Data Clustering as an Optimum-Path Forest Problem with Applications in Image Analysis

Leonardo Marques Rocha,[1] Fábio A. M. Cappabianco,[2] Alexandre Xavier Falcão[2]

[1] Department of Telecommunications, School of Electrical and Computer Engineering, University of Campinas, Brazil

[2] Department of Information Systems, Institute of Computing, University of Campinas, Brazil

Received 6 August 2008; accepted 5 March 2009

ABSTRACT: We propose an approach for data clustering based on optimum-path forest. The samples are taken as nodes of a graph, whose arcs are defined by an adjacency relation. The nodes are weighted by their probability density values (pdf) and a connectivity function is maximized, such that each maximum of the pdf becomes root of an optimum-path tree (cluster), composed by samples "more strongly connected" to that maximum than to any other root. We discuss the advantages over other pdf-based approaches and present extensions to large datasets with results for interactive image segmentation and for fast, accurate, and automatic brain tissue classification in magnetic resonance (MR) images. We also include experimental comparisons with other clustering approaches. Int J Imaging Syst Technol, 19, 50–68, 2009; Published online in Wiley InterScience (www.interscience.wiley.com). DOI 10.1002/ima.20191

Key words: optimum-path forest; clustering; image segmentation; meanshift; gm/wm classification

I. INTRODUCTION

The identification of natural groups of samples from a dataset, namely clustering (Duda et al., 2000) is a crucial step in many applications of data analysis. The samples are usually represented by feature vectors (e.g., points in $\Re^n$), whose similarity between them depends on a distance function (e.g., Euclidean). Natural groups are characterized by high concentrations of samples in the feature space, which form the domes of the probability density function (pdf), as illustrated in Figure 1a. These domes can be detected and separated by defining the "influence zones" of their maxima (Fig. 1b). However, there are different ways to define these influence zones (Cheng, 1995; Herbin et al., 1996) and the desired data partition may require to reduce the number of irrelevant clusters (Fig. 1c). To propose a more general and robust solution, we reformulate this strategy as an optimum-path forest problem in a graph derived from the samples.

The samples are nodes of a graph, whose arcs are defined by an adiacencv relation between them. The arcs are weighted by the distances between the feature vectors of their corresponding samples and the nodes are also weighted by their probability density values, which are computed from the arc weights. A path is a sequence of adjacent nodes and a connectivity function evaluates the strength of connectedness between its terminal nodes. Let $\mathcal{S}$ be a set of relevant maxima in the pdf (e.g., samples $\mathcal{A}$ and $\mathcal{B}$ in Fig. 1a). We wish that each sample in the dataset (e.g., sample $\mathcal{C}$ in Fig. 1a) be reached by a path from $\mathcal{S}$ whose minimum density value along it is maximum. The connectivity function assigns to any path in the graph, the minimum between the density values along it and a handicap value of its starting node. The handicap values work as filtering parameters on the pdf, reducing the numbers of clusters by choosing the relevant maxima. The maximization of the connectivity function for each sample, irrespective to its starting node, partitions the graph into an optimum-path forest, where each root (maximum of the pdf) defines an optimum-path tree (cluster) composed of its most strongly connected samples (Fig. 1c).

Some pdf-based approaches assume either explicitly, or often implicitly, that the domes have known shapes and/or can be fitted to parametric functions (MacQueen, 1967; Dempster et al., 1977; Bezdek, 1981; Jain et al., 1988). Given that the shapes may be far from hyper elliptical, which is the classical assumption, several other methods aim to obtain clusters by avoiding those assumptions (Cheng, 1995; Herbin et al., 1996). Among these approaches, the mean-shift algorithm seems to be the most popular and actively pursued in computer vision (Cheng, 1995; Comaniciu et al., 2000; Comaniciu et al., 2002; DeMenthon, 2002; Comaniciu et al., 2003; Wang et al., 2004; Yang et al., 2005). For each sample, it follows the direction of the pdf's gradient vector toward the steepest maximum around that sample. The pdf is never explicitly computed and each maximum should define an influence zone composed of all samples that achieve it. It is not difficult to see that this approach may present problems if the gradient vector is poorly estimated or has magnitude zero. Besides, if a maximum consists of neighboring points with the same density value, it may break its influence zone into multiple ones. This further increases the number of clusters which is usually higher than the desired one.

Correspondence to: Alexandre Xavier Falcao; e-mail: afalcao@ic.unicamp.br

Grant sponsors: The authors thank the financial support from CNPq and FAPESP, and Awate for the results from their method.

Figure 1. (a) A pdf of two relevant clusters in a 2D feature space (brighter samples show higher density values). The maxima $\mathcal{A}$ and $\mathcal{B}$ compete for sample $\mathcal{C}$ by offering it paths with some strength of connectedness. (b) The influence zones of the pdf's maxima and (c) the influence zones of its relevant maxima.

The proposed method circumvents those problems by first identifying one sample for each relevant maximum of the pdf and then by defining the influence zone of that maximum (robustness). It uses the image foresting transform (IFT), here extended from the image domain to the feature space (Falcão et al., 2004). The IFT has been successfully used to reduce image processing problems into an optimum-path forest problem in a graph derived from the image, by minimizing/maximizing a connectivity function. The image operator is computed from one or more attributes of the forest. The connectivity function we use in the feature space is dual of the one used for the IFT-watershed transform from a gray-scale marker in the image domain (Falcão et al., 2001; Lotufo et al., 2002), which computes a morphological reconstruction (Vincent, 1993) and a watershed transform (Beucher et al., 1979) in a same operation. That is, the obtained clusters are equivalent to the dual-watershed regions of the filtered pdf (the pdf without the irrelevant domes), being a more general solution than the one obtained by the popular mean-shift algorithm (Cheng, 1995).

The literature of graph-based approaches for data clustering is vast (Zahn, 1971; Hubert, 1974; Jain et al., 1988; Wu et al., 1993; Shi et al., 2000; Duda et al., 2000; Luxburg, 2007). Some methods create a neighborhood graph (such as a minimum-spanning tree, the Gabriel graph) from the data samples and then remove inconsistent arcs based on some criterion [e.g., the single-linkage algorithm (Hubert, 1974)]. Other approaches search for a global minimum cut in the graph to create the clusters (Wu et al., 1993; Shi et al., 2000). As far as we know, our approach is the first that models the clustering problem as an optimum-path forest problem. It extends the main ideas under relative-fuzzy connectedness among seeds (Herman et al., 2001; Saha et al., 2001) to other connectivity functions and applications where the seeds (root samples) have to be identified on-the-fly. Another approach based on optimum-path forest has been proposed for supervised classification (Papa et al., 2008). Our method differs from that in the graph model, connectivity function, learning algorithm, and application, which is in our case, unsupervised. Previous versions of our work have also been published (Cappabianco et al., 2008; Rocha et al., 2008). The present article merges and extends them by improving methods and results for large datasets, such as images.

The basic concepts on pdf estimation from arc-weighted graphs are given in Section II. The proposed method is presented in Sections III and IV describes its extension to large data sets. Experimental comparisons with other methods are presented in Section V. Results for interactive image segmentation and for fast, accurate and automatic classification of brain tissues are presented in Section VI, with experiments involving real and synthetic MR images, and another clustering approach as baseline (Awate et al., 2006). Section VII states our conclusions and discuss future work.

II. WEIGHTED GRAPHS AND PDF ESTIMATION

A dataset $\mathcal{N}$ consists of samples from a given application, which may be pixels, objects, images, or any other arbitrary entities. Each sample $s \in \mathcal{N}$ is usually represented by a feature vector $\vec{v}(s)$ and the distance between samples s and t in the corresponding feature space is given by a function $d(s,t)$ (e.g., $d(s,t) = ||\vec{v}(t) - \vec{v}(s)||$). Our problem consists of identifying high concentrations of samples which can characterize relevant clusters for that application. These clusters form domes in the pdf (Fig. 1a), which can be computed by Parzen Window (Duda et al., 2000). However, the shape of the Parzen kernel and its parameters may be chosen by several different ways (Katkovnik et al., 2000; Comaniciu et al., 2001; Comaniciu, 2003; Georgescu et al., 2003).

We say that a sample t is adjacent to a sample s (i.e., $t \in \mathcal{A}(s)$ or $(s,t) \in \mathcal{A}$) when they satisfy some adjacency relation. For example,

$$t \in \mathcal{A}_1(s) \quad \text{if} \quad d(s,t) \leq d_f, \quad \text{or} \tag{1}$$

$$t \in \mathcal{A}_2(s) \quad \text{if} \quad \begin{array}{l} t \text{ is a } k\text{-nearest neighbor of } s \\ \text{in the feature space,} \end{array} \tag{2}$$

where $d_f > 0$ and $k > 0$ are real and integer parameters, respectively, which must be computed by some optimization criterion, such as entropy minimization (Awate et al., 2006). In Section III.B, we present another equivalent option which finds the best value of k in Eq. (2) by minimizing a graph-cut measure. Once $\mathcal{A}$ is defined, we have a graph $(\mathcal{N}, \mathcal{A})$ whose nodes are the data samples in $\mathcal{N}$ and the arcs are defined by the adjacency relation $\mathcal{A}$. The distance values $d(s,t)$ between adjacent samples are arc weights and the pdf

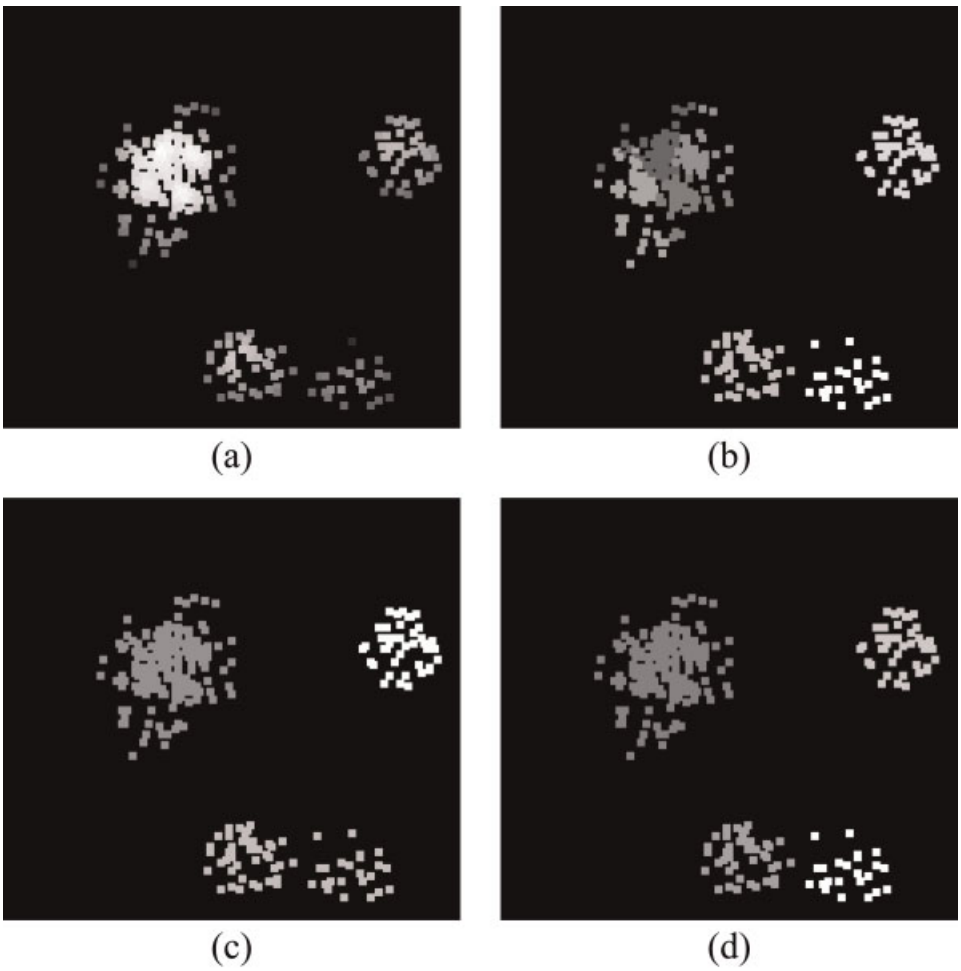


Figure 2. (a–b) A pdf by Eq. (3) and the influence zones of its maxima for $k = 17$ in Eq. (2). (c) The largest top-left cluster can be detected with $k = 40$, but the two clusters at the bottom are merged into one. (d) Our approach can eliminate the irrelevant clusters of (b) by parameter choice in the connectivity function.

values $\rho(s)$ (node weights) can be computed by some kernel. For example,

$$\rho(s) = \frac{1}{\sqrt{2\pi\sigma^2}|\mathcal{A}(s)|} \sum_{t \in \mathcal{A}(s)} \exp\left(\frac{-d^2(s,t)}{2\sigma^2}\right) \tag{3}$$

where σ can be fixed by

$$\sigma = \max_{\forall (s,t) \in \mathcal{A}} \left\{ \frac{d(s,t)}{3} \right\} \tag{4}$$

to guarantee that most adjacent samples are considered for pdf estimation. Note that σ is defined by the maximum arc-weight in $(\mathcal{N}, \mathcal{A})$ divided by 3, which may be different depending on the adjacency relation. Equation (2) defines a *knn*-graph $(\mathcal{N}, \mathcal{A}_2)$ and, although the kernel is Gaussian, only the k-nearest samples of s are used to compute its pdf value. We may also use kernels with different shapes and, although the Gaussian shape favors round clusters, the choice of the connectivity function leads to the detection of clusters with arbitrary shapes (Figs. 1b and 1c).

