cluster. Other properties of clusters are proposed based on human perception [1, 2, 3] (Figure 2 left) or specific tasks (texture discrimination from perspective distortion [4]), e.g., points having constant directional change of density in Figure 2 right. The properties of clusters have to be specified before the clustering is performed and are usually a priori unknown.

This work presents a new approach to hierarchical clustering of point patterns. Two hierarchical clustering algorithms are developed based on the new approach. The first algorithm detects clusters with similar locations of points (location-based clustering). This method achieves identical results as centroid clustering [5, 6, 7] with slight improvement in uniqueness of solutions. The second algorithm detects clusters with similar point separations (density-based clustering). This method can create clusters with points being spatially interleaved and having dissimilar densities called transparent clusters. Figure 3 shows two transparent clusters. The detection

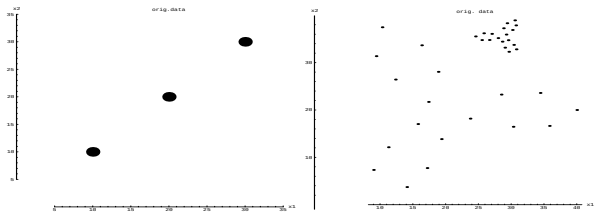

Figure 1: Ideal three (left) and two (right) clusters for location (left) and density (right) based clusterings.

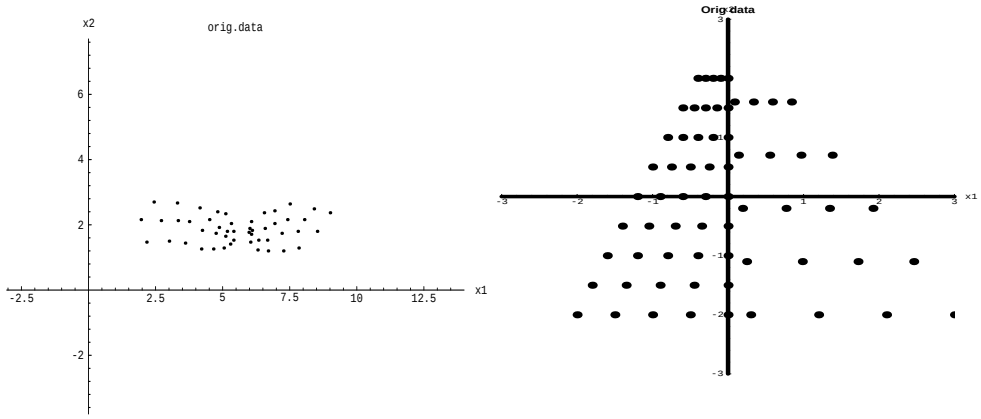

Figure 2: Illustration of other possible properties of points creating a cluster. Left - two clusters with smoothly varying nonhomogeneous densities taken from [1]. Right - two clusters with constant directional change of density.

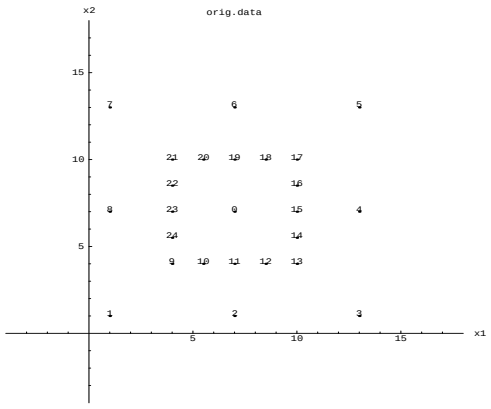

Figure 3: Two transparent clusters, $C_1 = \{0, 1, 2, .., 8\}$, $C_2 = \{9, ..., 24\}$.

of transparent clusters is a unique feature of the method among all existing clustering methods.

The two methods are developed using similarity analysis. The similarity analysis relates intra-cluster dissimilarity with inter-cluster dissimilarity. The dissimilarity of point locations or point separations is considered for clustering and is denoted in general as a dissimilarity of elements $\mathbf{e_i}$. Each method can be described as follows. First, every element $\mathbf{e_i}$ gives rise to one cluster $C_{\mathbf{e_i}}$ having elements dissimilar to $\mathbf{e_i}$ by no more than a fixed number θ. Second, a sample mean $\bar{\mu}_{\mathbf{e_i}}$ of all elements in $C_{\mathbf{e_i}}$ is calculated. Third, clusters would be formed by grouping pairs of elements if the sample

means computed at the two elements are similar. Fourth, the degree of dissimilarity θ is used to form several (multiscale) partitions of a given point pattern. A hierarchical organization of clusters within multiscale partitions is built by agglomerating clusters for increasing degree of dissimilarity. Lastly, the clusters that do not change for a large interval of θ are selected into the final partition. Experimental evaluation is conducted for synthetic point patterns, standard point patterns (8Ox - handwritten character recognition, IRIS - flower recognition) and point patterns obtained from image texture analysis. Performance of the clustering methods is compared with four other methods (partitional - FORGY, CLUSTER, hierarchical - single link, complete link [5]). Detection of clusters perceived by humans (Gestalt clusters [1]) is shown.

Location- and density-based clusterings are suitable for texture analysis. A texture is modeled as a set of uniformly distributed identical primitives [8] (see Figure 4). A primitive is described by a set of photometric, geometrical or topological features (e.g., color or shape). Spatial coordinates of a primitive are described by another set of features. Thus the point pattern obtained from texture analysis consists of two sets of features (e.g., centroid location and average color of primitives) and has to be decomposed first. Location-based clustering is used to form clusters corresponding to identical primitives in one subspace (color of primitives). Density-based clustering creates clusters corresponding to uniformly distributed primitives in the other subspace (centroid locations of primitives). The resulting texture is identified by combining clustering results in the two subspaces. This decomposition approach is also demonstrated on point patterns obtained in other application domains. In general, it is unknown how to determine the choice of subspaces. Thus an exhaustive search for the best division in terms of classification error is used in the experimental part for handwritten character recognition and taxonomy applications.

The salient features of this work are the following. First, a decomposition of the

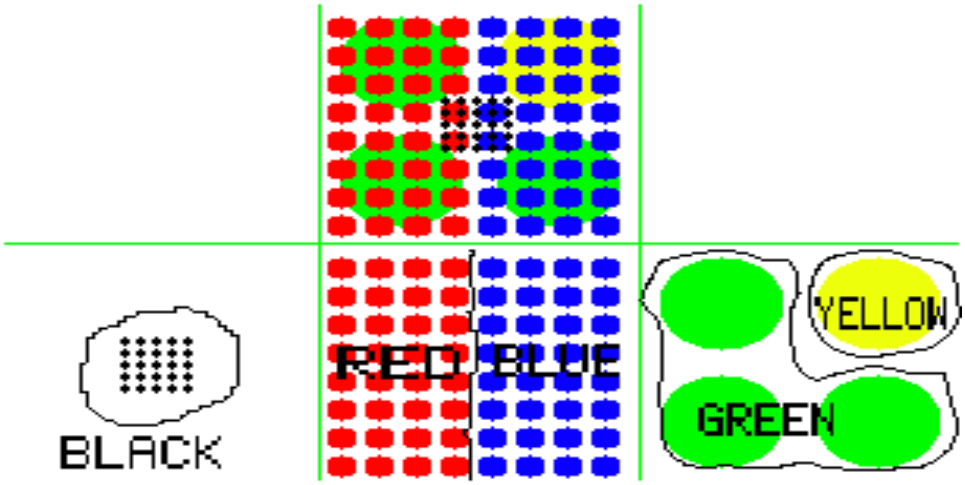

Figure 4: Example of textures.
Top - original image. Bottom - the resulting five textures (delineated by black line) obtained by location- and density-based clusterings.

clustering problem into two lower-dimensional problems is addressed. Second, a new clustering approach is proposed for detecting clusters having any constant property of points (location or density). Third, a density-based clustering method using the proposed approach separates spatially interleaved clusters having various densities, thus is unique among all existing clustering methods. The methods can be related to the graph-theoretical algorithms.

This paper is organized as follows. Section 2 provides a short overview of previous clustering methods. Theoretical development of the proposed clustering method is presented in Section 3. Analytical, numerical and experimental performance evaluations of the clustering method follow in Section 4. Section 5 presents concluding remarks.

2 Previous Work

Clustering is understood as a low-level unsupervised classification of point patterns [5, 9]. A classification method assigns every point into only one cluster without a

priori knowledge. All methods are divided into partitional and hierarchical methods. Partitional methods create a single partition of points while hierarchical methods give rise to several partitions of points that are nested.

Partitional clustering methods can be subdivided roughly into (1) error-square clusterings [5, 6], (2) clustering by graph theory [10, 2, 5, 3]) and (3) density estimation clusterings [7, 5, 11, 6]. Error-square clusterings minimize the square error for a fixed number of clusters. These methods require to input the number of sought clusters as well as the seeds for initial cluster centroids. Comparative analysis in this work is performed using the implementations of error-square clusterings called FORGY and CLUSTER [5]. FORGY uses only one K-means pass, where K is a given number of clusters in the final partition. CLUSTER uses the K-means pass followed by a forcing pass. During the forcing pass all mergers of the clusters obtained from the K-means pass are performed until the minimum square error is achieved. Clustering by graph theory uses geometric structures such as, minimum spanning tree (MST), relative neighborhood graph, Gabriel graph and Delaunay triangulation. The methods using these geometric structures construct the graph first, followed by removal of inconsistent edges of the graph. Inconsistent edges and how to remove edges are specified for each method. Due to computational difficulties, only methods using MST are used for higher than three dimensional point patterns. Density estimation clusterings have used the two approaches: (a) count the number of points within a fixed volume (mode analysis, Parsen window), (b) calculate the volume for a fixed number of points (k-nearest neighbors). These methods vary in their estimations (Parsen, Rosenblatt, Loftsgaarden and Quesenberry [11]).

