ELSEVIER

Pattern Recognition Letters 20 (1999) 899–909

Pattern Recognition Letters

www.elsevier.nl/locate/patrec

Histogram clustering for unsupervised segmentation and image retrieval ☆

Jan Puzicha [a,1], Thomas Hofmann [b,2], Joachim M. Buhmann [a,*]

[a] *Institut für Informatik III, Rheinische Friedrich-Wilhelms-Universität, Romerstrasse 164, 53117 Bonn, Germany*

[b] *Computer Science Division, University of California and International Computer Science Institute, Berkeley, CA, USA*

Received 10 November 1998; received in revised form 11 May 1999

Abstract

This paper introduces a novel statistical latent class model for probabilistic grouping of distributional and histogram data. Adopting the Bayesian framework, we propose to perform annealed maximum a posteriori estimation to compute optimal clustering solutions. In order to accelerate the optimization process, an efficient multiscale formulation is developed. We present a prototypical application of this method for unsupervised segmentation of textured images based on local distributions of Gabor coefficients. Benchmark results indicate superior performance compared to K-means clustering and proximity-based algorithms. In a second application the histogram clustering method is utilized to structure image databases for improved image retrieval. © 1999 Elsevier Science B.V. All rights reserved.

Keywords: Histogram clustering; Texture segmentation; Multiscale annealing; Image retrieval

1. Introduction

Grouping, segmentation, coarsening, and quantization are central and omnipresent topics in image processing and computer vision. In the unsupervised case, these tasks are essentially different instances and variations of the *clustering* or *grouping* problem, i.e., the general goal is to identify groups of *similar* image primitives. Depending on the specific problem these primitives can be entities like pixels, line elements, image patches and regions, real-world objects, or even complete images being part of an image sequence or an image database.

Two fundamental steps need to be addressed:

- *Modeling problem.* A precise mathematical notion of homogeneity or similarity between image primitives is required in order to formalize the clustering problem.
- *Computational problem.* For a given similarity measure, an efficient clustering algorithm has to be derived. The selection of a suitable clustering method is tightly coupled to the chosen similarity measure and its underlying data representation.

In this contribution, we mainly focus on the unsupervised segmentation of textured images as a prototypical application in low-level computer vision. Here the goal is to group pixels or small

☆ Electronic Annexes available. See www.elsevier.nl/locate/patrec.

* Corresponding author. Tel.: +49-228-73-4380; fax: +49-228-73-4382; e-mail: jb@cs.uni-bonn.de

[1] E-mail: jan@cs.uni-bonn.de

[2] E-mail: hofmann@cs.berkeley.edu

PII: S 0 1 6 7 - 8 6 5 5 (9 9) 0 0 0 5 6 - 2

image patches such that segments of identical texture are obtained.

Numerous techniques to unsupervised texture segmentation have been proposed over the past decades. In many classical approaches, local features are spatially smoothed and represented as *vectors in a metric space* (e.g., in (Jain and Farrokhnia, 1991; Mao and Jain, 1992)), thereby characterizing each texture by a specific average feature vector or *centroid*. The most commonly used distortion measure is a (weighted) squared Euclidean norm which effectively models the data by a Gaussian mixture model with one Gaussian for each texture. The clustering method of choice for vectorial data is the *K*-means algorithm and its variants. Since the Gaussian mixture assumption turns out to be inadequate in many cases, several alternative approaches have been utilized. One important class of methods is based on *proximity data*, usually obtained by applying statistical tests to the *local feature distribution* at two image sites (Geman et al., 1990; Hofmann et al., 1998; Ojala and Pietikäinen, 1998). As a major advantage, these methods do not require the specification of a suitable vector-space metric, since similarity is defined directly via the respective feature distributions. Agglomerative clustering (Ojala and Pietikäinen, 1998) and, more rigorously, optimization approaches to graph partitioning (Geman et al., 1990; Hofmann et al., 1998; Shi and Malik, 1997) have been proposed as clustering techniques, which we refer to as *pairwise dissimilarity clustering* (PDC).

The major contribution of this paper is a general method for grouping *feature distributions*, extending a technique known as distributional clustering in statistical language modeling (Pereira et al., 1993). In contrast to approaches based on feature vectors and proximities, this method performs grouping directly on the histogram or distributional data and does not require re-representing the data, neither as points in a metric space nor in terms of pairwise dissimilarities. Compared to *K*-means clustering, distributional clustering naturally includes component distributions with multiple modes and thus offers a higher modeling flexibility. As a major advantage over PDC it requires no external similarity measure to compare local feature statistics. In addition, distributional clustering provides a generative statistical model and gives an explicit characterization of textures that can be utilized in subsequent processing steps such as boundary localization (Schroeter and Bigun, 1995) or even for texture synthesis (Zhu et al., 1998). Distributional clustering also offers computational advantages, because it can be implemented efficiently by multiscale optimization techniques (Puzicha and Buhmann, 1998). Since feature histograms are processed directly, it avoids time-consuming stages of data extraction (e.g., pairwise comparisons in PDC), which is crucial in real-time applications like autonomous robotics.