In data clustering, we must take into account that clusters may present different concentrations and the desired solution depends on a data scale. We have observed that clusters with distinct concentrations are better detected, when we use $\mathcal{A}_2$. Besides, it is easier to find the best integer parameter k than the real parameter d_f for a given application. The scale problem; however, is not possible to solve without hard constraints. Figures 2a and 2b, for example, illustrate a pdf by Eq. (3) and the influence zones of its maxima, for $k = 17$ in Eq. (2). The two less-concentrated clusters at the bottom can be separated, but the largest and dense cluster at the top-left is divided into several influence zones. The pdf estimation is improved for the top-left cluster, when $k = 40$, but the two clusters at the bottom are merged into a single one (Fig. 2c). To obtain four clusters, as shown in Figure 2d, we change a parameter in the connectivity function such that the irrelevant clusters of Figure 2b are eliminated.

III. DATA CLUSTERING BY OPTIMUM-PATH FOREST

In Section III.A, we show how to detect "relevant maxima" in the pdf and to compute the influence zones of those maxima as an optimum-path forest in $(\mathcal{N}, \mathcal{A})$. A connectivity function is defined such that irrelevant maxima are naturally eliminated during the process and a single root sample is detected per maximum. These roots are

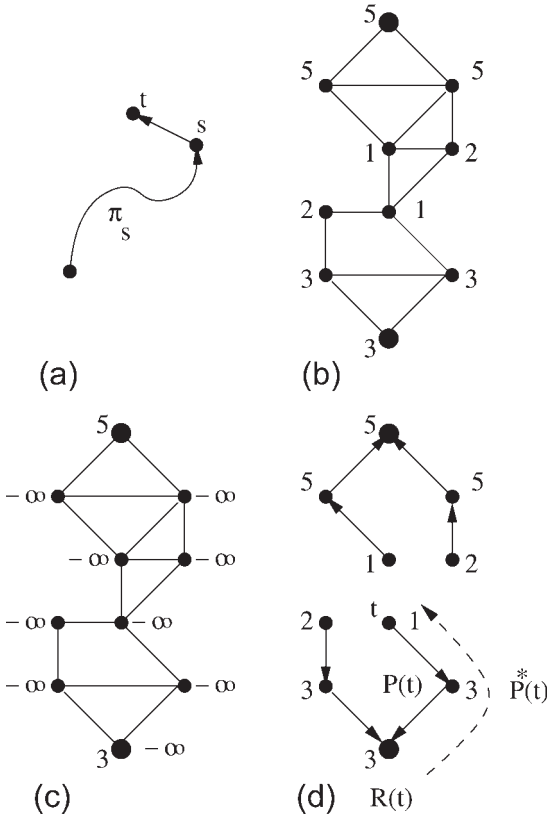


Figure 3. (a) Path π_s with possible extension $\langle s,t\rangle$. (b) A graph whose node weights are their pdf values $\rho(t)$. There are two maxima with values 3 and 5, respectively. The bigger dots indicate the root set $\mathcal{R}$. (c) Trivial path values $f_1(\langle t\rangle)$ for each sample t. (d) Optimum-path forest P for f_1 and the final path values $V(t)$. The optimum path $P^*(t)$ (dashed line) can be obtained by following the predecessors $P(t)$ up to the root $R(t)$ for every sample t.

labeled with distinct integer numbers and their labels are propagated to each of their most strongly connected samples, forming an optimum-path tree rooted at each maximum.

For adjacency relations given by Eq. (2), different choices of k lead to distinct optimum-path forests, whose labeled trees represent distinct cuts in the graph $(\mathcal{N},\mathcal{A})$. The best value of k is chosen as the one whose optimum-path forest minimizes a graph-cut measure (Section III.B).

A. Influence Zones from Relevant Maxima. A path π_t in $(\mathcal{N},\mathcal{A})$ is a sequence of adjacent nodes with terminus t. A path $\pi_t = \langle t\rangle$ is said trivial and $\pi_t = \pi_s \cdot \langle s,t\rangle$ is the concatenation of a path π_s by an arc $(s,t) \in \mathcal{A}$ (Fig. 3a). A sample t is connected to a sample s when there is a path from s to t.

Symmetric adjacency relations [e.g., $\mathcal{A}_1$ in Eq. (1)] result into symmetric connectivity relations, but $\mathcal{A}_2$ in Eq. (2) is an asymmetric adjacency. Given that a maximum of the pdf may be a subset of adjacent samples with a same density value, we need to guarantee connectivity between any pair of samples in that maximum. Thus, any sample of the maximum can be a representative and reach the other samples in that maximum and in their influence zones by an optimum path (Figs. 1 and 2). This requires extending the adjacency relation $\mathcal{A}_2$ to be symmetric in the plateaus of ρ in order to compute clusters.

$$\begin{array}{l}\text{if} \quad t \in \mathcal{A}_2(s),\\ s \notin \mathcal{A}_2(t) \quad \text{and}\\ \rho(s) = \rho(t), \quad \text{then}\\ \mathcal{A}_3(t) \leftarrow \mathcal{A}_2(t) \cup \{s\}.\end{array} \tag{5}$$

A connectivity function $f(\pi_t)$ assigns a value to any path π_t, representing a "strength of connectedness" of t with respect to its starting node $R(t)$ (root node). A path π_t is optimum when $f(\pi_t) \geq f(\tau_t)$ for any other path τ_t, irrespective to its root. We wish to choose t such that its maximization for every node t will constraint the roots of the optimum paths in the maxima of the pdf. That is, we wish to assign to every sample $t \in \mathcal{N}$ an optimum path $P^*(t)$ whose strength of connectedness $V(t)$ is the highest with respect to one among the pdf's maxima.

$$V(t) = \max_{\forall \pi_t \in (\mathcal{N},\mathcal{A})} \{f(\pi_t)\}. \tag{6}$$

The image foresting transform (IFT) (Falcão et al., 2004) solves the problem by starting from trivial paths for all samples. First, the maxima of $f(\langle t\rangle)$ are detected and then optimum paths are propagated from those maxima to their adjacent nodes, and from them to their adjacents, by following a nonincreasing order of path values. That is,

$$\text{if } f(\pi_s \cdot \langle s,t\rangle) > f(\pi_t) \quad \text{then} \quad \pi_t \leftarrow \pi_s \cdot \langle s,t\rangle. \tag{7}$$

The only requirement is that f must be smooth. That is, for any sample $t \in \mathcal{N}$, there is an optimum path $P^*(t)$ which either is trivial, or has the form $P^*(s) \cdot \langle s,t\rangle$ where

a. $f(P^*(s)) \geq f(P^*(t))$,
b. $P^*(s)$ is optimum,
c. for any optimum path $P^*(s)$, $f(P^*(s) \cdot \langle s,t\rangle) = f(P^*(t)) = V(t)$.

If we had one sample per maximum, forming a set $\mathcal{R}$ (bigger dots in Fig. 3b), then the maximization of function f_1 would solve the problem.

$$\begin{array}{rl} f_1(\langle t\rangle) & = \begin{cases} \rho(t) \text{ if } t \in \mathcal{R} \\ -\infty \quad \text{otherwise} \end{cases} \\ f_1(\pi_s \cdot \langle s,t\rangle) & = \min\{f_1(\pi_s), \rho(t)\}. \end{array} \tag{8}$$

Function f_1 has an initialization term and a path propagation term, which assigns to any path π_t the lowest density value along it. Every sample $t \in \mathcal{R}$ defines an optimum trivial path $\langle t\rangle$ because it is not possible to reach t from another maximum of the pdf without passing through samples with density values lower than $\rho(t)$ (Fig. 3b). The other samples start with trivial paths of value $-\infty$ (Fig. 3c), then any path from $\mathcal{R}$ has higher value than that. Considering all possible paths from $\mathcal{R}$ to every sample, the optimum path $P^*(t)$ will be the one which has the lowest density value along it is maximum.

The optimum paths are stored in a predecessor map P, forming an optimum-path forest with roots in $\mathcal{R}$—i.e., a function with no cycles that assigns to each sample $t \notin \mathcal{R}$ its predecessor $P(t)$ in the optimum path from $\mathcal{R}$ or a marker *nil* when $t \in \mathcal{R}$. The optimum path $P^*(t)$ with terminus t can be easily obtained by following $P(t)$ backwards up to its root $R(t)$ in $\mathcal{R}$ (Fig. 3d).

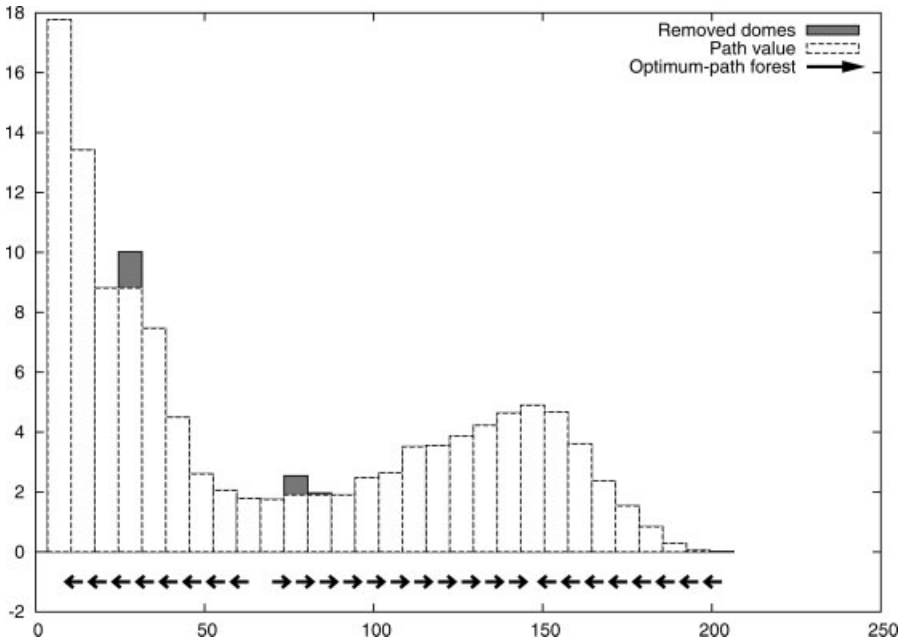


Figure 4. The boxes show an 1D pdf ρ with four maxima. The map V (white) indicates the removal of two irrelevant domes (gray) when $h(t) = \rho(t) - 2$. The 1D optimum-path forest P (vectors) shows the influence zones of the two remaining maxima.

Given that we do not have the maxima of the pdf, the connectivity function must be chosen such that its handicap values define the relevant maxima of the pdf. For $f_1(\langle t \rangle) = h(t) < \rho(t)$, for all $t \in \mathcal{N}$, some maxima of the pdf will be preserved and the others will be reached by paths from the root maxima, whose values are higher than their handicap values. For example, if

$$h(t) = \rho(t) - \delta, \quad \delta = \min_{(s,t)\in\mathcal{A}|\rho(t)\neq\rho(s)} |\rho(t) - \rho(s)|, \tag{9}$$

then all maxima of ρ are preserved. For higher values of δ, the domes of the pdf with height less than δ will not define influence zones.

