# GRID-BASED CLUSTERING IN THE CONTENT-BASED ORGANIZATION OF LARGE IMAGE DATABASES

*Iivari Kunttu*[1], *Leena Lepistö*[1], *Juhani Rauhamaa*[2], *and Ari Visa*[1]

[1]Tampere University of Technology
Institute of Signal Processing
P. O. Box 553, FIN-33101 Tampere, Finland

[2]ABB Oy
Paper, Printing, Metals & Minerals
P. O. Box 94, FIN-00381 Helsinki, Finland

Iivari.Kunttu@tut.fi

## ABSTRACT

In the image databases, there is often a need to organize the images automatically by their content. For the content-based organization of the images, clustering operations can be applied. We present a new, efficient method for the clustering of large image databases. The method is based on hierarchical clustering of the image database using grid. In this paper, the grid-based clustering methods are applied to fast image database browsing and retrieval. Proposed clustering methods are tested using real image databases.

## 1. INTRODUCTION

During recent years, there has been a strong growth in the number and the size of image databases. Nowadays, large image databases typically consist of different types of photographic archives. Also the number of industrial image databases has grown remarkably due to a rapid increase of industrial imaging systems. Therefore, a growing research interest has been focused on the area of content-based search and retrieval of the image databases.

Image database browsing and retrieval are two common applications of the image database search. Intensive research work has been done in the image retrieval, whereas browsing has received less attention. A problem with retrieval systems is the fact that they require an example image from the user to be used as a query image. Therefore, the user has to browse the database beforehand in order to find good example images for the query. This is a difficult and time-consuming task, especially in the case of large image databases (>10 000 images). Hence, an effective method for image database browsing is essential to show the user the database content. From the user's viewpoint, an effective browsing view is such that the representative images are presented in a hierarchical way. This means that the images can be browsed in different scales by moving from general view to more specific image groups in the database. Chen et al. [2], for example, have used agglomerative *k*-means clustering in image browsing. Another key problem with retrieval in large image databases is response time. Namely, for each query the whole database has to be searched to find the images that are most similar to the query image. When the database contains tens or hundreds of thousands of images, the response times increase remarkably. However, in a correctly organized database, the queries can be made only in the selected parts of the database, which makes the query responses faster.

Grid-based clustering methods make it possible to form arbitrarily shaped, distance independent clusters. In these methods, the feature space is quantized into cells using a grid structure. The cells can be merged together to form clusters. Grid-based clustering was originally based on the idea of Warnekar and Krishna [10] to organize the feature space containing patterns. Schikuta [9] has used topological neighbor search algorithm to combine the grid cells to form clusters. Agrawal et al. [1] have presented a density-based clustering method using grid.

In this paper, we present a grid-based clustering method that can be used in the content-based organization of image databases. The advantage of grid-based methods is fast processing time, which is dependent only on the number of the grid cells, not the number of database images. In the grid structure, arbitrarily shaped clusters can be formed by merging the cells together based on their density. In the image database browsing, the content of each cluster can be presented for the user using representative image(s) of the cluster. The hierarchical grid structure makes it possible to move up and down in the grid levels. This way the user can either zoom in, to select a specific image type of interest, or zoom out, to browse the variations in the image database contents. In image retrieval, increased efficiency can be achieved: In the clustered image database, only clusters of user's interest can be selected to the retrieval operations. Therefore, the whole database is not needed to be searched at each query.

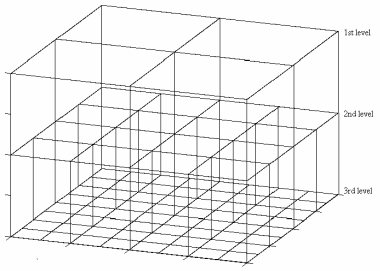


Figure 1. An example of hierarchical grid structure in two-dimensional feature space.

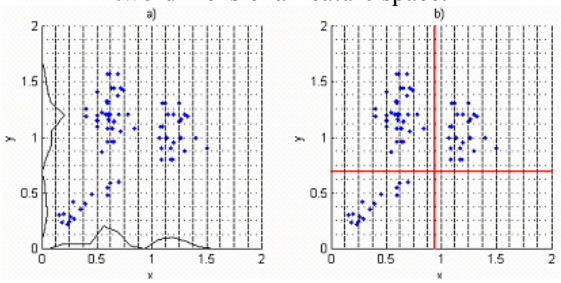


Figure 2. Density-based clustering of the datapoints in a two dimensional grid structure.