To stress the broad applicability of the histogram clustering algorithm, we also demonstrate its utility for coarsening and structuring of large image databases. In image retrieval, database items are typically represented by feature histograms, e.g. color and/or texture (Flickner et al., 1995; Picard et al., 1993; Hofmann et al., 1998) histogram-based clustering techniques can thus be used to guide and accelerate retrieval by grouping similar images and by identifying representative image prototypes. Moreover, the imposed group structure can be utilized to accelerate dissimilarity computations and even to improve retrieval performance (Willett, 1988).

2. Latent class models for histogram data

2.1. Specification of the asymmetric clustering model

To stress the generality of the proposed model we temporarily detach the presentation from the specific problem of image segmentation. Consider therefore the following more abstract setting: Let $\boldsymbol{X} = \{x_1, \dots, x_N\}$ denote a finite set of abstract objects with arbitrary labeling and let $\boldsymbol{Y} = \{y_1, \dots, y_M\}$ represent a domain of nominal features. The elementary observations consist of *dyadic* measurements $(x, y) \in \boldsymbol{X} \times \boldsymbol{Y}$, i.e., *joint occurrences* of elements from $\boldsymbol{X}$ and $\boldsymbol{Y}$, where we have used the simplified notation x and y to refer to generic elements from $\boldsymbol{X}$ and $\boldsymbol{Y}$, respectively. All observations are summarized in an N by M rectangular table $\boldsymbol{n}$ of counts $n(x,y)$, where $n(x,y)$ encodes the number of times an observation (x,y) has been made, i.e., how often a feature y has been observed for a particular x. Effectively, this defines for each x an *empirical distribution* or *histogram* over the feature set $\boldsymbol{Y}$ given by $\hat{P}(y \mid x) \equiv n(x,y)/n(x)$, where $n(x) \equiv \sum_{y \in \boldsymbol{Y}} n(x,y)$ is the number of observations for object x.

The proposed latent class model, which is referred to as *one-sided* or *Asymmetric Clustering Model* (ACM) [3] presupposes a latent class or cluster structure in the $\boldsymbol{X}$ space which is represented by a mapping $\boldsymbol{c} : \boldsymbol{X} \to \boldsymbol{C}$, $\boldsymbol{C} = \{c_1, \ldots, c_K\}$. Here $\boldsymbol{C}$ is a set of labels referring to the different clusters. Hence $\boldsymbol{c}(x)$ denotes the latent class associated with a particular object x. Furthermore, let us introduce for each cluster $c \in \boldsymbol{C}$ a prototypical (multinomial) probability distribution $P(y \mid c)$ over the feature space $\boldsymbol{Y}$ as well as a marginal distribution $P(x)$ on $\boldsymbol{X}$, summarized in a parameter vector θ. Given θ and the mapping $\boldsymbol{c}$ one may define a generative model of the data according to the following sampling scheme:

1. select an object $x \in \boldsymbol{X}$ with probability $P(x)$,
2. determine the latent class c according to the cluster membership $\boldsymbol{c}(x)$ of x,
3. select $y \in \boldsymbol{Y}$ from the cluster-specific conditional distribution $P(y \mid c)$.

According to the generative model, observations are assumed to be independent conditioned on the continuous parameters and the cluster assignments. Hence, the conditional probability to observe (x,y) is given by

$$P(x,y \mid \boldsymbol{c},\theta) = P(x)P(y \mid \boldsymbol{c}(x)). \tag{1}$$

Taking a Bayesian perspective, we introduce prior distributions $P(\boldsymbol{c},\theta)$ for all quantities which define the data generation process. By assuming prior independence of $\boldsymbol{c}$ and θ and by putting a non-informative uniform prior on θ, the posterior distribution is – up to a proper normalization – given by

$$P(\boldsymbol{c},\theta \mid \boldsymbol{n}) \propto P(\boldsymbol{c}) \prod_{x \in \boldsymbol{X}} P(x)^{n(x)} \prod_{y \in \boldsymbol{Y}} P(y \mid \boldsymbol{c}(x))^{n(x,y)}. \tag{2}$$

Returning to the texture segmentation problem, we identify $\boldsymbol{X}$ with the set of image locations or sites and $\boldsymbol{Y}$ with possible values of discrete or discretized local texture features computed from the image data. The distribution $\hat{P}(y \mid x)$ then represents a histogram of features occurring in an *image neighborhood* or *window* around some location x (cf. (Geman et al., 1990; Hofmann et al., 1998; Ojala and Pietikäinen, 1998)). Each class $c \in \boldsymbol{C}$ corresponds to a different texture which is characterized by a specific distribution $P(y \mid c)$ of features y. Since these multinomials are not constrained, they can virtually model any distribution of features. In particular, no further parametric restrictions on $P(y \mid c)$ are imposed. There is also no need to specify an additional noise model or, equivalently, a metric in feature space (Pereira et al., 1993). It should be emphasized that the class model and thus the optimal class assignment of a site x does not depend on the pixel position of x. The optimal class assignment only depends on the locally measured features.