Figure 4 shows an example where ρ is an 1D pdf. If $h(t) = \rho(t) - 2$, then the number of maxima is reduced from four to two. The map V and optimum-path forest P (vectors of the predecessor map) are shown in Figure 4, indicating the influences zones of the two remaining maxima. The number of clusters can also be reduced by removing domes with area or volume below a threshold. This is done when h_I results from an area or volume opening on the pdf (Salembier et al., 1998). We usually scale ρ within an interval $[1, K]$ (e.g., $K = 100$ or $K = 1000$) of real numbers, such that it is easier to set δ and to guarantee that $h(t) < \rho(t)$ by subtracting 1 from $h(t)$.

We also want to avoid the division of the influence zone of a maximum into multiple influence zones, each one rooted at a sample of that maximum. Given that the IFT algorithm first identifies the maxima of the pdf, before propagating their influence zones, we can change it to detect a first sample t per maximum, defining the set $\mathcal{R}$ on-the-fly. We then change $h(t)$ by $\rho(t)$ and this sample will conquer the remaining samples of the same maximum. Thus the final connectivity function f_2 becomes

$$f_2(\langle t \rangle) = \begin{cases} \rho(t) \text{ if } t \in \mathcal{R}. \\ h(t) \text{ otherwise.} \end{cases} \tag{10}$$
$$f_2(\pi_s \cdot \langle s, t \rangle) = \min\{f(\pi_s), \rho(t)\}.$$

Algorithm 1 presents the IFT modified for a graph $(\mathcal{N}, \mathcal{A})$ and connectivity function f_2. It identifies a single root in each relevant maximum, labels it with a consecutive integer number l, and computes optimum paths for f_2 from the roots, by following a nonincreasing order of path values. The optimum-path values are stored in V, while the root labels $L(t)$ and predecessors $P(t)$ are propagated to each sample t. The roots $R(t)$ do not need to be propagated.

Algorithm 1: Clustering by optimum-path Forest

INPUT:	Graph $(\mathcal{N}, \mathcal{A})$ and functions h and ρ, $h(t) < \rho(t)$ for all $t \in \mathcal{N}$.
OUTPUT:	Label map L.
AUXILIARY:	Path-value map V, predecessor map P, priority queue Q, variables tmp and $l \leftarrow 1$.

1. *For each* $t \in \mathcal{N}$, *set* $P(t) \leftarrow nil$, $V(t) \leftarrow h(t)$, *insert* t *in* Q.
2. *While* Q *is not empty, do*
3. *Remove from* Q *a sample* s *such that* $V(s)$ *is maximum.*
4. *If* $P(s) = nil$ *then set* $L(s) \leftarrow l$, $l \leftarrow l + 1$, $V(s) \leftarrow \rho(s)$.
5. *For each* $t \in \mathcal{A}(s)$ *such that* $V(t) < V(s)$, *do*
6. *Compute* $tmp \leftarrow \min\{V(s), \rho(t)\}$.
7. *If* $tmp > V(t)$, *then*
8. *Set* $L(t) \leftarrow L(s)$, $P(t) \leftarrow s$, *and* $V(t) \leftarrow tmp$.
9. *Update position of* t *in* Q.

Line 1 initializes maps and inserts all samples in Q. At each iteration of the main loop (Lines 2–9), an optimum path $P^*(s)$ with

value $V(s)$ is obtained in P when we remove its last sample s from Q (Line 3). Ties are broken in Q using first-in-first-out (FIFO) policy. That is, when two optimum paths reach an ambiguous sample s with the same maximum value, s is assigned to the first path that reached it. The test $P(s) = nil$ in Line 4 identifies $P^*(s)$ as a trivial path $\langle s \rangle$. Given that the optimum paths are found in a nonincreasing order of values, trivial paths indicate samples in the maxima. By changing $V(s)$ to $\rho(s)$, as defined by Eq. (10) and indicated in Line 4, we are forcing a first sample in each maximum to conquer the rest of the samples in that maximum. Therefore, $s \in \mathcal{R}$ becomes root of the forest in Line 4 and a distinct label l is assigned to it. Lines 5–9 evaluate if the path that reaches an adjacent sample t through s is better than the current path with terminus t and update Q, V, L, and P accordingly. Note that, the condition in Line 5 avoids evaluating adjacent nodes already removed from Q.

The computation of P was shown to facilitate the description of the algorithm. However, it is not needed for data clustering. One may initialize $L(t) \leftarrow nil$ in Line 1, remove $P(t) \leftarrow s$ in Line 8, and replace $P(s) = nil$ by $L(s) = nil$ in Line 4.

Algorithm 1 runs in $\Theta(|\mathcal{A}| + |\mathcal{N}| \log|\mathcal{N}|)$ if Q is a balanced heap data structure (Falcão et al., 2004). This running time may be reduced to $\Theta(|\mathcal{A}| + |\mathcal{N}|K)$ if we convert ρ and h to integer values in the range of $[0, K]$ and implement Q with bucket sorting (Falcão et al., 2000). We are using the heap implementation with real path values in this work.

B. Estimation of the Best *knn*-Graph. The results of Algorithm 1 will also depend on the choice of $\mathcal{A}$ (e.g., the value of k in the case of a *knn*-graph). Considering the influence zones a cut in the graph $(\mathcal{N}, \mathcal{A}_3)$ [Eq. (5)], we wish to determine the value of k which optimizes some graph-cut measure.

Clustering validity measures could be used but they usually assume compact and well separated clusters (Theodoridis et al., 1999; Halkidi et al., 2001). The measure should be independent of the shape of the clusters. Thus we use the graph-cut measure for multiple clusters as suggested in (Shi et al., 2000).

Let $1/d(s,t)$ be the arc weights in a *knn*-graph $(\mathcal{N}, \mathcal{A}_3)$. Algorithm 1 can provide in La graph cut for each value of $k \in [1, (|\mathcal{N}| - 1)]$. This cut is measured by $C(k)$.

$$C(k) = \sum_{i=1}^{c} \frac{W_i'}{W_i + W_i'}, \tag{11}$$

$$W_i = \sum_{(s,t) \in \mathcal{A}_3 | L(s) = L(t) = i} \frac{1}{d(s,t)}, \tag{12}$$

$$W_i' = \sum_{(s,t) \in \mathcal{A}_3 | L(s) = i, L(t) \neq i} \frac{1}{d(s,t)}, \tag{13}$$

The best cut is defined by the minimum value of $C(k)$, where W_i' considers all arc weights between cluster i and other clusters, and W_i considers all arc weights within cluster $i = 1,2,\ldots,c$. The desired minimum in $C(k)$ is usually within $k \in [1, k_{\max}]$, for $k_{\max} \ll |\mathcal{N}|$, which represents the most reasonable solution for a given scale. Therefore, we usually constrain the search within that interval.

IV. EXTENSIONS TO LARGE DATASETS

The choice of the adjacency parameter, d_f or k, by optimization requires the execution of Algorithm 1 several times (e.g., $k_{\max}$). Depending on the number of nodes and executions, the clustering process may take minutes running on modern PCs. Given that we have to compute and store the arcs, the problem becomes unsurmountable for 2D and 3D images with thousands of pixels and millions of voxels. Therefore, we present two possible extensions for large datasets.

A. Clustering with Size Constraint. Algorithm 1 is computed within a small subset $\mathcal{N}' \subset \mathcal{N}$ and then the classification of the remaining samples in $\mathcal{N} \setminus \mathcal{N}'$ is done one by one, as though the sample were part of the forest. In general, $\mathcal{N}'$ may be chosen by some random procedure. One can repeat the process several times and take a final decision by majority vote (Section VI.B). We then compute the best *knn*-graph $(\mathcal{N}', \mathcal{A}_3)$ as described before.

Let V and L be the optimum maps obtained from $(\mathcal{N}', \mathcal{A}_3)$ by Algorithm 1. A sample $t \in \mathcal{N} \setminus \mathcal{N}'$ is classified in one of the clusters by identifying which root would offer it an optimum path. By considering the adjacent samples $s \in \mathcal{A}_3(t) \subset \mathcal{N}'$, we compute ρ by Eq. (3), evaluate the paths $\pi_s \cdot \langle s,t \rangle$, and select the one that satisfies

$$V(t) = \max_{\forall (s,t) \in \mathcal{A}_3} \{\min\{V(s), \rho(t)\}\}. \tag{14}$$

Let the node $s^* \in \mathcal{N}'$ be the one that satisfies Eq. (15). The classification simply assigns as the cluster of t.

B. Clustering with Spatial Constraint. If we considerably reduce the number of arcs by adding some spatial constraint to the adjacency computation, then the entire image domain $\mathcal{N}$ can be used to form the nodes of the graph. For example, Algorithm 1 can be directly executed in $(\mathcal{N}, \mathcal{A}_4)$, where

$$t \in \mathcal{A}_4(s) \text{ if } d(s,t) \leq d_f \quad \text{and} \quad ||t - s|| \leq d_i. \tag{15}$$

The parameter d_f can be computed using the first approach in a small subset $\mathcal{N}' \subset \mathcal{N}$. This subset may consist, for example, of every 16×16 pixels obtained by uniform sampling in the original image (Section VI.A). The best *knn*-graph $(\mathcal{N}', \mathcal{A}_3)$ is computed and the maximum arc weight used to set σ by Eq. (4) and d_f in Eq. (16). Figure 5 illustrates four images and their respective pdfs, when $d_i = 5$ in Eq. (16) and the density values in Eq. (3) are scaled from $\lceil 1 - 100 \rceil$.

Smaller values of d_i increase efficiency, but they also increase the number of clusters. The choice of h in Eq. (10) then becomes paramount to reduce the number of irrelevant clusters. The next section shows results of both extensions to large datasets.

V. EXPERIMENTAL COMPARISONS WITH OTHER METHODS

OPF finds natural groups in a dataset, but does not guarantee a desired number of clusters. Other clustering methods can output a desired number of groups, but which groups correspond to each class cannot be solved based only on similarity functions and

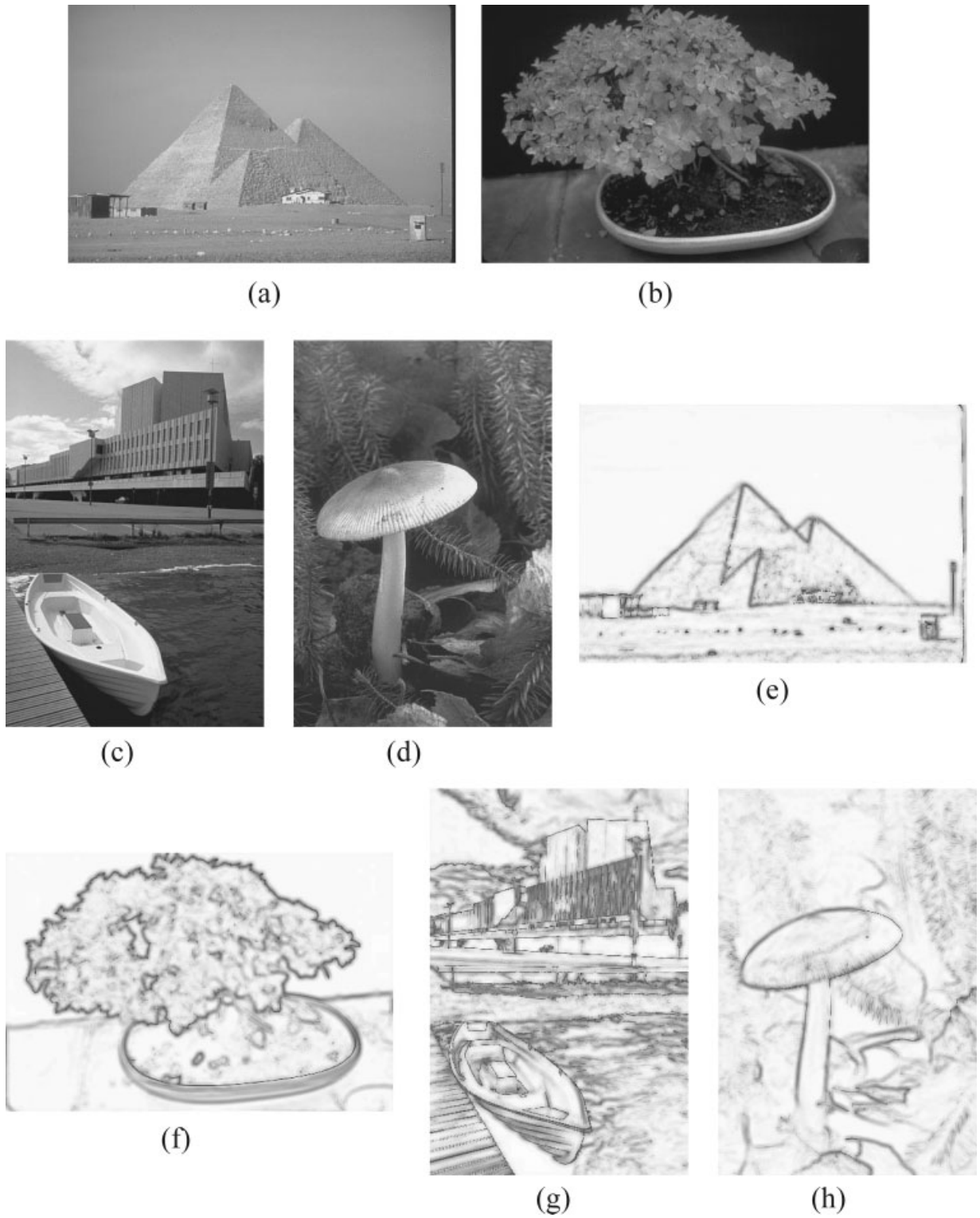


Figure 5. (a–d) Natural images and (e–h) their pdfs, computed with $d_i = 5$ in Eq. (16) and density values scaled from $[1 - 100]$ in Eq. (3).

optimality criteria. Even the number of groups per class is unknown in several applications. We illustrate the problem by evaluating OPF and those clustering methods in various labeled datasets.