## 2. CLUSTERING METHOD

Grid-based clustering methods have been used in some data mining tasks of very large databases [3]. In the grid-based clustering, the feature space is divided into a finite number of rectangular cells, which form a grid. In this grid structure, all the clustering operations are performed. The grid can be formed in multiple resolutions by changing the size of the rectangular cells. Figure 1 presents a simple example of a hierarchical grid structure in three levels that is applied to a two dimensional feature space. In the case of *d*-dimensional space, hyper rectangles (rectangular shaped cube [9]) of *d*-dimensions correspond to the cells. In the hierarchical grid structure, the cell size in the grid can be decreased in order to achieve a more precise cell structure. As in figure 1, the hierarchical structure can be divided into several levels of resolution. Each cell at the high level $k$ is partitioned to form a number of cells at the next lower level $k$+1. The cells at the level $k$+1 are formed by splitting the cell at level $k$ into smaller subcells. In the case of figure 1, each cell produces four subcells at next lower level.

When the images are searched at the low level, the number of the cells is typically high, and similar images occur in several cells. Therefore, the cells containing similar images are needed to be merged together to form clusters. The cells at a selected level can be merged into clusters of similar images using density-based clustering procedure. In the density-based clustering, a cluster is defined to be a region that has a higher density of points than its surrounding region [1], [3]. Hence, the clusters are separated from each other by regions of low density.

In the image databases, the feature vectors describing the image content are often high dimensional. In our approach, the high dimensional feature space is reduced into several low dimensional subspaces. This makes it possible to form density-based clusters in the grid structure. The idea of subspace clustering has been used in the work of Agrawal et al. [1]. In the subspace clustering, the subspaces are considered first separately. The final cluster structure is then formed by combining the clusters obtained in each subspace.

In the subspace clustering, $U$ subspaces are formed based on the original, *d*-dimensional feature space. These subspaces are partitioned using rectangular cells, in which density estimation is performed. Each dimension is partitioned into the same number of cells of equal size, which means that they have same volume. Therefore, the density in the cell can be approximated simply by finding the number of points that lie inside each cell [1]. Subclusters are formed in each subspace separately by finding the local minimums of the density function approximated for the cells. An example of this is presented figure 2a, in which the density functions of each dimension are plotted on the axes of two-dimensional feature space. The local minimums define the borders between the groups of cells that form subclusters. Let $S_u$ denote a set of cells that is defined using a grid structure in subspace $u$. In subclustering, the grid is divided into $n$ parts based on the local minimums of the density function. Hence $S_u$ consists of several sets of cells $\{S_{u1}, S_{u2}, \ldots, S_{un}\}$, which are called subclusters. When the subclustering has been performed in each $U$ subspaces, the resulting set of subclusters is $S = \{S_1, S_2, \ldots, S_U\}$, in which each set represents the subclusters of each subspace. Let $S$ denote the total number of the subclusters in all sets. The final clusters are formed by combining these subclusters. Final clusters $C$ are defined as an intersection of the subclusters of each subspace:

$$C = S_i \bigcap_{i=1,j=1}^{S} S_j \qquad i \neq j \tag{2.1}$$

Hence, a cluster is defined as a connected set of cells, in which the density is high in all dimensions. The division of the feature space of figure 2a is presented in figure 2b. The benefit of this approach is that any distance measurement between the datapoints is not needed. Also the computational cost of this approach is low, which makes it suitable to fast retrieval of image database.

## 3. APPLICATIONS AND EXPERIMENTS

In this section, we present two applications for the grid-based image database clustering. These applications are fast retrieval and browsing of large image databases. For testing, we use defect image databases that are collected

from paper and metal manufacturing processes. The reason for the collection of the defect image databases in process industry is the practical need of controlling the quality and production [8]. However, these image databases can be utilized only if the process operators are capable of browsing and effectively searching the image databases. This is important if the images of a certain defect type are needed to be found from the database. Preliminary work in the defect image clustering was presented in [6], in which the clustering procedure was based on *k*-nearest neighbor classifier.