2.2. Parameter estimation for the ACM

Taking $P(\boldsymbol{c},\theta \mid \boldsymbol{n})$ in (2) as a starting point, we propose to compute *maximum* a posteriori estimates, i.e., we have to maximize the log-posterior distribution which is equivalent to maximizing

$$\begin{aligned} \boldsymbol{L}(\boldsymbol{c},\theta;\boldsymbol{n}) &= \log P(\boldsymbol{c}) + \sum_{x \in \boldsymbol{X}} n(x) \log P(x) \\ &\quad + \sum_{x \in \boldsymbol{X}} \sum_{y \in \boldsymbol{Y}} n(x,y) \log P(y \mid \boldsymbol{c}(x)) \\ &= \log P(\boldsymbol{c}) + \sum_{x \in \boldsymbol{X}} n(x) \Bigg[\log P(x) \\ &\quad + \sum_{y \in \boldsymbol{Y}} \hat{P}(y \mid x) \log P(y \mid \boldsymbol{c}(x)) \Bigg]. \end{aligned} \tag{3}$$

Stationary equations are derived from (3) by differentiation. Using Lagrange parameters to ensure

[3] It is called asymmetric because clustering structure is inferred solely in the $\boldsymbol{X}$ space. The ACM is the most suitable model for histogram clustering out of a family of novel mixture models developed for general dyadic data (Hofmann and Puzicha, 1998).

a proper normalization of the continuous model parameters θ we obtain

$$P(x) = \frac{n(x)}{\sum_{x' \in \boldsymbol{X}} n(x')}, \tag{4}$$

$$P(y \mid c) = \sum_{x : \boldsymbol{c}(x) = c} \frac{n(x)}{\sum_{x' : \boldsymbol{c}(x') = c} n(x')} \hat{P}(y \mid x), \tag{5}$$

$$\boldsymbol{c}(x) = \arg \min_{a} \left\{ - \sum_{y \in \boldsymbol{Y}} \hat{P}(y \mid x) \log P(y \mid a) + \log P\left(\boldsymbol{c}_a^x\right) \right\}, \tag{6}$$

where $\boldsymbol{c}_a^x(x) = a$ and $\boldsymbol{c}_a^x(x') = \boldsymbol{c}(x')$ for $x' \neq x$ denotes the class assignments obtained by changing the assignment of x to class a. From (4) we see that the probabilities $P(x)$ are estimated independently of all other parameters. The maximum a posteriori estimates of the class-conditional distributions $\hat{P}(y \mid \boldsymbol{c})$ are linear superpositions of all empirical distributions for objects x belonging to cluster $\boldsymbol{c}$. Eq. (5) thus generalizes the *centroid condition* from *K*-means clustering to distributional clustering. Notice however, that the components of $P(y \mid c)$ define probabilities for feature values and do not correspond to dimensions in the original feature space; Eq. (5) averages over feature *distributions*, not over feature *values*. The formal similarity to *K*-means clustering is extended by (6), which is the analogon to the nearest-neighbor rule. In the maximum likelihood case, i.e., with a uniform prior $P(\boldsymbol{c})$, the optimal assignments for given parameters can be estimated in one step. In this case, the analogy between the stationary conditions for the ACM and for *K*-means clustering also holds for the model fitting algorithm. The likelihood can be maximized by an *alternating maximization* (AM) update scheme which calculates assignments for given centroids according to the nearest-neighbor rule (6) and recalculates the centroid distributions (5) in alternation. Both algorithmic steps increase the likelihood, and convergence to a (local) maximum of (3) is thus ensured. For general non-factorial $P(\boldsymbol{c})$, however, the assignments $\boldsymbol{c}(x)$ of different sites are coupled and (6) has to be iterated until convergence in analogy to the iterated conditional mode (ICM) algorithm.

2.3. Distributional clustering

In the maximum-likelihood case the ACM is similar to the distributional clustering model formulated by Pereira et al. (1993) as the minimization of the cost function

$$\boldsymbol{H}(\boldsymbol{c}, \theta; \boldsymbol{n}) = \sum_{x \in \boldsymbol{X}} D_{\mathrm{KL}} \left[\hat{P}(\cdot \mid x) \| P(\cdot \mid \boldsymbol{c}(x)) \right]. \tag{7}$$

Here D_{KL} denotes the cross entropy or Kullback–Leibler (KL) divergence. [4] Distributional clustering thus aims at finding prototypical ‘centroid’ distributions $P(y \mid c)$ and a partitioning $\boldsymbol{c}$ of $\boldsymbol{X}$ such that the total distortion between the prototype distributions of a cluster and the empirical distributions of objects assigned to it is minimized. In distributional clustering, the KL-divergence as a distortion measure for distributions has been motivated by the fact that the centroid equation (5) is satisfied at stationary points. Yet, after dropping $P(x)$ and $P(\boldsymbol{c})$ in (3) and ignoring a (data dependent) constant, we derive the formula

$$\boldsymbol{L}(\boldsymbol{c}, \theta; \boldsymbol{n}) = - \sum_{x \in \boldsymbol{X}} n(x) D_{\mathrm{KL}} [P(\cdot \mid x) \| P(\cdot \mid \boldsymbol{c}(x))]. \tag{8}$$

This proves that the choice of the KL-divergence as a distortion measure simply follows from the likelihood principle.