Two most commonly used hierarchical clusterings are single-link and complete-link methods [12, 5]. Both methods are based on graph theory. Every point represents a node in a graph and two nodes are connected with a link. A length of a link is

computed as the Euclidean distance between two points. Single- and complete-link clusterings begin with individual points in separate clusters. Next, all links having smaller length than a fixed threshold create a threshold graph. The single link method redefines current clustering if the number of maximally connected subgraphs of the threshold graph is less than the number of current clusters. The complete-link method does the same for maximally complete subgraphs. A connected subgraph is defined as any graph that connects all nodes (corresponding to points). A complete subgraph is any connected subgraph that has at least one link for all pairs of nodes (points). The implementations of single-link and complete-link clusterings based on Hubert's and Johnson's algorithms [5] are used for the comparative analysis in this work.

The use of clustering methods can be found in many applications related to remote sensing [13, 14, 15], image texture detection [16], taxonomy [12, 17], geography [18, 19] and so on. The objective of this work is to contribute to (1) the theoretical development of non-existing clustering methods and (2) the use of clustering for texture detection.

3 Location- and Density-Based Clusterings

First, a mathematical framework is established in Section 3.1. The clustering method is proposed in Section 3.2. The algorithms for hierarchical location- and density-based clusterings are outlined in Section 3.3 and related methods to the proposed ones are compared in Section 3.4.

3.1 Mathematical formulation

An n-dimensional (nD) point pattern is defined as a set of points $I = \{\mathbf{p_i}\}_{i=1}^{P}$ with coordinates $(p_1, p_2, \cdots, p_n)$. A general goal of unsupervised clustering is to partition a set of points I into non-overlapping subsets of points $\{C_j\}_{j=1}^{N}$; $I = \cup_{j=1}^{N} C_j$, $C_k \cap C_j = 0$

and $C_j = \{\mathbf{p_i}\}_{i \in W_j}$, where W_j is an index set from all integer numbers in the interval $[1, P]$. The subsets of points are called clusters and are characterized in this work by the similarity (dissimilarity) of point locations or point separations. A notion of an element $\mathbf{e_i} \in \Re^n$ (a feature vector) is introduced to refer either to a point location $\mathbf{p_i}$ or a point separation $d(l_{\mathbf{p_1},\mathbf{p_2}}) = \parallel \mathbf{p_1} - \mathbf{p_2} \parallel$ (the Euclidean distance between two points also called length of a link $d(l_{\mathbf{p_1},\mathbf{p_2}})$).

In general, every cluster of elements $C_j = \{\mathbf{e_k}\}$ can be characterized by its maximum intra-cluster dissimilarity $\theta = \max\{\parallel (\mathbf{e_i} \in C_j) - (\mathbf{e_k} \in C_j) \parallel\}$ (or minimum intra-cluster similarity θ) and minimum inter-cluster dissimilarity $\alpha = \min\{\parallel (\mathbf{e_i} \notin C_j) - (\mathbf{e_k} \in C_j) \parallel\}$ (maximum inter-cluster similarity α), where the dissimilarity (similarity) value of any two elements $\mathbf{e_i}, \mathbf{e_k}$ is defined as the Euclidean distance $\theta_{\mathbf{e_i},\mathbf{e_k}} = \parallel \mathbf{e_i} - \mathbf{e_k} \parallel$. Figures 5 and 6 show a cluster of points CF_j characterized by $\theta = \delta$ and $\alpha = \alpha_f$ and a cluster of point separations (links) CL_j characterized by $\theta = \varepsilon$ and $\alpha = \alpha_s$. One would like to obtain clusters with a minimum intra-cluster and maximum inter-cluster dissimilarity (maximum intra-cluster and minimum inter-cluster similarity) in order to decrease the probability of misclassification. Thus our goal is to partition a point pattern I into nonoverlapping clusters $\{C_j\}_{j=1}^{N}$ having a minimum intra-cluster and maximum inter-cluster dissimilarity of elements.

If clusters of elements are not clusters of points as in the case point separations then a mapping from the clusters obtained to clusters of points is performed. The mapping from clusters of point separations (links) to clusters of points takes two steps: (1) Construct a minimum spanning tree from the average values of individual clusters of links. (2) Form clusters of points sequentially from clusters of links in the order given by the minimum spanning tree (from smaller to larger average values of clusters).

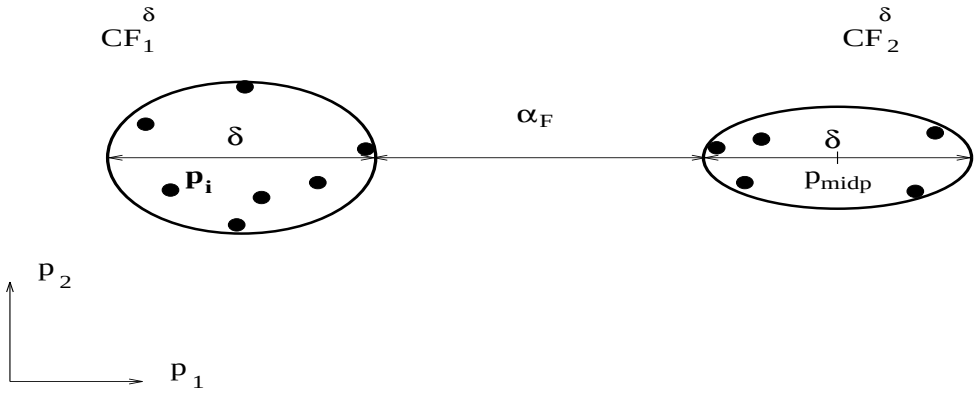

Figure 5: Characteristics of clusters of points.
Clusters of two-dimensional points $\mathbf{p_i} = (p_1, p_2)$ are illustrated. All points from one cluster are within a sphere having the center at $\mathbf{p_{midp}}$ and radius $\frac{\delta}{2}$.

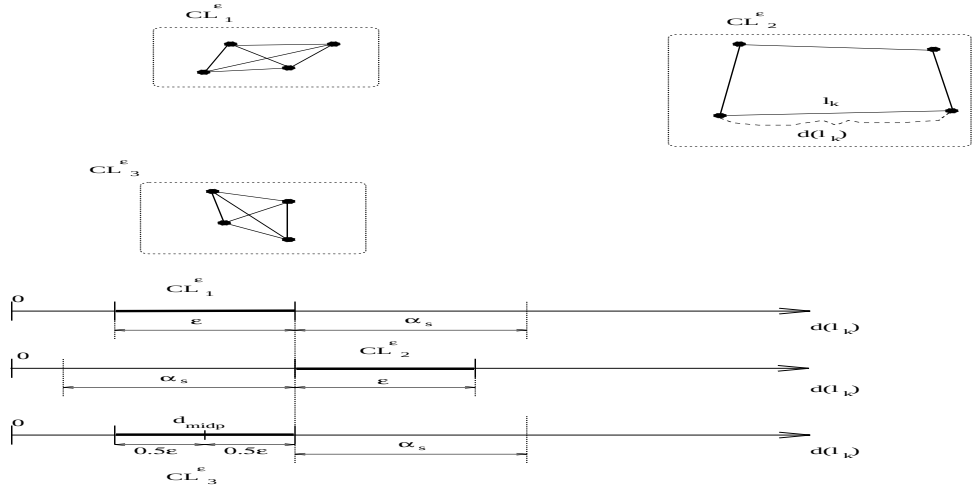

Figure 6: Characteristics of clusters of links.
Top: Three clusters of links partitioning two-dimensional points into three clusters of points. Bottom: Characteristics of the three clusters of links. The horizontal axis represents values of Euclidean distances between pairs of points $d(l_k)$. Links in each cluster of links differ in length by no more than ε.

3.2 The clustering method

Given a set of elements I and the goal, the unknown parameters of the classification problem are the values θ and α for each final cluster, as well as, the number of final clusters N. Two steps are taken to partition the input elements into clusters. First, a value of intra-cluster dissimilarity θ is fixed and clusters characterized by θ are formed by grouping pairs of elements. The result of the first step is a set of clusters denoted as $\{CE_m^\theta\}_{m=1}^{M_\theta}$ since they are only characterized by θ. Second, a value of θ is estimated. A cluster CE_j^θ with the estimated value θ is selected into the final partition $\{CE_j\}_{j=1}^N$. The choice of CE_j^θ is driven by a maximization of inter-cluster dissimilarity α and a minimization of intra-cluster dissimilarity θ. The final partition $\{CE_j\}_{j=1}^N$ is aimed to be identical with a ground truth partition $\{C_j\}_{j=1}^N$, which is assumed to exist for the purpose of evaluating the classification accuracy (number of misclassified elements). The development of the proposed classification method is described next by addressing the following issues.

(1) Given a fixed value of intra-cluster dissimilarity θ, how to estimate an unknown cluster at a single element?

(2) How to group pairs of elements based on estimates calculated at each element?

(3) How to estimate a value of intra-cluster dissimilarity θ of an unknown cluster of elements?