In this paper, we have used three defect image databases. Labeled paper and metal image databases contained 1204 and 1943 gray level images, respectively. Images of both databases were labeled manually in advance based on the defect type that they represent. In addition to them, we used also an unlabeled paper defect image database of about 10 500 images. The images in all databases images were indexed using color and shape descriptors. The gray level content of the defect images was described using color structure (CS) and color layout (CLD) descriptors of MPEG-7 standard [7]. Mean gray level and gray level variance of the images were also used as additional features. The dimensionality of the gray level descriptors was decreased by re-quantizing image gray levels. Experimental results [5] show that the defect image content can be generalized by decreasing the number of image gray levels, which yields to better classification results. The defect shape was described using five simple shape descriptors, convexity, compactness, principal axes ratio, circular variance and elliptic variance. These descriptors have proved to be effective in the classification of defect images [6]. Resulting feature space of gray level and shape descriptors was 15-dimensional.

### 3.1. Browsing and retrieval

The goal of image database browsing is to show the user a view of representative images of the database content. These images can be selected as the centroid images of each cell at a selected level of the hierarchical grid structure. In figure 3a browsing view of paper defects that is based on a grid-structure in the feature space is presented. This is an example of the use of the hierarchical browsing: When a desired image type is found at a certain level, the associative clusters can be taken into closer inspection (they can be zoomed in). Hence, two clusters are selected and the contents of them are presented in figures 3a and 3b. This way the user can browse the image types of his/her interest. The size of the browsing view can be changed by selecting different levels in the hierarchical structure.

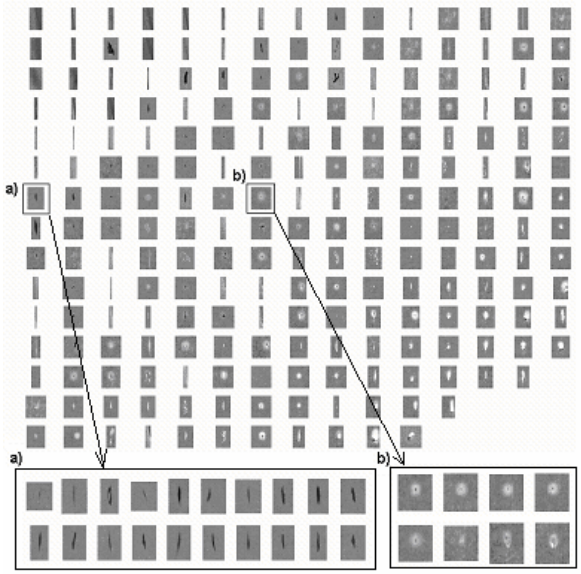


Figure 3. An example of the browsing view of the paper defect images. From this image, two clusters (a and b) are selected to a closer inspection. The contents of these clusters are presented in subfigures a and b.

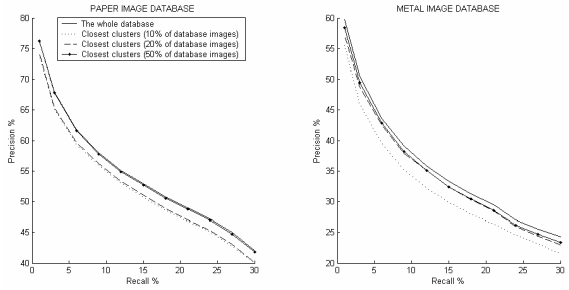


Figure 4. Average precision/recall curves of the queries made in the both image databases.

Grid-based clustering is applied to fast retrieval. The idea is to divide the image database into clusters before the retrieval operations are performed. This way the retrieval operations can be applied only to the clusters that are relevant to the query image. The relevant clusters are selected by comparing the query image to the representative (centroid) images of each cluster. The retrieval experiments were made using each database image as a query image in turn. In each query, we selected a set of clusters, which were closest to the query image. Because with the real image data the size of the clusters is very varying, the number of the clusters to be selected to the retrieval set varies also strongly. The number of selected clusters was limited so that the set of closest clusters contained only a certain number of images. The experiments were carried out using three sizes of cluster sets consisting of 10, 20, and 50% of the database images. In the experiments, retrieval accuracy was measured by defining an average precision/recall curve for each query

Table 1. Computing time of the clustering in the large paper defect database of 10 500 images.

| | Computing time (sec) | |
|---|---|---|
| Number of clusters | Grid-clustering | *k*-means |
| 10 | 1,3 | 46,2 |
| 20 | 2,3 | 82,2 |
| 30 | 2,9 | 121,2 |
| 40 | 4,5 | 130,3 |
| 50 | 6,3 | 150,6 |

(figure 4). In this figure, the retrieval comparison is made to the whole (unclustered) database. The figure shows that the precision value is almost the same as in the case of the whole database search, when the retrieval is applied for 50% of the database. In addition, using only 10% of the images, precision is not impaired significantly (but retrieval time is decreased 90%, because only 10% of the database is searched).