2.4. Deterministic annealing

A technique which allows us to improve the presented AM procedure by avoiding unfavorable local minima is known as *deterministic annealing* (DA). The key idea is to introduce a temperature parameter T and to replace the minimization of a combinatorial objective function by a substitute known as the *generalized free energy*. Details on this topic in the context of data clustering can be

[4] The KL divergence between two multinomials $P(y)$ and $Q(y)$ is an asymmetric measure defined as $D_{\mathrm{KL}}[P(\cdot) \| Q(\cdot)] = \sum_y P(y) [\log P(y) - \log Q(y)]$.

found in (Rose et al., 1990; Pereira et al., 1993; Hofmann et al., 1998; Puzicha et al., 1999). Minimization of the free energy corresponding to (8) yields the following equations for probabilistic assignments:

$$P\Big(\boldsymbol{c}(x) = a \mid \boldsymbol{n}, \hat{\theta}\Big) = \frac{\exp\Big(-n(x) D_{\mathrm{KL}}\Big[\hat{P}(\cdot \mid x) \| P(\cdot \mid a)\Big] \Big/ T\Big)}{\sum_{b \in \boldsymbol{C}} \exp\Big(-n(x) D_{\mathrm{KL}}\Big[\hat{P}(\cdot \mid x) \| P(\cdot \mid b)\Big] \Big/ T\Big)}. \tag{9}$$

This partition of unity is a very intuitive generalization of the nearest-neighbor rule in (6). For $T \to 0$ the arg-min operation performed in the nearest-neighbor rule is recovered. Since solutions in DA are tracked from high to low temperatures, we finally maximize the log-likelihood with respect to both, θ and $\boldsymbol{c}$, at $T = 0$. Notice that the DA procedure also generalizes the Expectation Maximization (EM) algorithm obtained for $T = 1$, in which $\boldsymbol{c}$ is treated as an unobserved variable. In the latter case (9) corresponds to the computation of posterior probabilities in the E-step. For the general case in (3) a more complex coupled system of transcendental equations for the posteriors $P(\boldsymbol{c}(x) = a \mid \boldsymbol{n}, \hat{\theta})$ is obtained. This system of equations is solved by a convergent iterative scheme (Hofmann et al., 1998). As an additional advantage it has been demonstrated empirically (Hofmann and Puzicha, 1998) and theoretically (Buhmann and Puzicha, 1999) that DA with finite stopping temperature $T > 0$ can be utilized to efficiently avoid data over-fitting.

2.5. Multiscale annealing

It is a natural assumption for most domains that adjacent image sites contain identical textures with high probability. This fact can be exploited to significantly accelerate the optimization of the likelihood by maximizing over a suitable nested sequence of subspaces in a coarse-to-fine manner, where each subspace has a greatly reduced number of class assignment variables. This strategy is formalized by the concept of *multiscale optimization* (Heitz et al., 1994; Nicholls and Petrou, 1993; Puzicha and Buhmann, 1998) which in essence leads to cost functions redefined on a coarsened version of the original image. In contrast to most multi-resolution optimization schemes the *original* cost function is optimized at all grids, only the configuration space is reduced by variable tying. We first sketch the general theory and then derive multiscale equations for histogram clustering.

Formally, we denote by $\boldsymbol{X}^0 = \boldsymbol{X}$ the original set of sites and we assume that a set of grid sites $\boldsymbol{X}^l = \{x_1^l, \ldots, x_{N^l}^l\}$ is given for each coarse grid level l. Typically, $\boldsymbol{X}^0 = \boldsymbol{X}$ corresponds to the set of pixel sites and $\boldsymbol{X}^{l+1}$ is obtained by sub-sampling $\boldsymbol{X}^l$ by a factor of 2 in each direction. Multiscale optimization proceeds not by coarsening the image, but by *coarsening the variable space*. Each coarse grid is associated with a reduced set of assignment variables $\boldsymbol{c}^l$. Thus a single variable $\boldsymbol{c}^l(x^l)$ is attached to each grid point x^l coding the texture class of the set of respective pixels. Restricting the consideration for notational convenience to the simplified model (8) the following coarse grid log-likelihood functions for $l \geqslant 1$ are obtained:

$$\boldsymbol{L}^l\big(\boldsymbol{c}^l, \theta; \boldsymbol{n}\big) = \sum_{x^l \in \boldsymbol{X}^l} \sum_{y \in \boldsymbol{Y}} n^l(x^l, y) \log P\big(y \mid \boldsymbol{c}^l(x^l)\big), \tag{10}$$

where $n^l(x^l, y)$ are the pooled counts $n(x, y)$ for all $x \in \boldsymbol{X}^0$ which are tied to x^l. Note, that $\boldsymbol{L}^l$ has the same functional form as $\boldsymbol{L}^0 = \boldsymbol{L}$ and, therefore, an optimization algorithm developed for $\boldsymbol{L}$ is applicable to any coarse grid cost function $\boldsymbol{L}^l$. Finally, after a segmentation $\boldsymbol{c}^{l+1}$ has been computed on the coarse grid $\boldsymbol{X}^{l+1}$, the solution is prolongated to the next level by initializing the variables $\boldsymbol{c}^l(x^l)$ from their associated variables $\boldsymbol{c}^{l+1}(x^{l+1})$.

Traditionally, clustering algorithms like K-means efficiently incorporate *splitting techniques* to obtain successive solutions for a growing number of clusters. We adopt this idea by successively splitting clusters with high distortion. Since the number of data objects is drastically reduced at coarser resolution levels, splitting strategy and coarse-to-fine optimization are interleaved. The question of choosing the maximal number of clusters for a given resolution has been addressed in a statistical learning theory context by

Buhmann and Puzicha (1999). We adopt these results by choosing $K^l_{\max} \sim N^l / \log N^l$ and selecting the proportionality factor on an empirical basis.