(1) Estimate of an unknown cluster derived from a single element

In order to create an unknown cluster C_j, every pair of elements in C_j should be grouped together. The grouping is based on a certain estimate of the cluster C_j computed at each element. The best estimate of an unknown cluster C_j is obtained at a single element $\mathbf{e_i}$ if the element $\mathbf{e_i}$ gives rise to a cluster $C_{\mathbf{e_i}}$ identical with the unknown one C_j. It would be possible to create the cluster $C_{\mathbf{e_i}} = C_j$ if the unknown cluster C_j of elements is characterized by a value of inter-cluster dissimilarity α larger

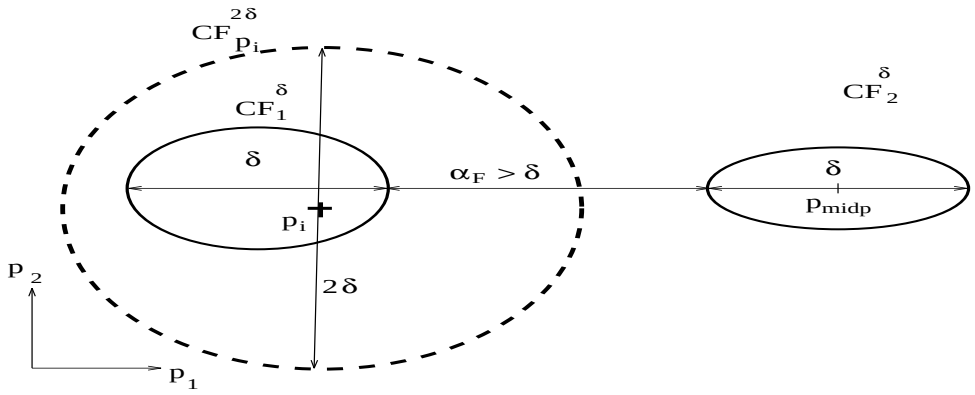

Figure 7: A cluster of points with $\alpha = \alpha_f > \delta = \theta$.

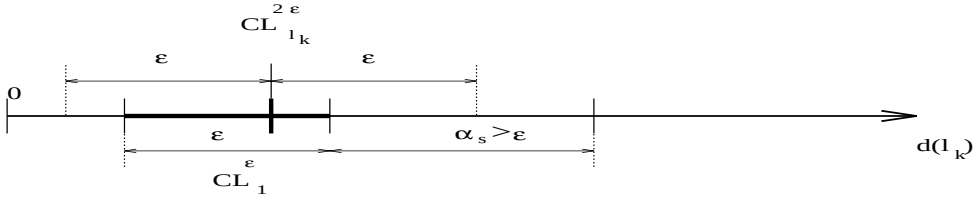

Figure 8: A cluster of links with $\alpha = \alpha_s > \varepsilon = \theta$.

than a value of intra-cluster dissimilarity θ. (see Figures 7 and 8). Under the assumption $\alpha > \theta$, the cluster $C_{\mathbf{e_i}} = \{\mathbf{e_k}\}$ is obtained from any element $\mathbf{e_i} \in C_j$ by grouping together all other elements $\mathbf{e_k}$ satisfying the inequality $\| \mathbf{e_i} - \mathbf{e_k} \| \leq \theta$. Thus for any two elements $\mathbf{e_1}$ and $\mathbf{e_2}$ from the cluster $C_{\mathbf{e_i}}$, their pairwise dissimilarity is always less than 2θ; if $\mathbf{e_1} \in C_{\mathbf{e_i}}$ and $\mathbf{e_2} \in C_{\mathbf{e_i}}$ then $\| \mathbf{e_1} - \mathbf{e_2} \| \leq 2\theta$. The last fact about 2θ intra-cluster dissimilarity leads to a notation $C_{\mathbf{e_i}} = C^{2\theta}_{\mathbf{e_i}}$.

(2) Grouping elements into clusters using similarity analysis

The final clusters $\{CE^{\theta}_m\}$ characterized by θ are obtained in the following way:

(a) Create clusters $C^{2\theta}_{\mathbf{e_i}} = \{\mathbf{e_k}\}$, such that $\| \mathbf{e_i} - \mathbf{e_k} \| \leq \theta$.

(b) Compare all pairs of clusters $C^{2\theta}_{\mathbf{e_i}}$.

(c) Assign elements into the final clusters of elements $\{CE^{\theta}_m\}$ based on the comparisons in (b).

Steps (b) and (c) are performed using similarity analysis. The similarity analysis

relates intra-cluster dissimilarity θ and inter-cluster dissimilarity α of an unknown cluster $C_j^{\theta,\alpha}$. The relationship between α and θ breaks down into two cases; $\alpha > \theta$ and $\alpha \leq \theta$.

For $\alpha > \theta$, an unknown cluster $C_j^{\theta,\alpha}$ has a value of inter-cluster dissimilarity larger than a value of intra-cluster dissimilarity. In this case, clusters $C_{\mathbf{e_i}}^{2\theta}$ are either identical to or totally different from an unknown cluster $C_j^{\theta,\alpha}$. Thus two elements $\mathbf{e_1}$ and $\mathbf{e_2}$ would belong to the same final cluster CE_m^θ if $C_{\mathbf{e_1}}^{2\theta} = C_{\mathbf{e_2}}^{2\theta}$. For $\alpha \leq \theta$, an unknown cluster $C_j^{\theta,\alpha}$ has a value of inter-cluster dissimilarity smaller or equal to a value of intra-cluster dissimilarity. In this case, clusters $C_{\mathbf{e_i} \in C_j^\theta}^{2\theta}$ are not identical to an unknown cluster $C_j^{\theta,\alpha}$. A cluster $C_{\mathbf{e_i}}^{2\theta}$ is a superset of $C_j^{\theta,\alpha}$ because the cluster $C_{\mathbf{e_i}}^{2\theta}$ also contains some exterior elements of $C_j^{\theta,\alpha}$ due to $\alpha \leq \theta$.

For $\alpha \leq \theta$, it is not known how to group elements into clusters and the analysis of this case proceeds. Our analysis assumes that the case $\alpha \leq \theta$ occurs due to a random noise. This assumption about random noise leads to a statistical analysis of similarity of clusters $C_{\mathbf{e_i}}^{2\theta}$. Two issues are investigated next: (i) a statistical parameter of a cluster $C_j^{\theta,\alpha}$ that would be invariant in the presence of noise and (ii) a maximum deviation of two statistically invariant parameters computed from clusters $C_j^{\theta,\alpha}$ and $C_{\mathbf{e_i}}^{2\theta}$. First, let us assume that deterministic values of elements are corrupted by a zero mean random noise with a symmetric central distribution. Then a sample mean (average) of elements would be a statistically invariant parameter because the mean of noise is zero. Although the sample mean of noise corrupted elements varies depending on each realization of noise, it is a fixed number for a given set of noise corrupted elements. Thus the sample mean $\bar{\mu}_j$ of noise corrupted elements in $C_j^{\theta,\alpha}$ is a statistically invariant parameter under the aforementioned assumptions about noise.

Second, a sample mean is computed from each cluster $C_{\mathbf{e_i}}^{2\theta}$ and is denoted as $\bar{\mu}_{\mathbf{e_i}}$. The deviation of $\bar{\mu}_{\mathbf{e_i}}$ from $\bar{\mu}_j$ is under investigation. If $C_{\mathbf{e_i}}^{2\theta}$ is a subset of C_j^θ

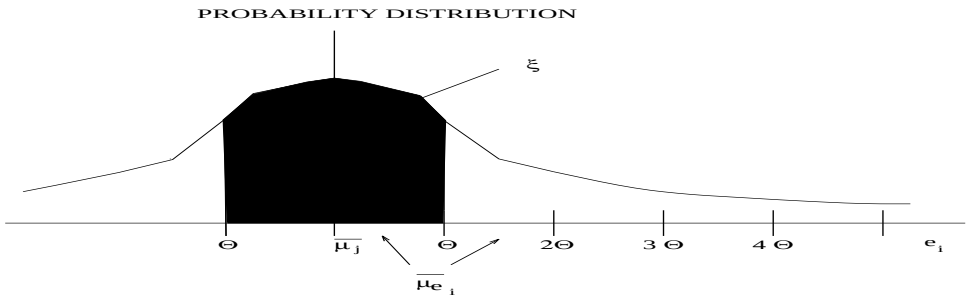

Figure 9: Confidence interval for 1D case of $\mathbf{e_i}$.

($C^{2\theta}_{\mathbf{e_i}} \subseteq C^{\theta}_{j}$) then the sample mean $\bar{\mu}_{\mathbf{e_i}}$ would not deviate by more than θ from $\bar{\mu}_j$; $\| \bar{\mu}_{\mathbf{e_i}} - \bar{\mu}_j \| \leq \theta$. This statement is always true. If there are two arbitrary subsets $C^{2\theta}_{\mathbf{e_i}} \subseteq C^{\theta}_{j}$ and $C^{2\theta}_{\mathbf{e_l}} \subseteq C^{\theta}_{j}$ then their sample means would not be more than θ apart, as well; $\| \bar{\mu}_{\mathbf{e_i}} - \bar{\mu}_{\mathbf{e_l}} \| \leq \theta$. If $C^{2\theta}_{\mathbf{e_i}}$ is a superset of C^{θ}_{j} ($C^{2\theta}_{\mathbf{e_i}} \supset C^{\theta}_{j}$) then the same deviation of $\bar{\mu}_{\mathbf{e_i}}$ from either $\bar{\mu}_j$ or $\bar{\mu}_{\mathbf{e_l}}$ is assumed as before for $\mathbf{e_i}, \mathbf{e_l} \in C^{\theta}_{j}$; $\| \bar{\mu}_{\mathbf{e_i}} - \bar{\mu}_j \| \leq \theta$ and $\| \bar{\mu}_{\mathbf{e_i}} - \bar{\mu}_{\mathbf{e_l}} \| \leq \theta$. The validity of the previous if statement depends on the ratio of elements from the true cluster C^{θ}_{j} and other clusters exterior to C^{θ}_{j}. Thus for the second issue, the sample mean $\bar{\mu}_{\mathbf{e_i}}$ is not expected to deviated from $\bar{\mu}_j$ by more than θ; $\| \bar{\mu}_{\mathbf{e_i}} - \bar{\mu}_j \| \leq \theta$ (see Figure 9) and any two elements $\mathbf{e_1}$ and $\mathbf{e_2}$ would be grouped together if their corresponding sample means $\bar{\mu}_{\mathbf{e_1}}$ and $\bar{\mu}_{\mathbf{e_2}}$ are not more than θ apart; if $\| \bar{\mu}_{\mathbf{e_1}} - \bar{\mu}_{\mathbf{e_2}} \| \leq \theta$ then $\mathbf{e_1}, \mathbf{e_2} \in CE^{\theta}_{j}$.