Consider a labeled dataset $\mathcal{N}$, where we know the correct class of each sample. A good clustering approach should find natural groups without mixing samples from distinct classes. By forcing the number of groups to be the same of the number of classes, for

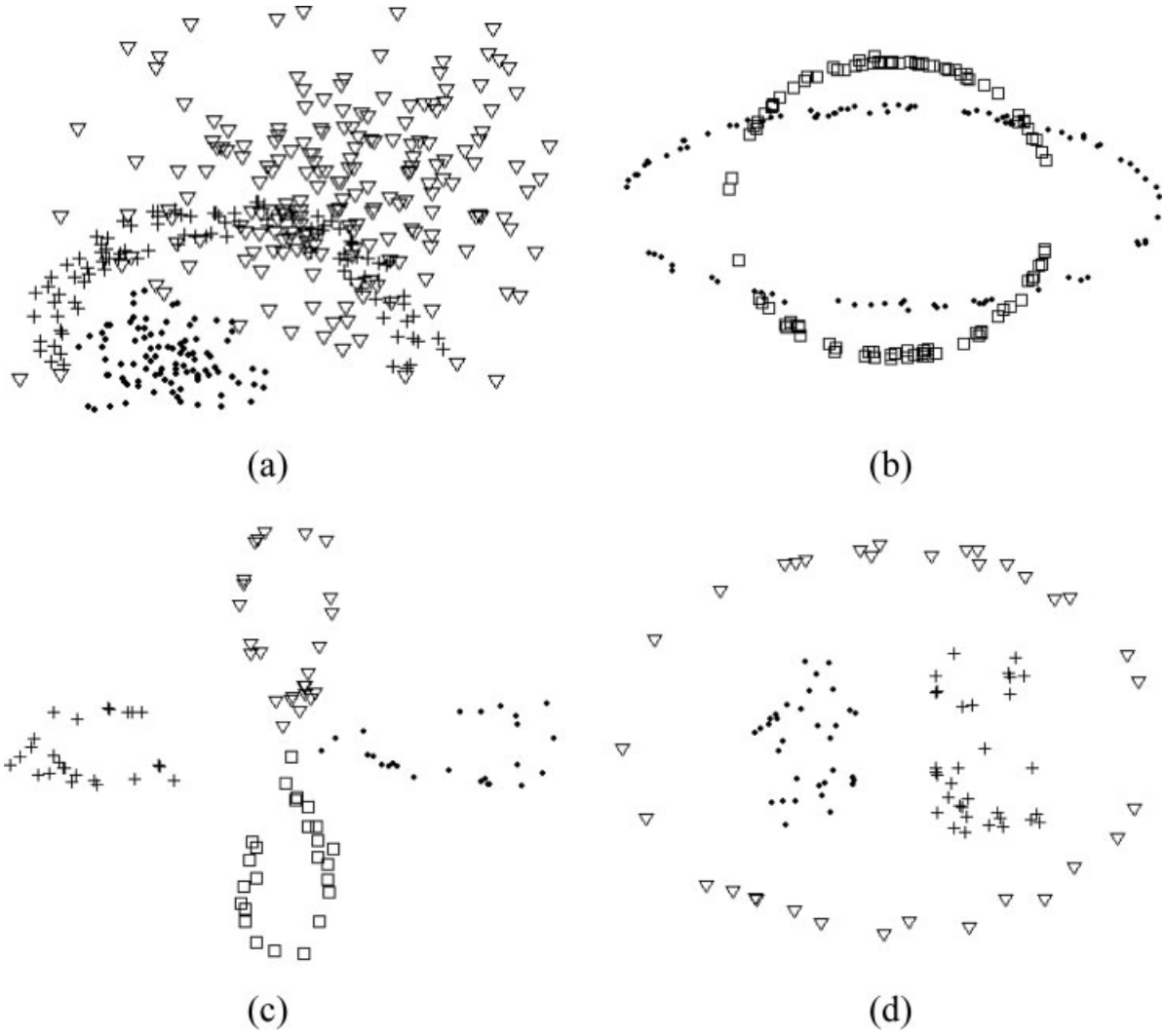


Figure 6. Datasets of 2D points: (a) cone-torus, (b) saturn, (c) petals, and (d) boat.

example, one may observe more mixture of distinct classes in some datasets. For the experiments of this section, we have selected synthetic datasets (Figs. 6 and 7), in which we expect one group per class, and three real datasets, MPEG-7 (MPEG-7, 2002), Wisconsin Breast Cancer (WBC), and Letter Recognition (LR) (Newman, 2007), in which we do not know the number of clusters per class. MPEG-7 consists of shapes (Fig. 8) and so we cluster it by using three shape descriptors: Beam Angle Statistics (BAS) (Arica and Vural, 2003), Fourier Coefficients (FC) (Persoon et al., 1977), and Multi-Scale Fractal dimensions (MSF) (Torres et al., 2004). These descriptors provide different degrees of class separation in the feature space. We expect better clustering performance as more separated are the classes in the feature space.

In OPF, all samples in a given optimum-path tree are assumed to have the same label of their root. To measure the mixture of classes in the clusters, we can verify the roots of the forest, assign the correct class to the label of each root, and propagate this label to the remaining samples of its optimum-path tree. The purity of the clustering is then measured as the percentage of correct classifications by this procedure. For other methods, which are not based on the optimum-path forest, we assign to each cluster the class of the majority of its samples and use the same measure of purity.

We have chosen the library CLUTO* for the experiments, because it provides six clustering methods, four similarity functions and 12 optimality criteria. We have evaluated all possible combinations for each dataset and Table I shows only the combinations with the highest purity values. We assigned a code to each combination and Table II shows their purity values for each dataset. We are using the same nomenclature of CLUTO for its parameters. The best methods were: graph—it computes a c-way clustering of a nearest-neighbor graph by the min-cut algorithm, bagglo—it is an agglomerative approach into c clusters, and rbr—it is a partitional approach into c clusters with global optimization. For at least one case, each similarity function presented the best result: cos, cosine function; corr, correlation coefficients; dist, inverse of the Euclidean distance; and jacc, extended Jaccard coefficient. The best optimality criteria were: i2—it maximizes the total similarity within each group; clink—the traditional

*URL: http://www.caip.rutgers.edu/riul/research/code/EDISON

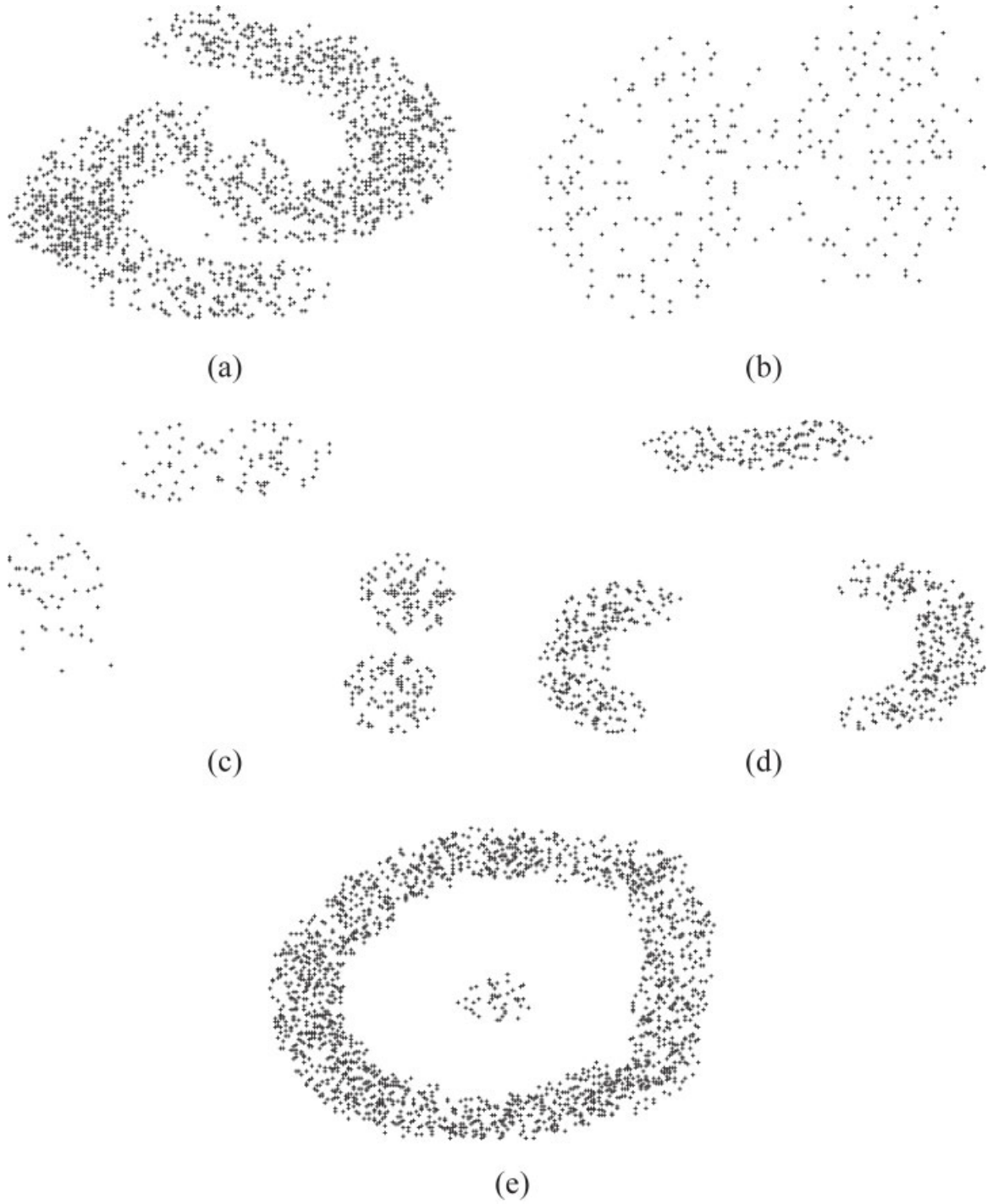


Figure 7. Datasets of 2D points: (a) data1, (b) data2, (c) data3, (d) data4, and (d) data5.

complete-link criterion; and g1p—it minimizes the similarity between distinct groups.

The number c of clusters in CLUTO is set to be the same number of classes for each dataset. Good purity values above 70.00% can be observed in Table II for the cases where each class can be represented by one group (data1-data5 and petals, cone-torus, and boat), except for saturn. Such one-to-one correspondence seems to be not valid for LR and MPEG-7, but it holds for WBC. The purity values indicate that the shape descriptor BAS can better separate the classes in the feature space than MSF and FC. Table II also presents the purity values obtained by OPF for each dataset.