One of the key advantages of the DA approach is the *inherent splitting strategy*. Clusters degenerate at high temperature and they successively split at *bifurcations* or *phase transitions* when T is lowered (Rose et al., 1990). Therefore at a specific temperature scale an (easily measurable) *effective number* K_T of clusters is visible. For a given resolution level l we anneal until K_T exceeds a predefined maximal number of clusters $K^l_{\max}$ at a certain temperature level T^*. After prolongation to level $l-1$ the DA optimization is continued at temperature T^*. This scheme has been introduced as *multiscale annealing* by Puzicha and Buhmann (1998).

3. Results

3.1. Implementation details

We applied the ACM to the unsupervised segmentation of textured images, where objects x correspond to image locations. Since the number of observed features is identical for all sites, one can simply set $P(x) = 1/N$. In the experiments, we have adopted the framework of Jain and Farrokhnia (1991), Hofmann et al. (1998), Puzicha and Buhmann (1998) and utilized an image representation based on the modulus of complex Gabor filters. For each site the empirical distribution of coefficients in a surrounding (filter-specific) window is determined. All reported segmentation results are based on a filter bank of twelve Gabor filters with four orientations and three scales. Each filter output was discretized into 16 equally sized bins. As a consequence of the conditional independence assumptions this results in a feature space $\boldsymbol{Y}$ of size $M = 192$. For each channel 256 Gabor coefficients were sampled in a local window of a size proportional to the scale of the filter. For the finest scale a rectangular 16×16 window was utilized. The final segmentations are computed on a 128×128 grid. The benchmark results are obtained on images which were generated from a representative set of 86 micro-patterns taken from the Brodatz texture album. A database of random mixtures (512×512 pixels each) containing 100 entities of five textures each (as depicted in Fig. 1) was constructed.

For most applications there is prior knowledge about inadmissible or unlikely texture label configurations. For example, segmentation results can be improved by suppressing small and highly fragmented regions. As a quality criterion we propose to count for each image site how many sites of the same class label are found in a small topological neighborhood. When the number of identically labeled pixels drops below a threshold, the label configuration is considered to be less

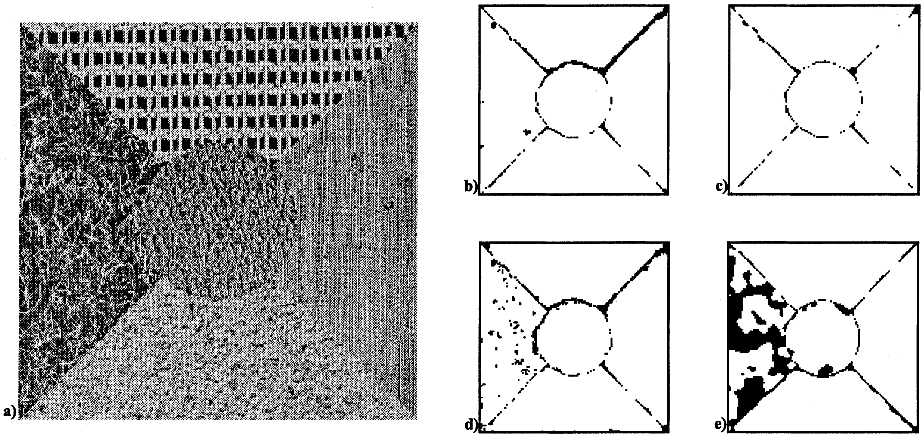

Fig. 1. Typical segmentation results with $K = 5$ for the algorithms under examination: (a) original image, (b) ACM with uniform prior, (c) ACM with topological prior (11), (d) PDC and (e) K-means. Misclassified blocks w.r.t. ground truth are depicted in black.

Table 1
Errors by comparison with ground truth over 100 randomly generated images with $K = 5$ textures, 512×512 pixels and 128×128 sites. For all algorithms multiscale annealing has been used for optimization. The median run-time over 100 images has been measured on a Pentium II 300 MHz

	Median (%)	20% quantile (%)	Run-time (s)
ACM with uniform prior	5.1	5	3.5
ACM with topological prior (11)	2.8	5	27.4
Pairwise clustering	5.8	8	3.6
Normalized cut	5.9	8	2.5
K-means clustering	6.9	6	0.94

likely, the log-likelihood being proportional to the difference to the threshold. The probability distribution

$$P(\boldsymbol{c}) = \frac{1}{Z}\exp\left(-\lambda H^p(\boldsymbol{c})\right), \qquad H^p(\boldsymbol{c}) = \sum_{x \in \boldsymbol{X}} (B - A(x))_+ \tag{11}$$

makes these considerations mathematically precise, where B denotes a threshold, $A(x)$ the number of neighboring sites assigned to the same class, Z a normalization parameter, λ a weighting parameter and $(\cdot)_+$ is the truncation function at zero. In the experiments we used a large value for λ to suppress small regions completely where small regions have been defined via $B = 10$ and a 7×7 topological neighborhood. [5] It is worth to note that for this, topological prior multiscale techniques are mandatory for optimization as (11) effectively erects barriers in the search space which cannot be traversed solely by single-site changes.