The inequality $\| \bar{\mu}_{\mathbf{e_1}} - \bar{\mu}_{\mathbf{e_2}} \| \leq \theta$ used for $\alpha \leq \theta$ can be applied to the case $\alpha > \theta$. There would be no classification error in the final partition $\{CE^{\theta}_{m}\}^{M}_{m=1}$ for the case $\alpha > \theta$ if the inequality was used. For $\alpha \leq \theta$, the classification error is evaluated in a statistical framework as a probability $Pr(\| \bar{\mu}_{\mathbf{e_i} \in C^{\theta}_{j}} - \bar{\mu}_j \| > \theta)$. The complement probability $Pr(\| \bar{\mu}_{\mathbf{e_i} \in C^{\theta}_{j}} - \bar{\mu}_j \| \leq \theta) = \xi$ corresponds to a confidence interval of the mean estimator with a confidence coefficient ξ and upper and lower confidence limits $\pm\theta$.

(3) Estimation of intra-cluster dissimilarity θ

The value of intra-cluster dissimilarity θ is a priori unknown for an unknown cluster

$C_j^{\theta,\alpha}$. An estimation of the value θ is based on the assumption that an unknown cluster C_j with a maximum inter-cluster dissimilarity α does not change the elements in C_j for a large interval of values θ. Thus clusters CE_j^θ that do not change their elements for a large interval of θ are selected into the final partition $\{CE_j\}_{j=1}^N$. The set $\{CE_j\}_{j=1}^N$ is an estimate of the ground truth partition $\{C_j\}_{j=1}^N$.

The procedure for an automatic selection of θ uses an analysis of hierarchical classification results and consists of four steps: (i) produce multiple sets of clusters by varying the value θ called multiscale classification, (ii) organize multiscale sets of clusters into a hierarchy of clusters, (iii) detect clusters that do not change their elements for a large interval of θ and (iv) select the value θ based on the analysis in Step (iii). Hierarchical organization of the output is defined as a nested sequence of sets of clusters along the scale axis. The nested sequence is understood as follows: a cluster obtained at scale θ cannot split at scale $\theta+\Delta$ and cannot merge at scale $\theta-\Delta$ with other clusters. The hierarchy of multiscale classification results is guaranteed by modifying elements within the final cluster CE_m^θ created at each scale θ to the sample mean of elements of the cluster. This implementation of hierarchical organization can be supported by the following fact. Two elements $\mathbf{e_1}$ and $\mathbf{e_2}$ which have identical values $\mathbf{e_1} = \mathbf{e_2}$ belong to the same cluster CE_m^θ for all scales $\theta \geq 0$.

3.3 Clustering algorithms

Proposed density-based clustering, where elements are links, requires (1) to map clusters of links into clusters of points and (2) to process a large number of links. These two issues are tackled before the final algorithms for location- and density-based clusterings are provided.

Specifics of density-based clustering

In order to obtain clusters of points, a mapping from clusters of links to clusters of

points is designed. The mapping consists of three steps. (1) Compute an average link length of each cluster of links. (2) Construct a minimum spanning tree from the average values of individual clusters of links. (3) Form clusters of points sequentially from clusters of links in the order given by the minimum spanning tree (from smaller to larger average values).

Knowing the mapping, the number of processed links is decreased by merging links in the order of the link distances $d(l_k)$ (from the shortest links to the longest links). Clusters CL_m^{ε} are created and the corresponding clusters of points CS_j^{ε} are derived immediately. No other links, which contain already merged points $\mathbf{p_i} \in CS_j^{\varepsilon}$, will be processed afterwards. When the union of all clusters of points includes all given points ($\cup CS_j^{\varepsilon} = \cup \mathbf{p_i}$) then no more links are processed.

Clustering algorithm for location-based clustering

(1) Set $\delta = 0$.

(2) Create a cluster $CF_{\mathbf{p_i}}^{2\delta} = \{\mathbf{p_k}\}$ at each point $\mathbf{p_i}$, such that $\parallel \mathbf{p_i} - \mathbf{p_k} \parallel \leq \delta$.

(3) Calculate sample means of $CF_{\mathbf{p_i}}^{2\delta}$; $\bar{\mu}_{\mathbf{p_i}} = \frac{1}{M_{\mathbf{p_i}}} \sum_{k=1}^{M_{\mathbf{p_i}}} \mathbf{p_k} \in CF_{\mathbf{p_i}}^{2\delta}$.

(4) Group together any two points $\mathbf{p_1}$ and $\mathbf{p_2}$ into a cluster CF_m^{δ} if $\parallel \bar{\mu}_{\mathbf{p_1}} - \bar{\mu}_{\mathbf{p_2}} \parallel \leq \delta$.

(5) Assign the sample mean of a cluster CF_m^{δ} to all points in CF_m^{δ} (for all m).

(6) Increase δ and repeat from Step 2 until all points are clustered into one cluster.

(7) Select those clusters CF_m^{δ} into the final partition $\{CF_j\}_{j=1}^{N}$ that do not change over a large interval of δ values.

Clustering algorithm for density-based clustering

(1) Set $\varepsilon = 0$ and calculate point separations $d(l_k)$ (length of links) for all pairs of points $\mathbf{p_i}$.

(2) Order $d(l_k)$ from the shortest to the longest; $d(l_1) \leq d(l_2) \leq \ldots$.

(3) Create clusters of links $CL_{l_k}^{2\varepsilon}$ for each individual link l_k, such that $d(l_k) = d_{midp} \leq d(l_1) + \varepsilon$.

(4) Calculate sample means $\bar{\mu}_{l_k} = \frac{1}{M_{l_k}} \sum_{i=1}^{M_{l_k}} d(l_i \in CL_{l_k}^{2\varepsilon})$.

(5) Group together pairs of links l_1 and l_2 sharing one point $\mathbf{p_i}$ into a cluster of links CL_m^{ε} if $\mid \bar{\mu}_{l_1} - \bar{\mu}_{l_2} \mid \leq \varepsilon$.

(6) Assign those unassigned points to clusters CS_j^{ε} which belong to links creating clusters CL_m^{ε}.

(7) Remove all links from the ordered set, which contain already assigned points.

(8) Perform calculations from Step 3 for increased upper limit $d(l_1) \leftarrow d(l_1) + \varepsilon$ until there exist unassigned points.

(9) Assign the link average of a cluster CL_m^{ε} to all links from the cluster CL_m^{ε} (for all m).

(10) Increase ε and repeat from Step 2 until all points are partitioned into one cluster.

(11) Select those CL_m^{ε} clusters into the final partition $\{CS_j\}_{j=1}^{N}$ that do not change over a large interval of ε values.

3.4 Related clustering methods

Location-based clustering is related to centroid clustering [5] and density-based clustering is related to Zahn's method [1]. Centroid clustering achieves results identical to the proposed location-based clustering although the algorithms are different (see [5]). The only difference in performance is in the case of equidistant points, when the proposed method gives a unique solution, while the centroid clustering method does not, due to sequential merging and updating of point coordinates.

Zahn's method consists of the followings steps: (1) Construct the minimum spanning tree (MST) for a given point pattern. (2) Identify inconsistent links in the MST. (3) Remove inconsistent links to form connected components (clusters). A link is called inconsistent if the link distance is significantly larger than the average of nearby link distances on both sides of the link. The proposed density-based clustering

differs from the Zahn's clustering in the following ways: (1) We use the average of the largest set of link distances (descriptors of $CL_{l_k}^{2\varepsilon}$) rather than nearby link distances for defining inconsistent link and this leads to more accurate estimates of inconsistent links. (2) We replace the threshold for removing inconsistent links ("significantly larger" in the definition of inconsistent links) with a simple statistical rule. (3) We work with all links from a complete graph[1] rather than a few links selected by MST (this is crucial for detecting transparent clusters).

4 Performance Evaluation

The problem of image texture analysis is introduced in Section 4.1. This problem statement explains our motivation for pattern decomposition followed by using both location- and density-based clusterings. Theoretical and experimental evaluations of the methods follow next. The evaluation focuses on (1) clustering accuracy in Section 4.2, (2) detection of Gestalt clusters in Section 4.3, and (3) performance on real applications in Section 4.4.

4.1 Image texture analysis

An image texture is modeled as a set of uniformly distributed identical primitives shown in Figure 4. Each primitive in Figure 4 is characterized by its color and size. All primitives having similar colors and shapes are uniformly distributed therefore the centroid coordinates of all texture interior primitives have similar inter-neighbor distances. The goal of image texture analysis is to (1) obtain primitives, (2) partition the primitives into sets of primitives called texture and (3) describe each texture using interior primitives and their distribution. In this work, all primitives are found

[1] Links between all pairs of points create a complete graph according to the notation in graph theory.

based exclusively on their color. A homogeneity based segmentation [20] is applied to an image. The segmentation partitions an image into homogeneous regions called primitives (similar colors are within a region). A point pattern is obtained from all primitives (regions) by measuring an average color and centroid coordinates of each primitive.