The optimality criterion is the optimum-path forest with normalized minimum cut [Eq. (11)] and the similarity function is equivalent to dist in Table I. One may improve the results of OPF with better similarity function and optimality criterion

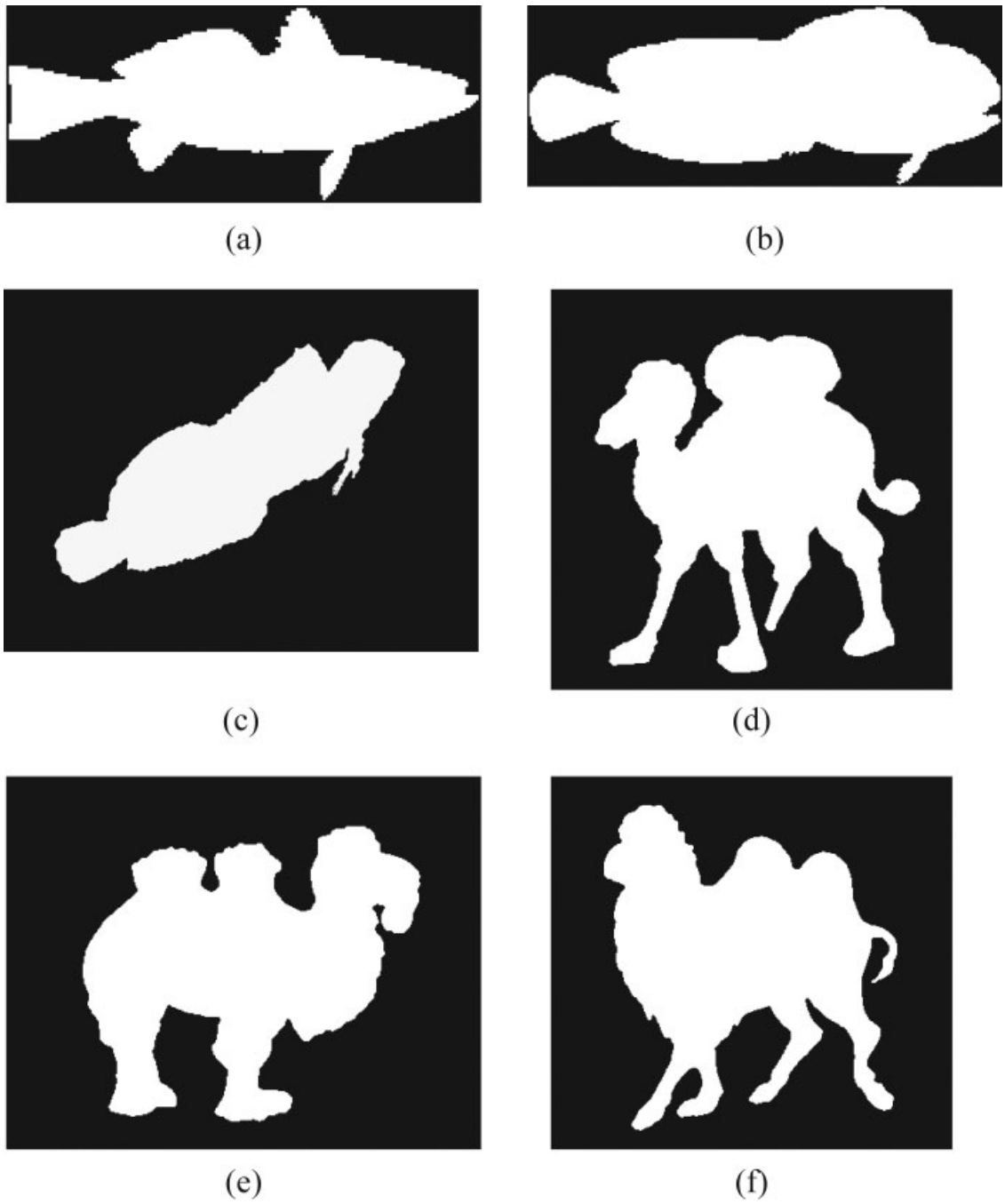


Figure 8. Examples of shapes in MPEG-7 from the classes (a–c) fish and (d–f) camel.

for the best k. The parameters $k_{\max}$ and $h(t)$ were found experimentally and using the volume opening on the pdf (Salembier et al., 2000). The general idea was to minimize the number of clusters for purity values above 70.00%. For data1-data5, petals, boat and cone-torus, OPF obtained good purity values, sometimes higher than CLUTO, with the desired number of classes, except for saturn. For WBC, OPF required four clusters to achieve result similar to CLUTO. In MPEG-7, higher is the separability of the classes in the feature space, less is the number of clusters obtained with each shape descriptor, but this number is much higher than the number of classes. Any attempt to further reduce the number of clusters will drastically reduce the purity values.

We may conclude that it is possible to obtain good and sometimes better results with OPF (e.g., see data3-data5). For a given application, we need to investigate the best distance (similarity) function, optimality criterion for the best k, $k_{\max}$, and $h(t)$.

VI. RESULTS IN IMAGE SEGMENTATION

A multidimensional and multiparametric image $\hat{I}$ is a pair $(\mathcal{N}, \vec{I})$ where $(\mathcal{N} \subset Z^n)$ is the image domain in n dimensions and $\vec{I}(s) = \{I_1(s), I_2(s), \ldots, I_m(s)\}$ is a vectorial function, which assigns

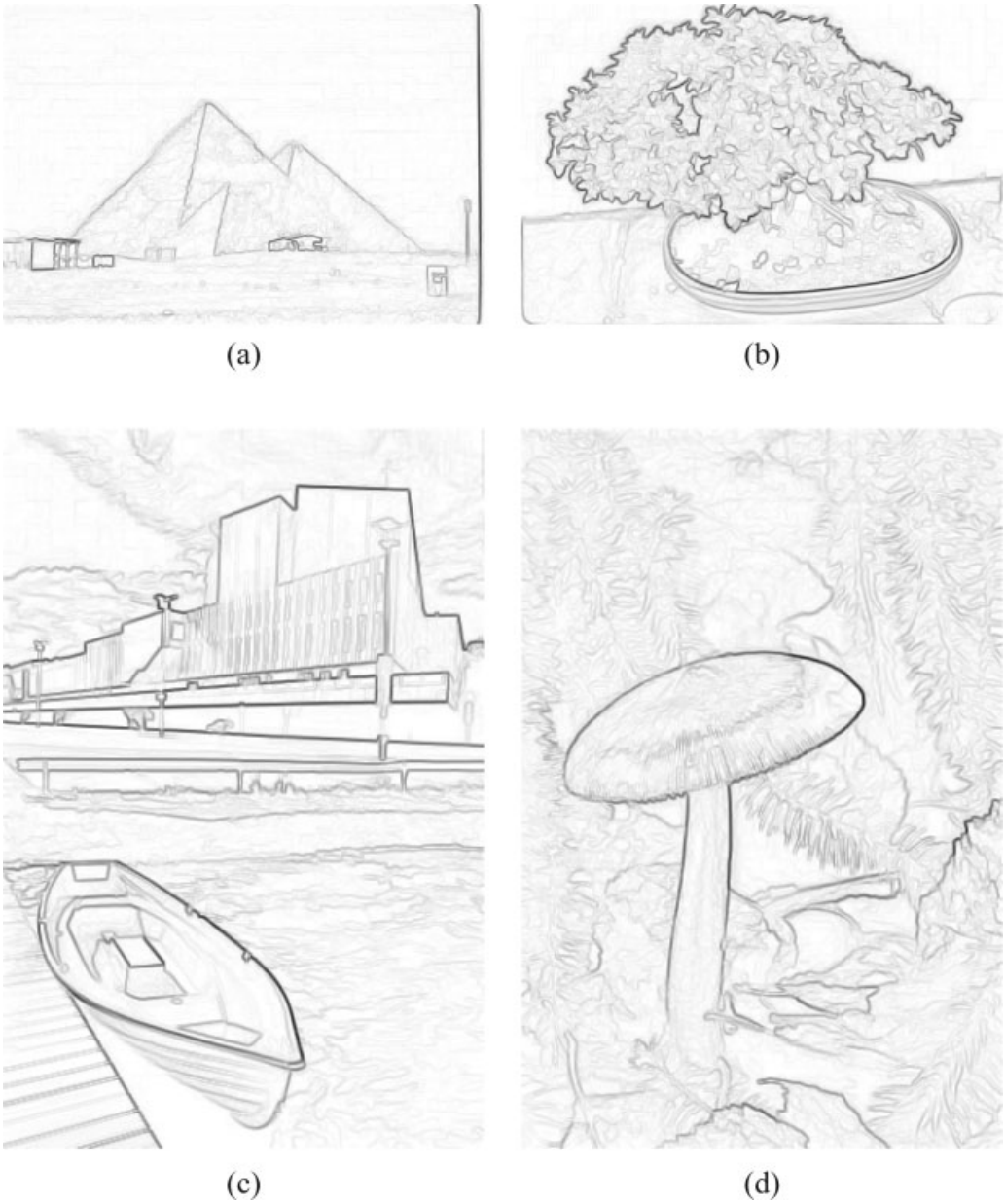


Figure 9. (a–d) Gradient images computed from the images in Figures 5a–5d using Eq. (19). Lower brightness values indicate higher gradient values.

m image properties (parameters) to each pixel $t \in \mathcal{N}$. For example, $\{I_1(t), I_2(t), I_3(t)\}$ may be the red, green, and blue values of t in a color image $\hat{I}$. We present segmentation results for 2D (natural scenes) and 3D (MR-images) datasets in this section.

A. Natural Scenes. Objects in natural scenes usually consist of a single connected component each, but parts of the background may present similar image features. The clustering with spatial constraint seems to be more suitable in this case, because the clusters can be broken into disconnected regions such that similar parts of object and background are more likely to fall in different regions (Fig. 10).

The graph $(\mathcal{N}, \mathcal{A}_4)$ can be created as described in Section IV.B, but the image features play an important role in the segmentation results. Instead of using $\vec{I}(s)$ as the image features of each pixel $s \in \mathcal{N}$, we describe in Section VI.A.1 other options based on image smoothing in several scales. Note that the choice of the best feature set for a given segmentation task is subject for a future work, given the variability of the natural scenes.

Algorithm 1 computes a filtered pdf in V (inferior reconstruction of ρ from h) and the dual-watershed regions of it in L (the influence zones of the maxima of V). This represents an extension of the IFT-watershed transform from gray-scale marker (Lotufo et al., 2002) from the image domain to the feature space. Section 6.1.2 then

Table I. Code for the best combinations of method, similarity function, and optimality criterion in CLUTO.

Code	Method	Similarity	Optimality Criterion
1	Graph	dist	i2
2	Graph	jacc	i2
3	Bagglo	cos	clink
4	rbr	corr	g1p

The purity values for these combinations in the respective datasets are listed in Table II. We are using the same nomenclature of CLUTO for its parameters, as described in the text.

presents a comparative analysis of the proposed approach with respect to (Lotufo et al., 2002) and the mean-shift algorithm (Cheng, 1995).

Finally, the clustering results are not usually enough to solve image segmentation. Some global information is needed to indicate which regions compose the object (Fig. 10). We then take the user's help for this task. Section VI.A.3 presents an interactive approach, where the user involvement is reduced to draw markers that either merge object regions or split a selected region, when clustering fails in separating object and background (Figs. 11a–11h). The method used for region splitting is the IFT-watershed transform from labeled markers (Lotufo et al., 2000).

A.1. Multiscale Image Features. Multiscale image smoothing can be computed by linear convolutions with Gaussians (Lindeberg, 1994) and/or by various types of levelings (Vincent, 1993; Salembier et al., 1995; Salembier et al., 1998; Meyer, 2004). In this article, we are using sequences of opening by reconstruction and closing by reconstruction, computed over each image band I_i, $i = 1,2,...m$, for disks of radii $j = 1,2,...,S$ (e.g., $S = 4$). Gaussian filters can provide smoother contours than morphological reconstructions, but the latter better preserves the natural indentations and protrusions of the shapes.