For comparison we utilized *K*-means and two proximity-based clustering algorithms. For the *K*-means algorithm a spatial smoothing step was applied to the modulus of the Gabor coefficients before clustering (cf. Jain and Farrokhnia, 1991). The PDC algorithm is based on a normalized cost function, which is invariant to linear transformation of the proximity data (Hofmann et al., 1998; Puzicha et al., 1999). The *normalized cut* has been developed only for $K = 2$ (Shi and Malik, 1997). The corresponding normalized cost function is equivalent to the normalized association which generalizes to $K > 2$ (Puzicha et al., 1999) and which has been used in the experiments. The χ^2 test statistic applied to the Gabor channel histograms was utilized to obtain proximity data, where we have computed approximately 80 randomly selected dissimilarity scores for each image site (Shi and Malik, 1997; Hofmann et al., 1998). The multiscale annealing technique was utilized for all clustering cost functions.

3.2. Segmentation results

The question examined in detail is concerned with the benefits of the ACM in comparison to other clustering schemes. A typical example with $K = 5$ clusters is given in Fig. 1. The error plots demonstrate that the segmentations achieved by ACM have the highest quality. Most errors occur at texture boundaries where texture information is mixed due to the spatial support of the Gabor filters and the extent of the neighborhood used for computing the local feature statistics. The *K*-means clustering cost function exhibits substantial deficits to correctly model the segmentation task. These observations are confirmed by the benchmark results in Table 1. We report the median, since the distributions of the empirical errors are highly asymmetric. In addition, the percentage of segmentations with an error rate larger than 20% is reported, which we define as the percentage of segmentations where the essential structure has not been detected. For the ACM, a median error of 5.1% has been achieved compared to 5.7% for PDC and 5.8% for the normalized cut. The percentage of segmentations, where the essential structure has been detected, is highest for the

[5] These parameters reflect the specific prior knowledge about the expected region size in the image. However, small variations of B hardly affect the segmentation results.

ACM with 95%. Moreover, using the topological prior (11) further improves the segmentation results, which leads to a median error as low as 2.8%. The *K*-means model yields significantly worse results with a median error of 6.9%. The excellent segmentation quality obtained by the ACM histogram clustering algorithm is confirmed by the results on more difficult segmentation tasks in Fig. 2. The mixture of $K = 16$ different Brodatz textures has been partitioned accurately with an error rate of 4.7%. The errors basically correspond to boundary sites. The results obtained for the mondrians of aerial images are also satisfactory: disconnected texture regions of the same type have been identified correctly, while problems again occur at texture boundaries. The segmentation quality achieved on outdoor images in Figs. 3 and 4 are both visually and semantically satisfying.