Given the pattern, a decomposition of features is performed first. One set of features corresponds to the centroid measurements and the other to the color measurements of primitives. Two lower dimensional patterns are created from these features. The location-based clustering is applied to the pattern consisting of the color feature and the density-based clustering is applied to the pattern consisting of the centroid coordinate features. Clustering results are combined and shown in Figure 4 (bottom). The cluster similarity in each subspace provides a texture description characterized by similarity of primitives and uniformity of distribution.

The density-based clustering was applied to the pattern shown in Figure 10 (top). The points from the pattern are numbered from zero to the maximum number of points. The output in Figure 10 (bottom) shows three clusters labeled by the number of a point from each cluster that has the minimum value of its number. Partial spatial occlusions of blobs in the original image gave rise to a corrupted set of features corresponding to the centroid coordinates of primitives. From this follows that the lower dimensional points are not absolutely uniformly distributed in the corresponding subspace. The value of similarity ε was selected manually. The method demonstrates its exceptional property of separating spatially interleaved clusters which is a unique property of the clustering methods described here.

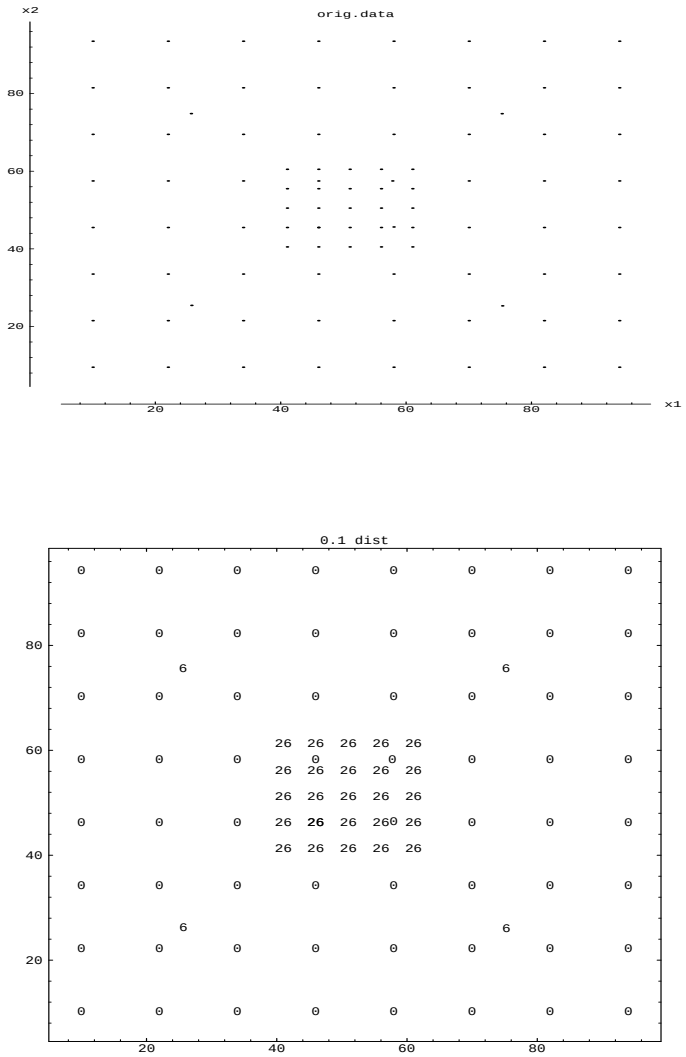

Figure 10: Spatially interleaved clusters.

Top - original point pattern. Bottom - density-based clustering at $\varepsilon = 0.1$.

4.2 Accuracy and computational requirements

Experimental analysis of clustering accuracy is evaluated by measuring the number of misclassified points with respect to the ground truth. Clustering accuracy is tested for (1) synthetic point patterns generated using location and density models of clusters and (2) standard test point patterns (80x, IRIS), which have been used by several other researches to illustrate properties of clusters (80x is used in [5] and IRIS in [12, 5, 1]). Computational requirements are stated. Experimental results are compared with four other clustering methods, two hierarchical methods - single link and complete link, and two partitional - FORGY and CLUSTER [5].

Synthetic and standard point patterns

A point pattern is generated and the points are numbered. Detected clusters are shown pictorially as sets of points labeled by same number. The common number for a cluster corresponds to the number of a point that has the minimum value of its number. Two models were used to generate synthetic point pattern. First, three locations $\mathbf{gc_1} = (10, 10)$, $\mathbf{gc_2} = (20, 20)$, and $\mathbf{gc_3} = (30, 30)$ in a two-dimensional space gave rise to a synthetic pattern with three clusters. These three locations were perturbed by Gaussian noise (zero mean, variation σ) with various values of the standard deviation σ. The number of points derived by perturbations of each location varied as well. Figure 11 shows two realizations of synthetic patterns (left column). Results obtained from the location-based clustering are shown in Figure 11, right column.

Second, a 2D synthetic point pattern (64 points) was generated with four clusters (30, 10, 12, 12 points) of different densities. The point pattern is shown in Figure 12 (left). Points from the pattern were corrupted by uniform noise $\Delta \pm 0.5$ and by Gaussian noise $\sigma = 0.25$ (see Figure 12 middle and right). Results obtained from density-based clustering method for the point patterns are shown in Figure 13.

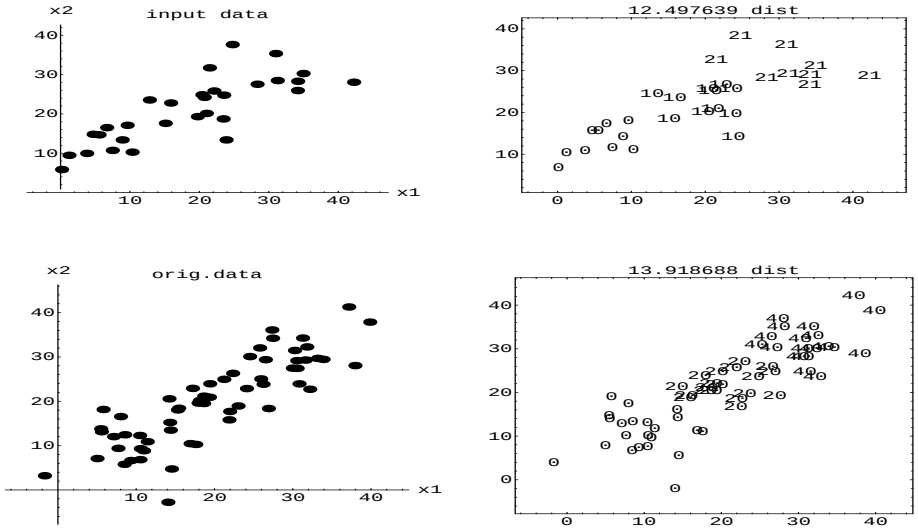

Figure 11: Clusters detected by location-based clustering.
Left column - three clusters, 30 points (top) and 60 points (bottom), Gaussian noise (zero mean,$\sigma = 5$). Right column - Results of location-based clustering corresponding to the left column point patterns ($\delta = 12.497639$ top, $\delta = 13.918688$ bottom).

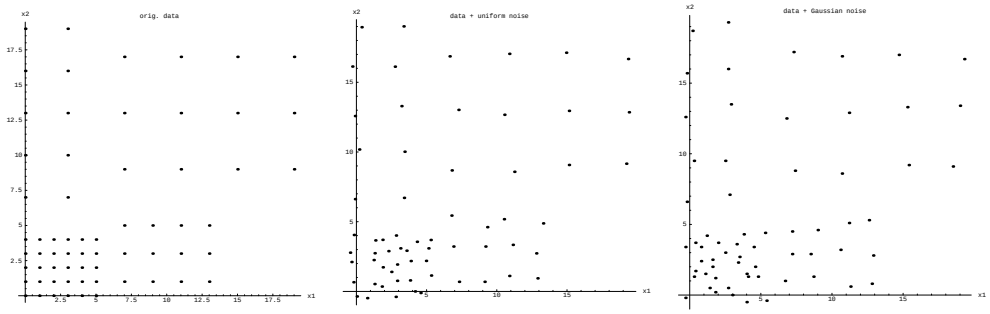

Figure 12: Synthetic pattern with four clusters of different densities.
Internal link distances between points from the four clusters are equal to 1, 2, 3 and 4 (first left). Locations of points are corrupted by noise with uniform (middle) and Gaussian (right) distributions.

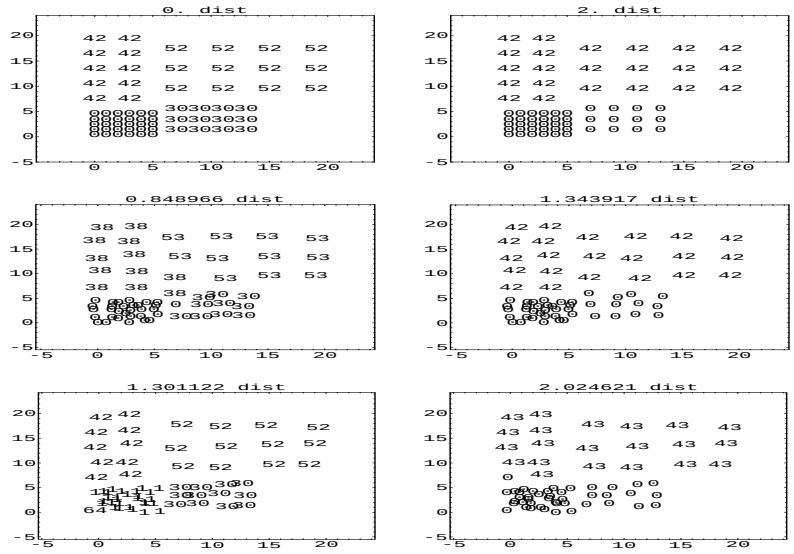

Figure 13: Clusters detected by density-based clustering.
Clustering results for the patterns shown in Figure 12. Cluster labels for points without noise (top row), with uniform noise (middle row) and with Gaussian noise (bottom row). The number above each plot refers to the value of ε.