Let $\vec{v}_i(s) = (v_{i.1}(s), v_{i.2}(s), \ldots, v_{i.S}(s))$ be the pixel intensities $v_{i,j}(s), j = 1, 2, \ldots, S$, of the multiscale smoothing on each band I_i, $i = 1,2,3$ of an RGB image. The feature vector $\vec{v}(s)$ assigned to each pixel $s \in \mathcal{N}$ is $(v_{1.1}(s), \ldots, v_{1.S}(s), v_{2.1}(s), \ldots, v_{2.S}(s), v_{3.1}(s), \ldots, v_{3,s}(s))$, and the distance $d(s,t)$ between these vectors is Euclidean.

The multiscale image features are also used for gradient computation in both IFT-watershed transforms, from gray-scale marker (Lotufo et al., 2002) and from labeled marker (Lotufo et al., 2000). A gradient image $(\mathcal{N}, G)$ is computed using adjacency relation $\mathcal{A}_5$ (8-neighborhood), as follows.

$$t \in \mathcal{A}_5(s) \quad \text{if} \quad ||t - s|| \leq \sqrt{2}, \tag{16}$$

$$\vec{G}_i(s) = \sum_{j=1}^{S} \sum_{\forall t \in \mathcal{A}_5(s)} |v_{i,j}(t) - v_{i,j}(s)]\vec{st}, \tag{17}$$

$$G(s) = \max_{i=1,2,3} \left|\left|\vec{G}_i(s)\right|\right| \tag{18}$$

where $\vec{st}$ is the unit vector connecting s to t in the image domain (Fig. 9).

A.2. Comparative Analysis. When comparing segmentation methods, we must be careful to avoid experimental comparisons between different implementations. The mean-shift code† requires adjustments of some parameters, uses different image features, and merges the labeled clusters based on a distance criterion between maxima (Comaniciu et al., 2000). The same criterion could be applied in our approach, with no guarantee that object and background will be separated. For this reason, we believe that the clustering should minimize the number of object's regions as much as possible and let the user to complete the process (Section VI.A.3).

Figures 10a–10d present the labeled clusters of Algorithm 1 for f_2 with $h(t) = \rho(t) - 1$ and $\rho(t) \in [1,100]$ (Figs. 5e–5h). These results are similar to those of the mean-shift approach (Cheng, 1995), when the mean-shift merges the influence zones of samples in a same maximum and solves gradient problems on plateaus (Section I. These objects are divided into several regions, but their boundaries are preserved. To reduce the number of regions for interactive segmentation, we run Algorithm 1 with h computed by volume opening on ρ (Salembier et al., 2000) (Figs. 10e–10h). The IFT-watershed transform from gray-scale marker uses the volume closing to create a marker $h(t) > G(t)$ and runs the IFT on an image graph $(\mathcal{N}, \mathcal{A}_5)$ to minimize a connectivity function f_4 [see the duality with Eq. (10)].

$$f_4(\langle t \rangle) = \begin{cases} G(t) & \text{if } t \in R \\ h(t) & \text{otherwise} \end{cases} \tag{19}$$
$$f_4(\pi_s \cdot \langle s, t \rangle) = \max\{f_4(\pi_s), G(t)\}$$

where $\mathcal{R}$ is the set of the relevant minima in G, which become the only minima of V (superior reconstruction of G from the marker h). Their influence zones appear in L. The constraint d_f in Eq. (16) allows a higher radius $d_i = 5$ than the one used in Eq. (17). This together with the use of ρ rather than G usually reduces the number of regions with respect to the number obtained by the IFT-watershed from gray-scale marker (Figs. 10i–10l).

A.3. Interactive Segmentation. The regions in Figures 10e–10h are obtained by separating the clusters into 4-connected image components. The partition helps the user to identify which regions compose the object and select markers to merge them (Figs. 11a–11d). It also shows when a region includes object and background (e.g., Fig. 11d), but their pixels can be easily separated with an IFT-watershed transform from labeled markers (Lotufo et al., 2000) constrained to that region. The markers are labeled as internal and external seed pixels, forming a set $\mathcal{R}$. The IFT algorithm runs on an

Table II. The columns show the datasets and their number of classes (nclasses), the purity values obtained by CLUTO with the parameter combination (code) indicated in Table I, and the purity values of OPF to obtain a minimum number of groups (ngroups) with purity above 70.00%.

Dataset (nclasses)	CLUTO (code)	OPF (ngroups)
Data1 (2)	99.37 (1)	99.09 (2)
Data2 (2)	98.59 (1)	97.53 (2)
Data3 (5)	88.24 (1)	99.71 (5)
Data4 (3)	74.64 (3)	100.00 (3)
Data5 (2)	97.73 (2)	100.00 (2)
LR (26)	39.43 (1)	70.83 (256)
MPEG7-BAS (70)	57.36 (4)	82.86 (258)
MPEG7-FC (70)	33.36 (1)	76.86 (671)
MPEG7-MSF (70)	43.29 (1)	77.00 (587)
Petals (4)	100.00 (2)	98.00 (4)
Saturn (2)	58.00 (2)	82.50 (13)
Boat (3)	79.00 (1)	74.00 (3)
Cone-torus (3)	72.00 (1)	72.00 (3)
WBC (2)	95.70 (1)	94.84 (4)

†URL: http://www.bic.mni.mcgill.ca/brainweb

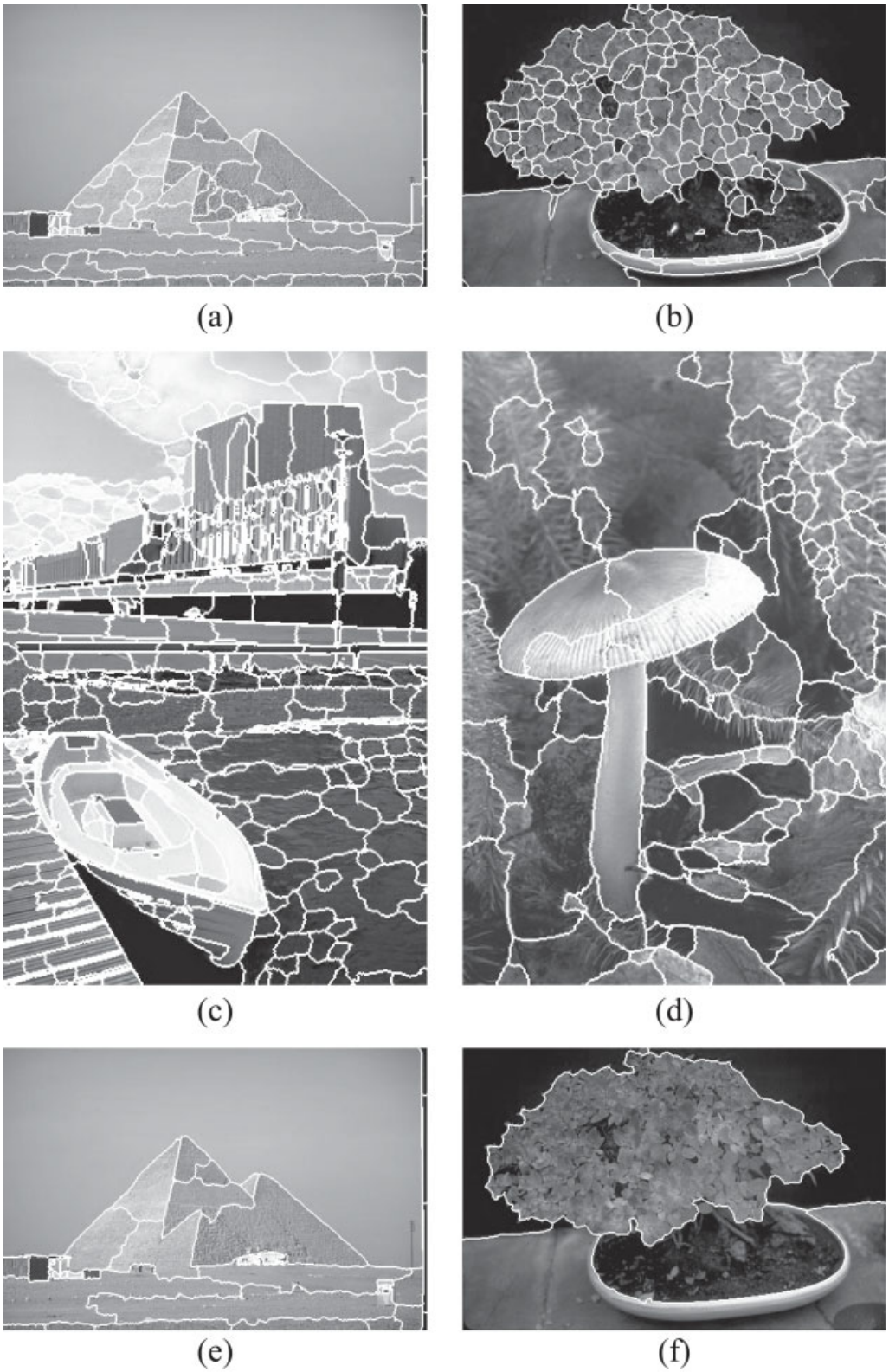
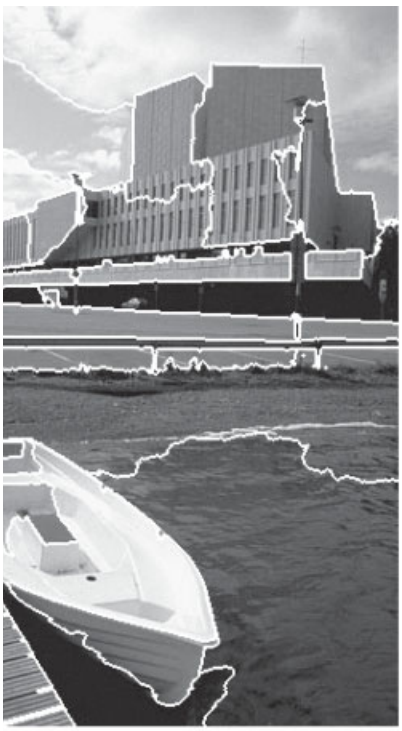


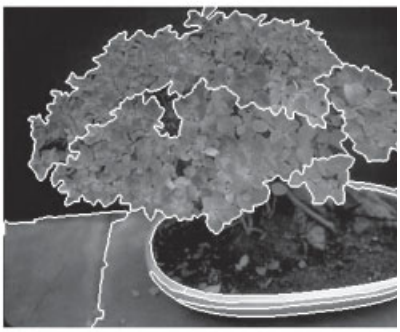
Figure 10. Clustering results using Algorithm 1 for f_2 with (a–d) $h(t) = \rho(t) - 1$ and (e–h) h from volume opening on ρ. (i-l) Results with IFT-watershed from gray-scale marker (Lotufo et al., 2002).