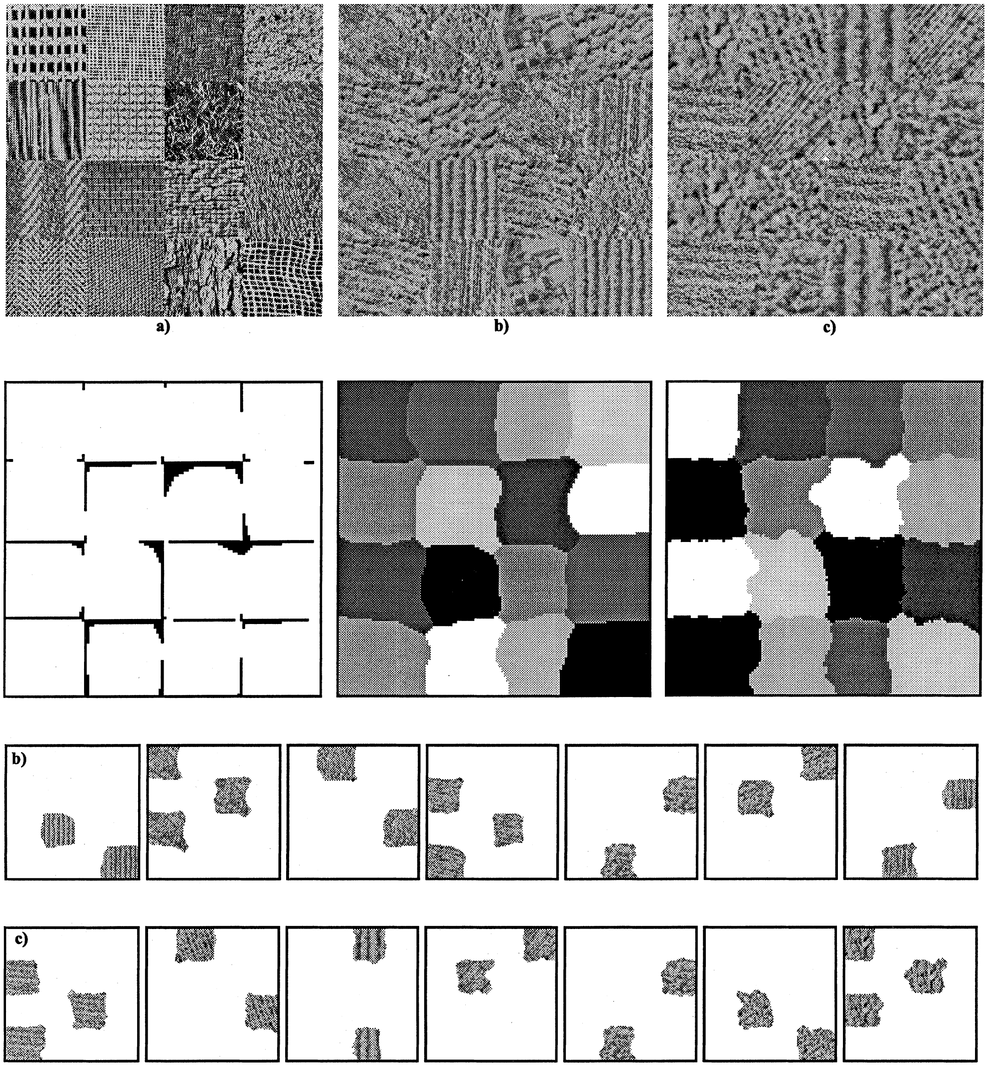

Fig. 2. Typical segmentation results obtained by ACM with topological prior: (a) on a mondrian of 16 different Brodatz textures (misclassified blocks w.r.t. ground truth are depicted in black), (b) and (c) mondrians of seven different textures taken from aerial images.

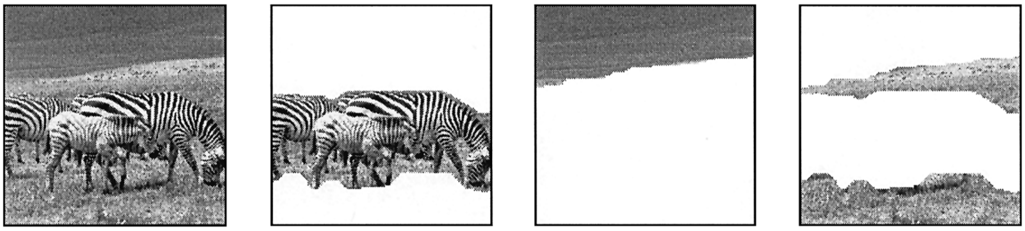

Fig. 3. Typical segmentation result on a real-world image with $K = 3$ segments obtained by ACM with topological prior.

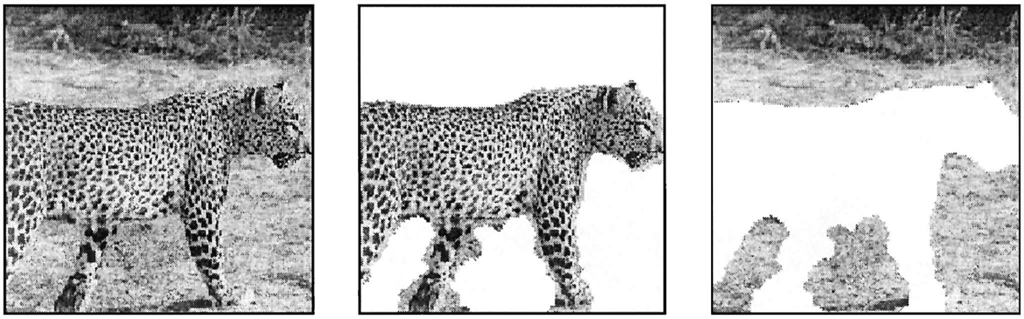

Fig. 4. Typical segmentation result on a real-world image with $K = 2$ segments obtained by ACM with topological prior.

3.3. Structuring image databases

In the experiment on image retrieval, we used 60 samples from the Brodatz album to create a database of homogeneously textured images. Each of the 60 sample images has been divided into 256 partly overlapping sub-images of size 64×64 resulting in a database with 15 360 images. For each item in the database a histogram-based texture descriptor is extracted as in the segmentation application. Clustering solutions with 60 groups for this database are then compared with the known ground truth data in order to evaluate the quality of a grouping algorithm.

It is obvious from Table 1 that multiscale annealing enables highly efficient optimization for image segmentation. For clustering of databases, however, there is no canonical coarsening procedure and the developed multiscale techniques are not directly applicable. Therefore, the following alternative approach has been utilized:

1. A hierarchical, fast clustering technique (e.g. agglomerative clustering) is applied to compute a low-quality clustering solution. We utilized Ward's method in the experiments.
2. The computed hierarchy then serves as a topology to generate coarse grids. While agglomerative techniques compute inferior solutions, objects joined in an early stage of the hierarchy are still expected to have a high probability to belong to the same cluster in the optimal partition.

In Table 2 we compare the results for a clustering solution obtained by Ward's method with partitionings obtained by single scale and multiscale ACM clustering. The most important results can be summarized as follows:

- Ward's method is clearly outperformed in terms of quality. ACM-based clustering performs very well, leading to an error rate as low as 6.62% for 60 groups compared to 21.2% for the agglomerative technique.
- The ICM-based local optimization process is accelerated by a factor of 12.5 in terms of overall run-time when the hierarchy computed by Ward's method is exploited as a topology to generate coarse grids.
- Using multiscale annealing techniques it becomes possible to further improve clustering quality. Multiscale annealing results only in a

Table 2
Error-rates and run-time in minutes for structuring of an image database. The overall run-time includes the computation of pairwise comparisons with 480 neighbors on average for Ward's method and the total run-time of Ward's method for multiscale optimization to compute the topology for the multiscale operator

	Error-rate (%)	Run-time	Overall run-time (min)
Ward's method	21.1	3.7 min	15.4
Single scale ICM	7.6	336.0 min	336.0
Single scale annealing		> 7 days	
Multiscale ICM	6.7	11.5 min	26.9
Multiscale annealing	6.6	14.2 min	29.6

Table 3
Advantages and disadvantages of the clustering algorithms: *K*-means, histogram clustering, pairwise dissimilarity clustering/normalized cut

	K-means	ACM	PDC / NC
Underlying data type	Vector	Histogram	Proximity
Computational complexity of data extraction	Lowest	Medium	Highest
Computational complexity of optimization	Lowest	Medium	Medium
Segmentation quality	Lowest	Highest	Medium
Generative statistical model provided	Yes	Yes	No
Implementation effort	Low	Low	High

slight increase in terms of run-time compared to multiscale ICM, while the experiments with single scale annealing have been interrupted due to exorbitant run-time.