We selected two standard point patterns obtained from (1) handwritten character recognition problem (recognition of 80X with 8 features) and (2) flower recognition problem (Fisher-Anderson iris data denoted as IRIS; recognition of iris setosa, iris versicolor and iris virginica with 4 features). The features are shown in Figure 14. The data set 80x contains 45 points with 3 categories each of size 15 points. The data set IRIS contains 150 points with 3 categories each of size 50. Results expressed in terms of misclassified points are in Table 3. Decomposition of features followed by location- and density-based clusterings is explored for each point pattern (80X and IRIS). It is unknown how to determine the choice of features for the decomposition. The goal is to create lower dimensional point patterns showing inherent tendency to form sets of points with similar locations or approximately constant density. An exhaustive search for the best division in terms of classification error was used.

Comparisons with other clustering methods

Two partitional clustering methods (FORGY, CLUSTER) and two hierarchical clus-

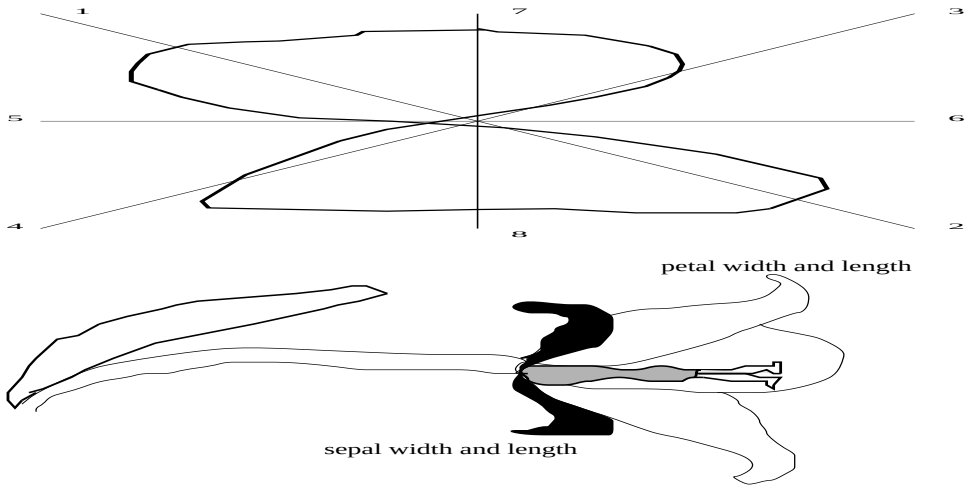

Figure 14: Features for standard point patterns.
Features are shown for the 80x (top) and the IRIS (bottom) standard data.

tering methods (single and complete link) were compared with the proposed methods. Compared four methods are fully described in [5]. The comparison is based on the number of misclassified points with respect to the ground truth. The two hierarchical methods were selected because they cluster points using links (clustering by graph theory) that is similar to the proposed density-based clustering. The other two methods, FORGY and CLUSTER, cluster points using their coordinates that is similar to the proposed location-based clustering.

The misclassified points for hierarchical methods were counted from the best possible non-overlapping point pattern partition (the closest to the ground truth) with dominant labels within correct clusters. The misclassified points for partitional methods were counted from the closest partition to the ground truth for the two input values (variables), (1) a random seed location for the initial clustering and (2) a number of expected clusters in the result. A summary of clustering results in terms of misclassified points is provided in Tables 1, 2 and 3 for synthetic and standard data shown in Figures 11, 12 and 14. The order of methods based on their performance is shown in the most right column of each table. The performance criterion is the sum

of misclassified points for several point patterns with known ground truth clusters (shown in the second column from right in each table).

The best method for a class of point patterns shown in Figure 11 is the proposed location-based clustering (see Table 1). A class of point patterns shown in Figure 12 was clustered the most accurately by the proposed density-based clustering (see Table 2). A combination of location- and density-based clusterings applied to 80X and IRIS data led to the best clustering results (see Table 3). The eight-dimensional point pattern 80X was decomposed experimentally into two lower-dimensional spaces; one $n_s = 4$-dimensional subspace (features 1,2,7,8) and one $n_f = 4$-dimensional subspace (features 3,4,5,6) in order to achieve the result stated in Table 3. By applying the location-based clustering to $n_f = 4$-dimensional points followed by the density-based clustering applied to $n_s = 4$-dimensional points we could separate 0 from 8X and then 8 from X. The four-dimensional point pattern IRIS was decomposed experimentally as well, but the clustering results were not better than the results from location-based clustering applied alone. All six methods used for the comparison were applied to a class of point patterns with spatially interleaved clusters, e.g., Figures 3 and 10. Proposed density-based clustering outperforms all other methods because it is the only method that is able to separate spatially interleaved clusters.

Time and memory requirements

A time requirement for running each method is linearly proportional to the number of processed elements (N_{point} points, N_{link} links) and to the number of used elements for a sample mean calculation at each element ($N_{CF^{2\delta}_{\mathrm{Pi}}}$ and $N_{CL^{2\varepsilon}_{l_k}}$). The number of processed links N_{link} was reduced by sequential mapping of clusters of links to clusters of points therefore the time requirement was lowered. Time measurements were taken for various patterns. For example, the user time needed for clustering a point pattern having 60 points (similar to one in Figure 11 bottom) takes in average $0.06s$

Table 1: Number of misclassified points resulting from clustering data in Figure 11.

method / data	$\sigma = 3$ *30 pts*	$\sigma = 3$ *60 pts*	$\sigma = 5$ *30 pts*	$\sigma = 5$ *60 pts*	$\sum$	perform. order
locat. clus.	0	1	1	3	5	1.
dens. clus.	1	11	8	11	31	6.
single link	1	2	7	9	19	5.
complete link	1	2	5	6	14	4.
CLUSTER	2	1	2	3	8	3.
FORGY	1	0	2	3	6	2.

Table 2: Number of misclassified points resulting from clustering data in Figure 12.

method / data	*no noise*	$\Delta = \pm 0.5$	$\sigma = 0.25$	$\sum$	order
locat. clus.	0	13	10	23	5.
dens. clus.	0	3	1	4	1.
single link	9	14	13	36	6.
complete link	0	11	4	15	2.
CLUSTER	6	6	6	18	4.
FORGY	5	6	6	17	3.

Table 3: Number of misclassified points resulting from clustering 80X and IRIS data.

method / data	*80x 45 pts*	*IRIS 150 pts*	$\sum$	perform. order
locat. + dens. clus.	7	14	21	1.
locat. clus.	24	14	38	4.
dens. clus.	18	24	42	5.
single link	24	25	49	7.
complete link	12	34	46	6.
CLUSTER	15	16	31	3.
FORGY	7	16	23	2.

per one location-based clustering and 1.33s per one density-based clustering on Sparc 20 machine. The size of memory required is linearly proportional to the number of processed elements (N_{point} points, N_{link} links).

4.3 Detection of Gestalt clusters

Gestalt clusters are two-dimensional clusters of points that are perceived by humans as point groupings [1, 10]. The goal of this Section is to test the properties of the proposed methods for detecting and describing Gestalt clusters. Properties of the location- and density-based methods are demonstrated using the data of sample cluster problems from [1] and [2]. The sample problems [1] are (1) composite cluster problem, (2) particle track description, (3) touching clusters, (4) touching Gaussian clusters and (5) density gradient detection.

Each of the problems tackles one or more configurations of clusters in a given point pattern. The configurations of clusters refer to the properties of individual clusters in a point pattern by which the clusters are detected. The properties are, for example, location and density of clusters, distribution of points within a cluster (Gaussian clusters), spatial shape of clusters (lines in particle tracks problems), mutual spatial position of clusters (touching clusters, surrounding clusters), density gradient of clusters. Point patterns containing clusters with the abovementioned properties are in Figures 15, 16, 17 and 18. Results corresponding to Gestalt clusters are shown as well. All results were obtained using the proposed methods. There are two acceptable results in Figure 16 for the case of touching clusters with identical densities. The choice of the method and the similarity parameter of a shown result are made manually.

The proposed approach to clustering of elements (points or links) gave rise to location- and density-based clusterings. These clusterings can detect and describe

Gestalt clusters equally well as graph-theoretical methods using minimum spanning tree [1]. Cases of point patterns similar to Figure 18 require special treatment using the graph-theoretical methods (detecting and removing denser cluster followed by clustering the rest of the points). This drawback is not present in the proposed methods.

4.4 Experimental results on real data.

Experimental results on real data are reported for image texture analysis and synthesis. The analysis is conducted by (1) segmenting image into regions, (2) creating a point pattern from regions, (3) clustering point pattern and (4) presenting application dependent results. In the following application, the goal is to represent image texture in a very concise way suitable for storage or coding of texture. In order to achieve this goal, a density of texture primitives (homogeneous regions) is assumed to be constant over the whole texture. Thus a concise representation of textures will consist of description of primitives and spatial distributions.