(g)

(h)

(i)



(j)

(k)

(l)

Figure 10. (*Continued*)

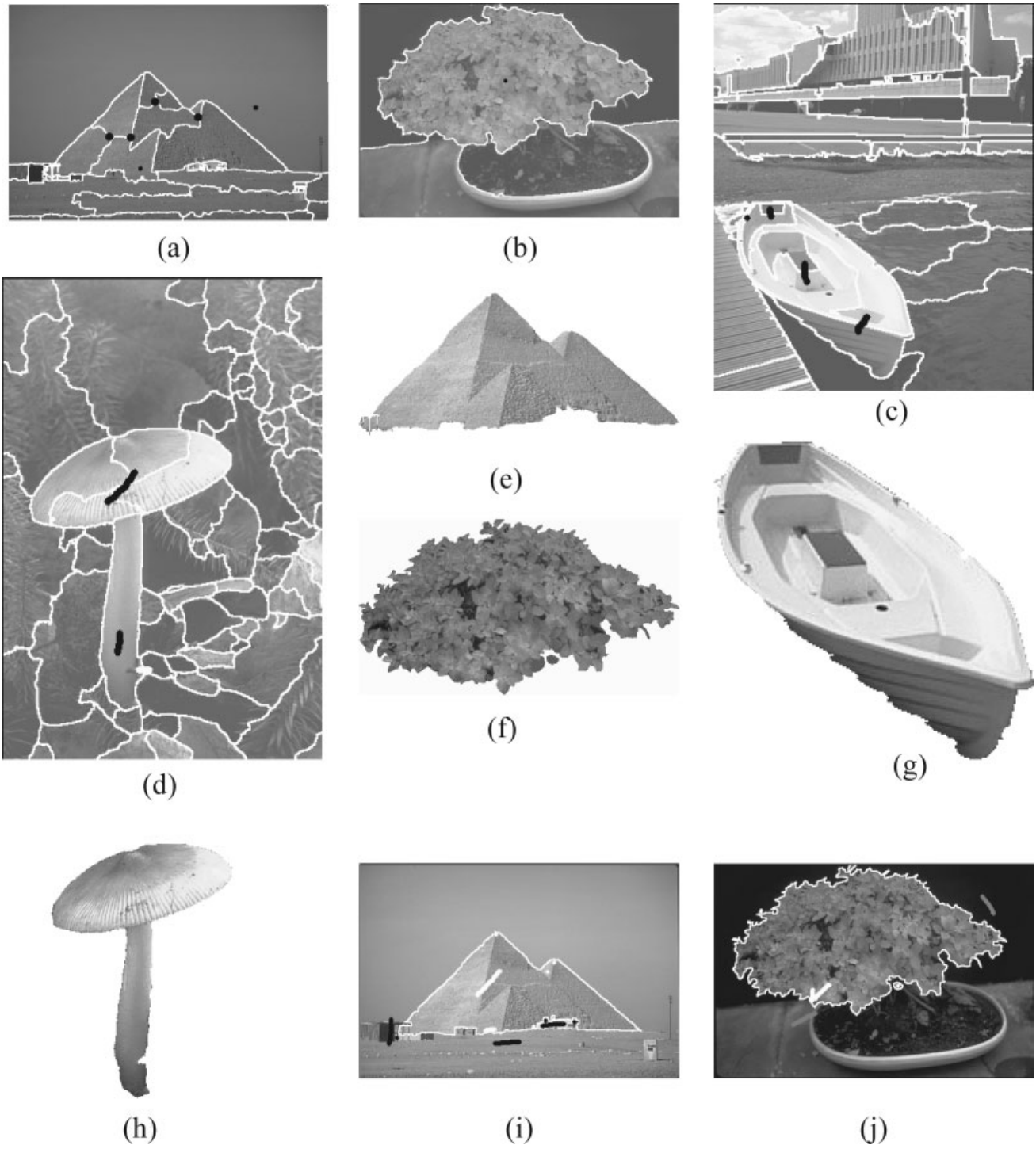


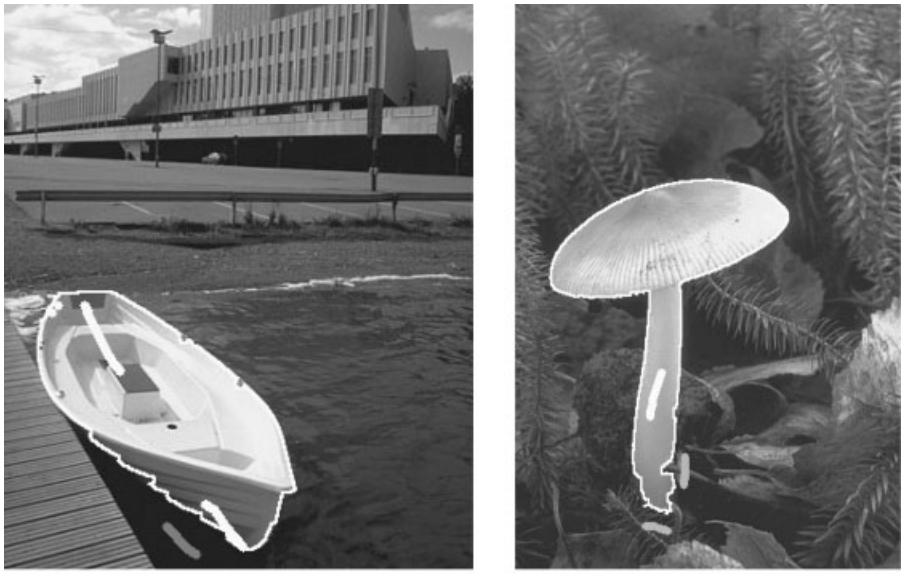
Figure 11. (a–d) The user selects markers to merge regions and/or separate object and background in a given region. (e–h) Segmentation results. (i–l) Similar results with the IFT-watershed transform from labeled markers. User's involvement can be reduced with the visual guidance of (a–d).

image graph $(\mathcal{N}, \mathcal{A}_5)$ to minimize a connectivity function f_3 [see the duality with Eq. (8)].

$$f_3(\langle t \rangle) = \begin{cases} G(t) \text{ if } t \in \mathcal{R} \\ +\infty \quad \text{otherwise} \end{cases} \qquad (20)$$
$$f_3(\pi_s \cdot \langle s,t \rangle) = \max\{f_3(\pi_s), G(t)\}.$$

The object region is redefined by the optimum-path forest rooted at the internal seeds.

Figures 11e–11h show the resulting segmentation from the markers and regions of Figures 11a–11d. Similar results could be obtained from the gradient images in Figures 9a–9d by using only the IFT-watershed transform from labeled markers (Figs. 11i–11l).



(k) (l)

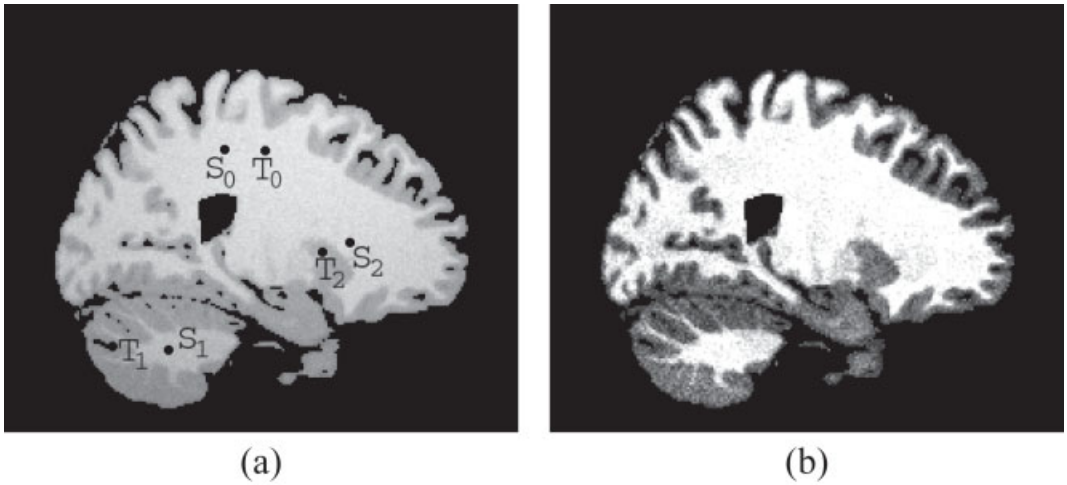
Figure 11. (*Continued*)

However, the proposed method helps the user to find directly the effective locations for the markers, usually reducing the number of markers and user's involvement.

B. MR-Images of the Brain. The classification of the brain tissues is a fundamental task in several medical applications (Kesslak et al., 1991; Jack et al., 1992; Zijdenbos et al., 1994; Juottonen et al., 1998). In this section, we present a fast, accurate and automatic approach for gray-matter (GM) and white-matter (WM) classification in MRT1-images of the brain, but it can be extended to other imaging protocols.

An MRT1-image of the brain is a pair $(\mathcal{N}, I)$, where $\mathcal{N}$ contains millions of voxels whose intensities $I(t)$ are usually darker in GM than in WM (exceptions might occur because of noise, inhomogeneity, and partial volume). Our problem consists of finding two clusters, one with GM voxels and the other with WM voxels. The clustering with size constraint is used for this purpose (Section IV.A.).

The most critical problem is the inhomogeneity. We first reduce it by transforming $I(t)$ into a new voxel intensity $J(t)$, $\forall t \in \mathcal{N}$ (Section VI.B.1). A graph $(\mathcal{N}', \mathcal{A}_3)$ is created by subsampling 0.02% of the voxels in $\mathcal{N}$, such that 0.01% of these voxels have values below the mean intensity inside the brain and 0.01% above it. This usually



(a) (b)

Figure 12. The effect of inhomogeneity in the original image (a) is not present in the corrected one (b). In (a), the inhomogeneity does not affect nearby voxels, such as S_i and T_i for $i = 0,1,2$, independent of their tissues. However, far away voxels from distinct tissues, such as S_1 and T_2, may be classified in the same cluster due to their nearby intensities ($I(S_1) = 1737$ and $I(T_2) = 1712$) and the interval of intensities between voxels from a same tissue ($I(S_2 - I(S_1) = 485$) and $I(T_2) - I(T_1) = 429$). In (b), the intensities within a same tissue are considerably reduced ($J(S_1) = J(S_2) = 156$ and $J(T_1) - J(T_2) = 107$), while the intensity difference between voxels from distinct tissues increases ($J(S_1) - J(T_2) = 704$).

Table III. GM classification of the synthetic images: mean and standard deviation of the Dice similarities using OPF_1 (Cappabianco, et al., 2008) the proposed method OPF_2, the hybrid approach of OPF with Bayes OPF_2+Bayes, and the Awate's method (Awate, 2006).

	Dice Similarity Mean ± Std. Dev. (%)			
Phantom GM	OPF_1	OPF_2	OPF_2 + Bayes	Awate
3%, 20%	95.15 ± 0.17	95.47 ± 0.05	95.50 ± 0.02	91.32
5%, 20%	95.10 ± 0.17	95.30 ± 0.05	95.51 ± 0.04	90.78
7%, 20%	94.36 ± 1.03	95.49 ± 0.02	95.00 ± 0.08	90.13
9%, 20%	94.06 ± 0.27	94.95 ± 0.01	93.98 ± 0.04	89.32
3%, 40%	90.90 ± 1.28	93.57 ± 0.07	93.50 ± 0.03	91.32
5%, 40%	91.23 ± 1.25	93.27 ± 0.08	93.51 ± 0.04	90.78
7%, 40%	91.10 ± 0.72	93.50 ± 0.03	92.91 ± 0.05	90.13
9%, 40%	90.66 ± 1.21	92.84 ± 0.02	92.30 ± 0.04	89.32

allows a fair amount of samples from both GM and WM tissues. A feature vector $\vec{v}(t)$ consists of the value $J(t)$ and the values of its 18 closest neighbors in the image domain. When a neighbor is out of the brain, we repeat $J(t)$ in the vector. The arc-weights are Euclidean distances between their corresponding feature vectors and the pdf is computed by Eq. (3) using the best value of $k \in \lceil 1, 30 \rceil$. The method usually finds two clusters within this range. When it finds more than two clusters, we force two clusters by assigning a GM label to those with mean intensity below the mean intensity in the brain and a WM label otherwise. Equation (15) is evaluated to classify the remaining voxels in $\mathcal{N} \setminus \mathcal{N}'$. Finally, the whole process is executed a few times (e.g., 7) and the class with majority vote is chosen for every voxel in order to guarantee stability. The method has been evaluated for real and synthetic images (Section VI.B.2). It represents an advance with respect to our previous approach (Cappabianco et al., 2008), which did not use neither inhomogeneity reduction nor majority vote.

B.1. Inhomogeneity Reduction. We reduce inhomogeneity based on three observations. First, it affects little the intensities of nearby voxels in a same tissue (e.g., S_o and T_o in Fig. 12a). Second, similar observation is valid for intensity differences between WM and GM voxels (e.g., S_i and T_i, $i = 1, 2$, in Figure 12a, respectively) in nearby regions of the image domain. Third, most voxels on the surface of the brain belongs to GM. The third observation led us to identify reference voxels for GM on the surface of the brain. Another clustering by optimum-path forest (OPF) is executed to divide the voxels on the surface of the brain into GM and WM voxels. The GM voxels are used as reference. Let t be a voxel in the brain, $C(t)$ be the closest reference voxel of t on the surface of the brain, and $\mathcal{V}_{C(t)}$ be the set of reference voxels within an adjacency radius equal to 6 mm from $C(t)$ in the image domain. The purpose $\mathcal{V}_{C(t)}$ is to avoid outliers among reference voxels. The new intensity $J(t)$ is the average of the following intensity differences.

$$J(t) = \frac{1}{|\mathcal{V}_{C(t)}|} \sum_{\forall r \in \mathcal{V}_{C(t)}} |I(t) - I(r)|. \quad (21)$$

After transformation, we expect similar intensities for GM voxels and similar intensities for WM voxels all over the brain (Fig. 12b), with higher differences between these tissues.