4. Conclusion

The ACM histogram clustering model combines the expressive power of PDC with the efficiency of the conventional *K*-means clustering and provides a fast, accurate and reliable algorithm for unsupervised texture segmentation. As the ACM is an unsupervised method, no texture classes have to be defined a priori and no training set is necessary. The same algorithmic parameters have been used in all segmentation experiments demonstrating the robustness of the proposed algorithm. Compared to PDC and the normalized cut, the time-consuming algorithmic step of computing pairwise dissimilarities between objects has been avoided. Yet, the benchmark results strongly indicate that the ACM provides improved segmentation quality, especially when combined with a topological prior. Moreover, statistical texture class models are provided for subsequent processing steps making the ACM a highly interesting alternative to PDC and the normalized cut in the segmentation context. The generic applicability of the novel histogram clustering algorithm has been confirmed by the experiments on structuring large image databases. The advantages and disadvantages of all three clustering methods are summarized in Table 3. As a general clustering scheme this model can be extended to color and motion segmentation, region grouping and even integrated sensor segmentation simply by choosing appropriate features.

Acknowledgements

It is a pleasure to thank Hans du Buf for providing the aerial image mixtures in Fig. 2. This work has been supported by the German Research Foundation (DFG) under grant #BU 914/3–1, by an M.I.T. Faculty Sponser's Discretionary Fund and by an ICSI Postdoctoral Fellowship.

References

Buhmann, J., Puzicha, J., 1999. Unsupervised learning for robust texture segmentation. In: Proceedings of the Dagstuhl

Seminar on Empirical Evaluation of Computer Vision Algorithms (to appear).

Flickner, M. et al., 1995. Query by image and video content: The QBIC system. IEEE Computer (September) 23–32.

Geman, D., Geman, S., Graffigne, C., Dong, P., 1990. Boundary detection by constrained optimization. IEEE Trans. Pattern Anal. Machine Intell. 12 (7), 609–628.

Heitz, F., Perez, P., Bouthemy, P., 1994. Multiscale minimization of global energy functions in some visual recovery problems. CVGIP: Image Understanding 59 (1), 125–134.

Hofmann, T., Puzicha, J., 1998. Statistical models for co-occurrence data. AI-MEMO 1625. MIT Press, Cambridge.

Hofmann, T., Puzicha, J., Buhmann, J., 1998. Unsupervised texture segmentation in a deterministic annealing framework. IEEE Trans. Pattern Anal. Machine Intell. 20 (8), 803–818.

Jain, A., Farrokhnia, F., 1991. Unsupervised texture segmentation using Gabor filters. Pattern Recognition 24 (12), 1167–1186.

Mao, J., Jain, A., 1992. Texture classification and segmentation using multiresolution simultaneous autoregressive models. Pattern Recognition 25, 173–188.

Nicholls, G., Petrou, M., 1993. Multiresolution representation of Markov random fields. VSSP-TR-3/93 ACT-ST-272-89, Department of Electronic and Electrical Engineering, University of Surrey.

Ojala, T., Pietikäinen, M., 1998. Unsupervised texture segmentation using feature distributions. Pattern Recognition 32 (3).

Pereira, F., Tishby, N., Lee, L., 1993. Distributional clustering of English words. In: Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics, Columbus, OH, pp. 183–190.

Picard, R., Kabir, T., Liu, F., 1993. Real-time recognition of the entire Brodatz texture database. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 638–639.

Puzicha, J., Buhmann, J., 1998. Multi-scale annealing for real-time unsupervised texture segmentation. In: Proceedings of the International Conference on Computer Vision (ICCV'98), pp. 267–273.

Puzicha, J., Hofmann, T., Buhmann, J., 1999. A theory of proximity based clustering: Structure detection by optimization Pattern Recognition 33 (2).

Rose, K., Gurewitz, E., Fox, G., 1990. A deterministic annealing approach to clustering. Pattern Recognition Letters 11, 589–594.

Schroeter, P., Bigun, J., 1995. Hierarchical image segmentation by multi-dimensional clustering and orientation-adaptive boundary refinement. Pattern Recognition 28 (5), 695–709.

Shi, J., Malik, J., 1997. Normalized cuts and image segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'97), pp. 731–737.

Willett, P., 1988. Recent trends in hierarchical document clustering: a critical review. Inform. Process. Management 24 (5), 577–597.

Zhu, S., Wu, Y., Mumford, D., 1998. Frame: Filters, random field and maximum entropy: Towards a unified theory for texture modeling. Internat. J. Comput. Vision 27 (2).