Figure 19 (left) shows an image with a regular texture (tablecloth) having approximately constant density of dark, bright and gray rectangles. A decomposition of the tablecloth into sets of dark, bright and gray rectangles was performed by (1) creating a point pattern with three features (centroid locations and average intensity value of segmented regions), (2) using the density-based clustering (see the result of clustering in Figure 20) and (3) separating those regions into one image that gave rise to the points grouped into one cluster during the clustering. The decomposition is shown in Figure 21 (top). A possible synthesis of the image is shown in Figure 21 (bottom). The synthesis starts with painting the background (intensity of the largest region) followed by laying a region representative from each cluster at all centroid locations from the cluster. In this way, a textured image is represented more efficiently than

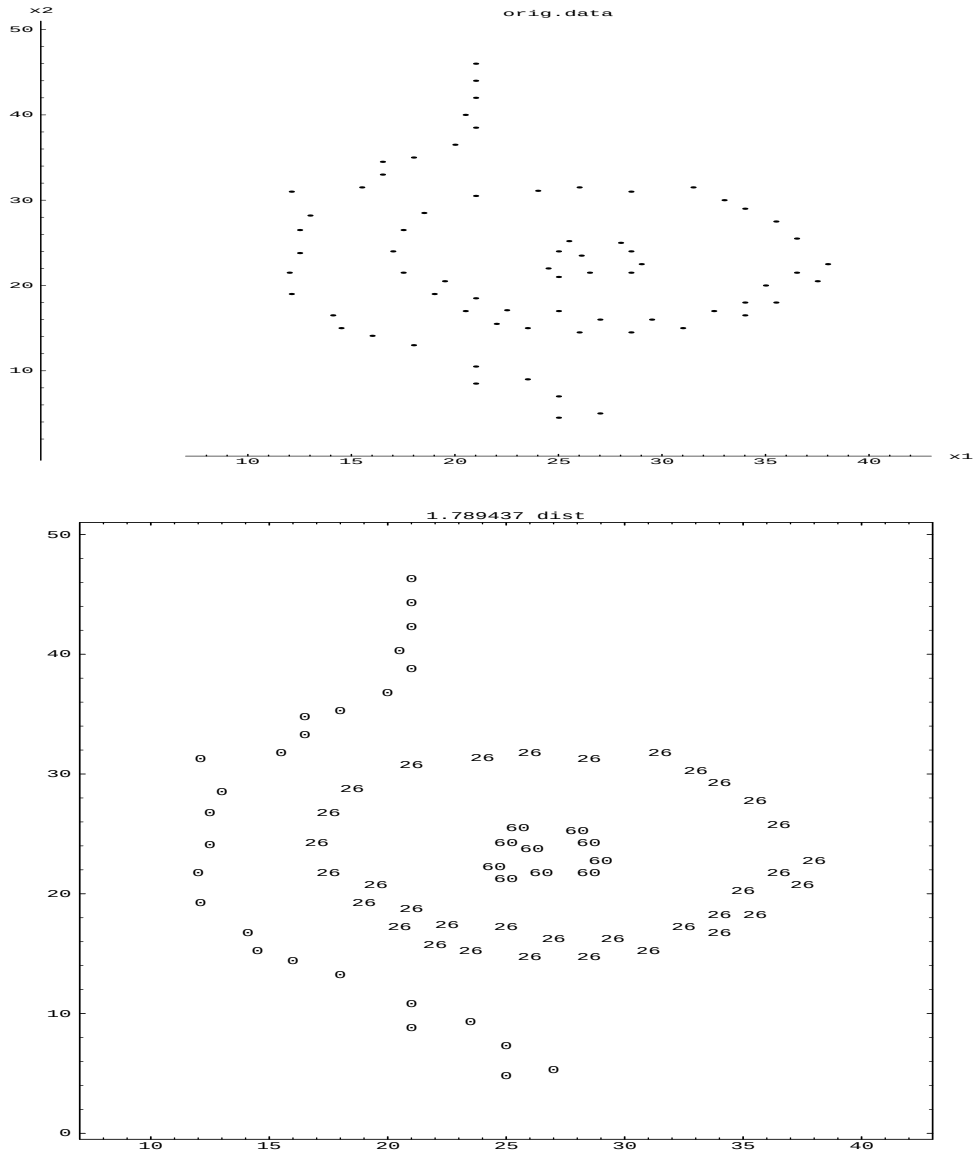

Figure 15: Point pattern showing a composite cluster problem and a problem of the particle track description.
Top - original data, bottom - result of density-based clustering at $\varepsilon = 1.789437$.

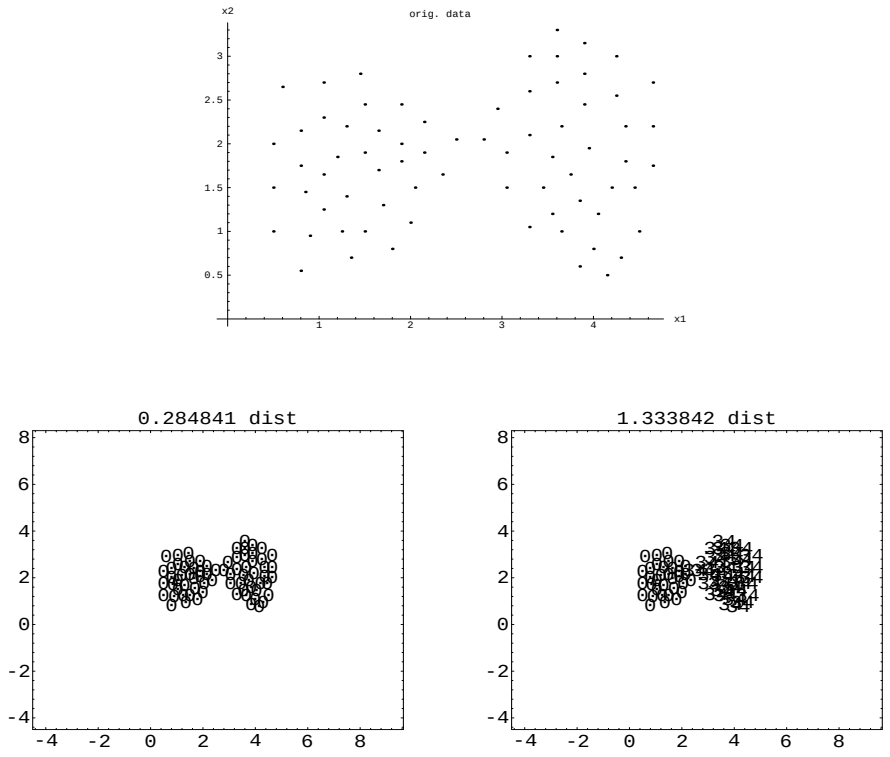

Figure 16: A problem of touching clusters.
Top - original data, bottom - results of density- (left) and location- (right) based clusterings at $\varepsilon = 0.284841$ and $\delta = 1.333842$.

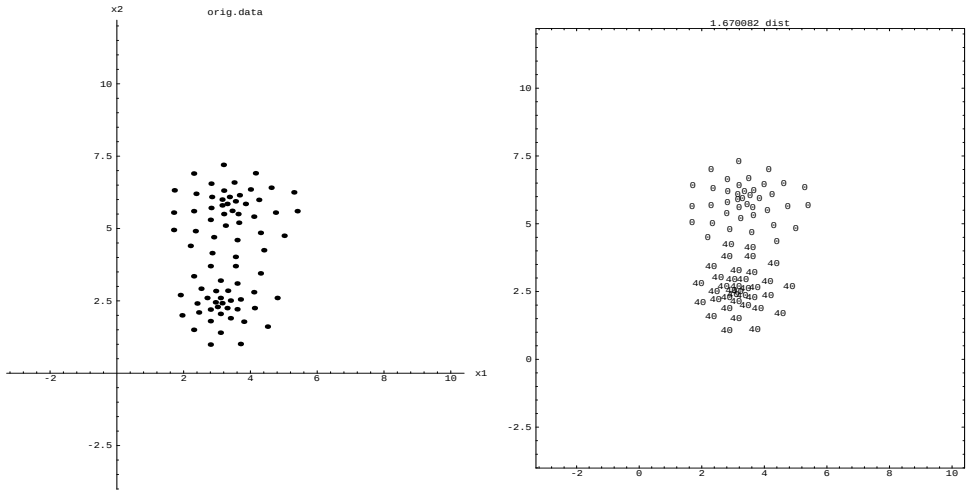

Figure 17: A problem of touching Gaussian clusters.
Left - original data, right - result of location-based clustering at $\delta = 1.670082$.

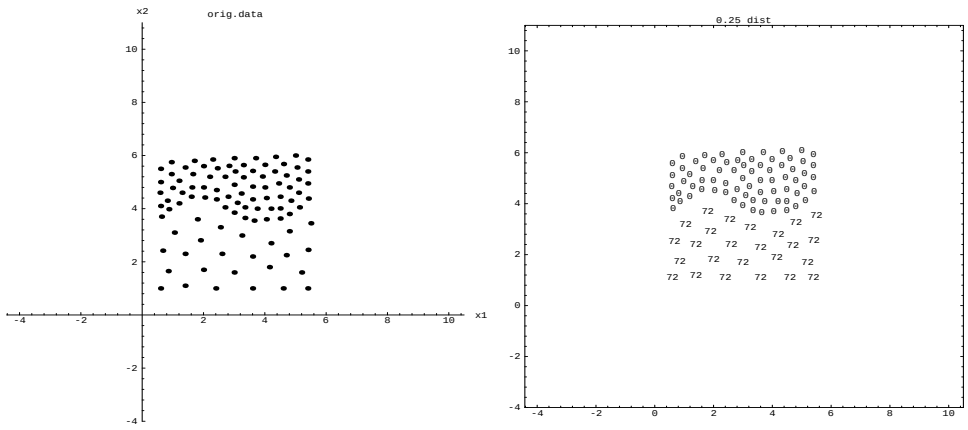

Figure 18: A problem of density gradient detection.
Left - original data, right - result of density-based clustering at $\varepsilon = 0.25$.

any single pixel or region based description.