### 3.2. Computational cost

The computational complexity of our clustering method was measured using a large, unlabeled paper defect image database that contained about 10 500 images. Table 1 presents the computing time of the clustering for different numbers of clusters. The computing time of grid-based clustering is compared to that of *k*-means clustering. The computation was made using Matlab on a PC with 2.4 GHz Pentium 4 CPU and 523 MB primary memory.

Table 1 shows that there is a significant difference between the computational cost of the grid-based clustering and *k*-means algorithm. The reason for this difference is that *k*-means is an iterative algorithm, which makes it heavy, especially in the case of large databases.

## 4. DISCUSSION

In this paper we have presented a grid-based approach to the organization of image databases. We presented two practical applications for the use of this technique, image database browsing and fast retrieval. In the browsing, the hierarchical cluster structure makes it possible to zoom in, to select a specific image type to closer inspection, or zoom out, to browse the variations of the image database content. The second application is fast image retrieval. Especially in the case of large image databases, it is time-consuming to go through the whole database at each query. However, when the queries are carried out in the clustered database, the queries can be focused only in the clusters that contain most relevant images. Hence the whole database is not needed to be searched, which saves computational time in the retrieval.

In the case of large image databases (>10 000 images), the clustering method should be fast. We compared the speed of our grid-based clustering method to *k*-means algorithm that has been used in hierarchical image browsing for example in [2]. The results presented in figure 5 show that grid-based approach outperforms *k*-means algorithm in computational lightness.

In conclusion, the grid-based clustering procedure proved to be effective in the content-based organization of large image databases. Therefore this method is suitable to be used in the retrieval and browsing of large image databases, which require fast processing time.

## 5. ACKNOWLEDGMENT

The authors wish to thank the Technology Development Centre of Finland (TEKES's grant 40397/01) for financial support.

## 6. REFERENCES

[1] R. Agrawal, J. Gehrke, D. Gunopulos, P. Raghavan, "Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications", *Proceedings of 1998 ACM-SIGMOD*, pp. 94-105, 1998.

[2] J.-Y. Chen, C. A. Bouman, J. C. Dalton, "Hierarchical Browsing and Search of Large Image Databases", *IEEE Transactions on Image Processing*, Vol. 9, No. 3, pp. 442-455, March 2000.

[3] J. Han, M. Kamber, *Data Mining: Concepts and Techniques*, Academic Press, San Diego, 2001.

[4] J. Iivarinen and A. Visa, "An adaptive texture and shape based defect classification," *Proceedings of the 14th International Conference on Pattern Recognition*, Vol. 1, pp. 117-122, 1998.

[5] I. Kunttu, L. Lepistö, J. Rauhamaa, A. Visa, "Image Correlogram in Image Database Indexing and Retrieval", *Proceedings of 4th European Workshop on Image Analysis for Multimedia Interactive Services*, pp. 88-91, 2003.

[6] I. Kunttu, L. Lepistö, J. Rauhamaa, and A. Visa, "Classification Method for Defect Images Based on Association and Clustering", *Proceedings of SPIE*, Vol. 5098, pp. 19-27, 2003.

[7] B. S. Manjunath, J.-R. Ohm, V. V. Vasuvedan, A. Yamada, "Color and Texture Descriptors", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 11, No. 6, pp. 703-715, 2001.

[8] J. Rauhamaa, and R. Reinius, "Paper Web Imaging with Advanced Defect Classification", *Proceedings of the 2002 TAPPI Technology Summit*, 2002.

[9] E. Schikuta, "Grid-Clustering: An Efficient Hierarchical Clustering Method for Very Large Data Sets", *Proceedings of the 13th International Conference on Pattern Recognition*, Vol. 2, pp. 101-105, 1996.

[10] C. S. Warnekar, G. Krishna, "A Heuristic Clustering Algorithm Using Union of Overlapping Pattern-Cells", *Pattern Recognition*, Vol. 11, No. 2, 1979 pp. 85-93