5 Conclusion

We have presented a new clustering approach using similarity analysis. Two hierarchical clustering algorithms, location- and density-based clusterings, were developed based on this approach. The two methods process locations or point separations denoted as elements $\mathbf{e_i}$. The methods start with grouping elements into clusters $C_{\mathbf{e_i}}$ for every element $\mathbf{e_i}$. All elements in $C_{\mathbf{e_i}}$ are dissimilar to $\mathbf{e_i}$ by no more than a fixed value θ. The dissimilarity of two elements is defined as their Euclidean distance. A sample mean $\bar{\mu}_{\mathbf{e_i}}$ of all elements in $C_{\mathbf{e_i}}$ is calculated. Clusters are formed by grouping elements having similar $\bar{\mu}_{\mathbf{e_i}}$. The resulting set of clusters is identified among all clusters obtained by varying θ. Those clusters that do not change over a large interval of θ are selected into the final partition in order to minimize intra-cluster dissimilarity and maximize inter-cluster dissimilarity.

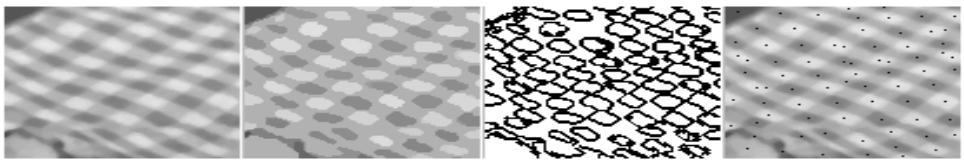

Figure 19: Image texture analysis: Image "Tablecloth".
Left to right: original image, segmented image, contours of segmented image, centroids of segmented regions overlapped with the original image.

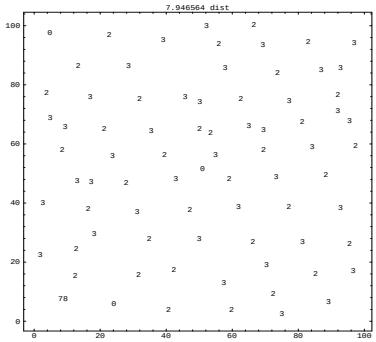

Figure 20: Result of density-based clustering.
A point pattern obtained from Figure 19 is clustered. Numerical labels denote the clusters corresponding to dark blobs of the tablecloth (label 2), white blobs of the tablecloth (label 3), a piece of banana shown in the left corner (label 78) and background with left top triangle and shading of the banana (label 0).

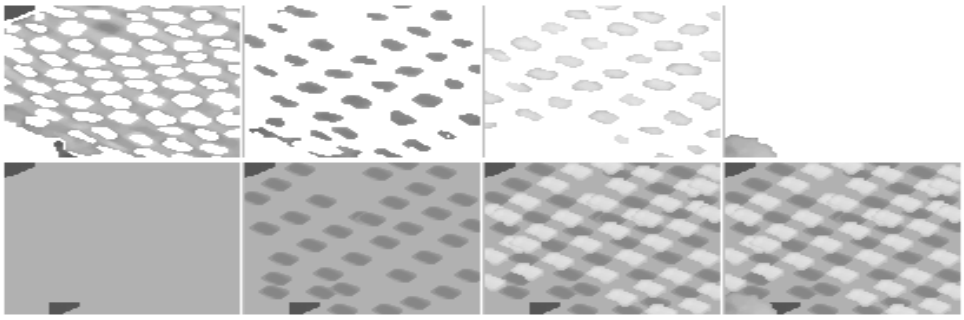

Figure 21: Texture analysis and synthesis.
The image shown in Figure 19 is decomposed and reconstructed. Top row - image decomposition (analysis) based on obtained clusters shown in Figure 20, bottom row - image reconstruction (synthesis).

Location-based clustering achieves results identical to centroid clustering. Density-based clustering can create clusters with points being spatially interleaved and having dissimilar densities. The separation of spatially interleaved clusters is a unique feature of the density-based clustering among all existing methods. The accuracy and computational requirements of the proposed methods were evaluated thoroughly. Synthetic point patterns and standard point patterns (8Ox - handwritten character recognition, IRIS - flower recognition) were used for quantitative experimental evaluation of accuracy. Performance of the clustering methods was compared with four other methods. Correct detections of various Gestalt clusters were shown.

Location- and density-based clusterings were used for image texture analysis. A texture was defined as a set of uniformly distributed identical primitives. Primitives were found by segmenting an image into color homogeneous regions. A point pattern was obtained from textured images by measuring centroid locations and average colors of primitives. Features of this point pattern were divided into two sets, because each set of features required a different clustering model. The centroid locations of primitives, were hypothesized to have uniform distribution therefore the density-based clustering was applied to form clusters in this lower-dimensional subspace corresponding to features of the centroid locations. Properties of primitives, such as color, were modeled to be identical within a texture, therefore the location-based clustering was applied to form clusters in the second lower-dimensional subspace corresponding to the color feature. Resulting texture was identified by combining clustering results in the two subspaces. In a nutshell, this clustering problem required (1) a point pattern decomposition into two lower-dimensional point patterns, (2) location- and density-based clusterings to form clusters from the two point patterns and (3) texture identification using both clustering results. The decomposition approach motivated by image texture analysis was explored for point patterns that originated from hand-

written character recognition and taxonomy problems.

The contributions of this work can be summarized as (1) addressing a decomposition of the clustering problem into two lower-dimensional problems, (2) proposing a new clustering approach for detecting clusters having a constant property of interior points, such as location or density, and (3) developing a density-based clustering method that separates spatially interleaved clusters having various densities.

Acknowledgments

The authors greatfully acknowledge all people who provided the data for experiments. Point patterns from [1] and [2, 21] - Mihran Tuceryan, Texas Instruments; Standard point patterns 80X and IRIS - Chitra Dorai with permission of Anil Jain; The authors thank for providing the four clustering methods to Chitra Dorai and Professor Anil Jain from the Pattern Recognition and Image Processing Laboratory, Michigan State University. This research was supported in part by Advanced Research Projects Agency under grant N00014-93-1-1167 and National Science Foundation under grant IRI 93-19038.

References

[1] C. T. Zahn, "Graph-theoretical methods for detecting and describing gestalt clusters," *IEEE Transaction on Computers*, vol. C-20, pp. 68–86, January 1971.

[2] M. Tuceryan, *Extraction of Perceptual Structure in Dot Patterns.* PhD thesis, Dept. of Computer Science, University of Illinois at Urbana-Champaign, 1986.

[3] N. Ahuja, "Dot pattern processing using voronoi neighborhoods," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 4, pp. 336–343, May 1982.

[4] D. Blostein and N. Ahuja, "Shape from texture: Integrating texture-element extraction and surface estimation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 1233–1251, December 1989.

[5] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data.* Prentice Hall Inc., Englewood Cliffs, New Jersey, 1988.

[6] B. S. Everitt, *Cluster analysis.* Edward Arnold, a division of Hodder and Stoughton, London, 1993.

[7] M. Nadler and E. Smith, *Pattern Recognition Engineering.* John Wiley and sons inc., Canada, 1993.

[8] P. Bajcsy and N. Ahuja, "Uniformity and homogeneity based hierarchical clustering," in *Proceedings of the 13th International Conference on Pattern Recognition*, vol. B, (Vienna, Austria), pp. 96–100, 1996.

[9] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis.* A Wiley Interscience publication, NY, 1973.

[10] N. Ahuja and B. J. Schachter, *Pattern Models.* John Wiley and sons inc., USA, 1983.

[11] E. by R. Bernstein, *Digital Image processing for remote sensing.* IEEE Press, The institute of Electrical and Electronics Engineers, Inc. New York, 1978.

[12] P. H. Sneath and R. R. Sokal, *Numerical Taxonomy.* W. H. Freeman and company, San Francisco, 1973.

[13] E. by P. H. Swain and S. M. Davis, *Remote sensing: The quantitative approach.* McGraw-Hill, Inc., 1978.

[14] H. Hanaizumi, S. Chino, and S. Fujimura, "A binary division algorithm for clustering remotely sensed multispectral images," *IEEE Transaction on Instrumentation and Measurement*, vol. 44, pp. 759–763, June 1995.

[15] Y. Wong and E. C. Posner, "A new clustering algorithm applicable to multi-scale and polarimetric sar images," *IEEE Transaction on Geoscience and Remote Sensing*, vol. 31, pp. 634–644, May 1993.

[16] M. Tuceryan and A. Jain, "Texture segmentation using voronoi polygons," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 12, no. 2, pp. 211–216, 1990.

[17] M. G. Kendall, *The Advanced Theory of Statistics, Volume II.* Hafner publishing company, New York, 1951.

[18] A. S. Fotheringham and F. B. Zhan, "A comparison of three exploratory methods for cluster detection in spatial point patterns," *Geographical analysis*, vol. 28, pp. 200–218, July 1996.

[19] A. Getis and B. Boots, *Models of Spatial Processes. An approach to the study of point, line and area patterns.* Cambridge University Press, 1978.

[20] P. Bajcsy and N. Ahuja, "Segmentation of multidimensional images," in *Proceedings on Image Understanding Workshop*, (Palm Springs, California), pp. 937–942, 1996.

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