Table IV. WM classification of the synthetic images: mean and standard deviation of the Dice similarities using OPF_1 (Cappabianco, et al., 2008), the proposed method OPF_2, the hybrid approach of OPF with Bayes OPF_2+Bayes, and the Awate's method (Awate, 2006).

	Dice Similarity Mean ± Std. Dev. (%)			
Phantom WM	OPF_1	OPF_2	OPF_2+Bayes	Awate
3%, 20%	93.43 ± 0.19	94.10 ± 0.04	93.74 ± 0.06	94.85
5%, 20%	93.40 ± 0.20	93.89 ± 0.04	93.75 ± 0.09	94.27
7%, 20%	92.55 ± 0.93	93.91 ± 0.02	92.79 ± 0.16	93.66
9%, 20%	91.93 ± 0.54	93.08 ± 0.05	91.01 ± 0.09	92.94
3%, 40%	88.30 ± 0.64	91.75 ± 0.06	91.23 ± 0.04	94.85
5%, 40%	88.19 ± 0.67	91.40 ± 0.05	91.04 ± 0.10	94.27
7%, 40%	87.77 ± 0.81	91.39 ± 0.03	89.93 ± 0.13	93.66
9%, 40%	87.03 ± 0.73	90.45 ± 0.04	88.48 ± 0.10	92.94

Table V. GM classification of the ISBR images: mean and standard deviation of the Dice similarities using OPF_1 (Cappabianco, et al., 2008), the proposed method OPF_2, the hybrid approach OPF_2+Bayes, and two variants of the Awate's method (Awate, 2006) with different affinity thresholds.

	Dice Similarity ± Std. Dev. (%)			
IBSR GM	OPF_1	OPF_2	OPF_2 + Bayes	Awate
1	92.22 ± 0.87	90.33 ± 0.09	90.34 ± 0.12	83.33
2	90.99 ± 2.93	91.72 ± 0.02	87.54 ± 0.30	85.34
3	93.86 ± 0.14	91.99 ± 0.10	91.13 ± 0.13	87.25
4	88.19 ± 5.97	92.32 ± 0.10	90.33 ± 0.18	83.24
5	90.20 ± 1.73	90.33 ± 0.02	88.00 ± 0.09	86.41
6	85.02 ± 4.21	89.42 ± 0.05	89.68 ± 0.11	81.62
7	91.22 ± 3.35	91.34 ± 0.08	87.29 ± 0.15	81.07
8	88.46 ± 4.39	90.80 ± 0.02	88.27 ± 0.10	78.06

B.2. Evaluation. We selected eight synthetic images with 181 × 217 × 181 voxels from the Brainweb database,[†] with noise from 3, 5, 7, and 9%, and inhomogeneity 20 and 40%, respectively. We have also performed the same experiment for the first eight real images (with 9-bit intensity values) from the IBSR dataset.[‡] In those datasets, ground-truth images are available, and so we computed the Dice similarity between ground truth and the segmentation results. For each image, we executed the methods 9 times to compute mean and standard deviation of the Dice similarities. The methods OPF_1 and OPF_2 represent our previous (Cappabianco et al., 2008) and current approaches for GM/WM classification. The majority vote in OPF_2 was computed over seven executions. The classification of the remaining voxels by Eq. (15) can be substituted by a Bayesian classifier. By doing that, any loss in effectiveness reinforce the importance of the connectivity in the feature space for pattern classification. We then include a third approach, which uses OPF_2 to classify the subsamples $\mathcal{N}'$ followed by a Bayesian classifier on $\mathcal{N} \setminus \mathcal{N}'$ and majority vote over seven executions (OPF_2 + *Bayes*). We have also obtained from Awate the results of their clustering approach based on Markov model and registration with a probabilistic atlas (Awate et al., 2006).

Tables III and IV show the classification results for GM and WM on the synthetic images. The results on the ISBR images are shown in Tables V and VI. In the case of the ISBR images, there are two variants of the Awate's method based on different affinity thresholds for registration with the probabilistic atlas. The mean effectiveness of OPF_2 is superior than those obtained by OPF_1 and OPF_2 + *Bayes*. The inhomogeneity reduction and majority vote usually improve the clustering by OPF, and the connectivity in the

[†]URL: http://www.bic.mni.mcgill.ca/brainweb
[‡]URL: www.cma.mgh.harvard.edu/ibsr

Table VI. WM classification of the ISBR images: mean and standard deviation of the dice similarities using OPF_1 (Cappabianco, et al., 2008), the proposed method OPF_2, the hybrid approach OPF_2+Bayes, and two variants of the Awate's method (Awate, 2006) with different affinity thresholds.

	Dice Similarity Mean ± Std. Dev. (%)			
IBSR WM	OPF_1	OPF_2	OPF_2 + Bayes	Awate
1	84.98 ± 2.03	84.41 ± 0.10	77.14 ± 0.57	85.25
2	86.55 ± 2.93	87.96 ± 0.09	74.10 ± 1.07	87.78
3	86.07 ± 0.85	85.61 ± 0.11	77.17 ± 0.56	84.42
4	85.99 ± 3.31	86.07 ± 0.11	73.60 ± 0.82	78.91
5	84.59 ± 1.40	85.54 ± 0.07	74.83 ± 0.38	86.37
6	83.00 ± 3.32	87.94 ± 0.05	88.11 ± 0.80	83.28
7	87.39 ± 2.79	87.04 ± 0.25	74.86 ± 0.50	86.58
8	86.05 ± 3.41	88.09 ± 0.09	79.43 ± 0.45	83.20

feature space [Eq. (15)] seems to be important for classification. These results are also good as compared with those obtained by the Awate's method. Given that the standard deviation of their method seems to be very small (Awate et al., 2006), we may conclude that their approach better classifies WM than GM, as compared with OPF_2. The computational time for each execution of the OPF clustering is about 50 seconds on modern PCs, plus 20 seconds for inhomogeneity reduction. Five executions are usually enough to obtain good results with majority vote. Therefore GM/WM classification can take about 5.33 minutes using OPF_2, being about six times faster than the approach proposed in (Awate et al., 2006).

VII. CONCLUSIONS

We presented a clustering approach based on optimum-path forest (OPF) with two possible extensions to large datasets. The method identifies the influence zones of relevant maxima of the pdf based on the choice of a connectivity function. We showed the advantages of the OPF clustering over some baseline approaches, which include theoretical aspects and practical results. We also included experimental comparisons with other clustering methods, which output a desired number of clusters, and showed that OPF can achieve good and better results in some cases. OPF was shown to be fast and accurate for automatic GM/WM classification using real and synthetic images, and useful to guide the user's actions in the interactive segmentation of natural scenes. The results of OPF for GM/WM classification were similar to those obtained by (Awate et al., 2006), being usually better for GM than WM and about six times faster than that.

The effectiveness of the OPF clustering depends on the features, distance function, optimality criterion for the best k, $k_{\max}$, and $h(t)$. In the case of large datasets, it also depends on a representative subsampling process. These aspects need further investigation in the context of each application. The user can also provide labeled subsamples by drawing markers in the image and the OPF approach can be easily extended to supervised and semisupervised classification. This was not exploited for interactive segmentation, but the idea is the same. The subsampling process and pdf estimation can also take advantage of the registration with a probabilistic atlas for GM/WM separation. Our future work goes in this direction.

ACKNOWLEDGMENTS

This paper was presented at the 12th International Workshop on Combinatorial Image Analysis (Rocha et al., 2008).

REFERENCES

N. Arica and F.T.Y. Vural, BAS: A perceptual shape descriptor based on the beam angle statistics. Pattern Recognition Lett 24 (2003), 1627–1639.

S.P. Awate, T. Tasdizen, N. Foster, and R.T. Whitaker, Adaptive Markov modeling for mutual-information-based, unsupervised MRI brain-tissue classification. Med Image Anal 10 (2006), 726–739.

S. Beucher and C. Lantuejoul, Use of watersheds in contour detection, Internation workshop on image processing, real-time edge and motion detection/estimation, Rennes, 1979, pp. 17–21.

J.C. Bezdek, Pattern recognition with fuzzy objective function algorithms. Kluwer, Kluwer Academic Publishers Norwell, MA, 1981.

F.A.M. Cappabianco, A.X. Falcão, and L.M. Rocha, Clustering by optimum path forest and its application to automatic GM/WM classification in MR-T1 images of the brain, The Fifth IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI), 2008, pp. 428–431.

Y. Cheng, Mean shift, mode seeking, and clustering, IEEE trans pattern analysis machine intelligence 17 (1995), 790–799.

D. Comaniciu, An algorithm for data-driven bandwidth selection, IEEE trans pattern analysis machine intelligence 25 (2003), 281–288.

D. Comaniciu and P. Meer, A robust approach toward feature space analysis, IEEE Trans Pattern Analysis Machine Intelligence 24 (2002), 603–619.

D. Comaniciu and P. Meer, Kernel-based object tracking. IEEE Trans Pattern Analysis Machine Intelligence, 25 (2003), 564–577.

D. Comaniciu, V. Ramesh, and P. Meer, Real-Time tracking of non-rigid objects using mean shift, IEEE conference on computer vision and pattern, 2000, pp. 142–151.

D. Comaniciu, V. Ramesh, and P. Meer, The variable bandwidth mean shift and data-driven scale selection, Proc of the eighth IEEE international conference on computer vision, volume 1, 2001, pp. 438–445.

D. DeMenthon, Spatio-temporal segmentation of video by hierarchical mean shift analysis, Proc of statistical methods in video processing workshop, 2002.

A.P. Dempster, N.M. Laird, and D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, J Royal Stat Soc Series B Methodological 39 (1977), 1–38.

R.O. Duda, P.E. Hart and D.G. Stork, Pattern classification, Wiley-Interscience, New York 2, 2001.

A.X. Falcão, J. Stolfi, and R.A. Lotufo, The image foresting transform: Theory, algorithms, and applications, IEEE Trans Pattern Analysis Machine Intelligence 26 (2004), 19–29.

A.X. Falcão and J.K. Udupa, A 3D generalization of user-steered live wire segmentation, Med Imag Anal 4 (2000), 389–402.

A.X. Falcão, B.S. da Cunha, and R.A. Lotufo, Design of connected operators using the image foresting transform, Proc of SPIE Med Imaging 4322 (2001), pp. 468–479.

B. Georgescu, I. Shimshoni, and P. Meer, Mean shift based clustering in high dimensions: A texture classification example, IEEE Comput Soc (2003), 456–463.

M. Halkidi and M. Vazirgiannis, Clustering validity assessment: Finding the optimal partitioning of a data set, Proc of the IEEE Intl Conf on Data Mining, 2001, pp. 187–194.

M. Herbin, N. Bonnet and P. Vautrot, A clustering method based on the estimation of the probability density function and on the skeleton by influence zones, Pattern Recognition Lett (1996), 1141–1150.

G.T. Herman and B.M. Carvalho, Multiseeded segmentation using fuzzy connectedness. IEEE Trans on Pattern Analysis Machine Intelligence 23 (2001), 460–474.

L.J. Hubert, Some applications of graph theory to clustering, Psychometrika 39 (1974), 283–309.

C.R. Jack, R.C. Petersen, P.C O'Brien, E.G Tangalos, MR-based hippocampal volumetry in the diagnosis of Alzheimer's disease, Neurology 42 (1992), 183–188.

A.K. Jain, and R.C. Dubes, Algorithms for clustering data, Prentice-Hall, Englewood Cliffs, NJ, 1988.

A.K. Jain, R.P.W. Duin and J. Mao, Statistical pattern recognition: A review, IEEE Trans Pattern Analysis Machine Intelligence 22 (2000), 4–37.

K. Juottonen, M. Lehtovirta, P.J.R. Helisalmi, S Sr., and H. Soininen, Major decrease in the volume of the entorhinal cortex in patients with Alzheimer's disease carrying the apolipoprotein and $\varepsilon 4$ allele, J Neurol Neurosurg Psych 65 (1998), 322–327.

V. Katkovnik and I. Shmulevich, Nonparametric density estimation with adaptive varying window size, Proc of the conf on image and signal processing for remote sensing, 2000, pp. 25–29.

J.P. Kesslak, O. Nalcioglu, and C.W. Cotman, Quantification of magnetic resonance scans for hippocampal and parahippocampal atrophy in Alzheimer's disease, Neurology 41 (1991), 51.

T. Lindeberg, Scale-space theory: A basic tool for analysing structures at different scales, J Appl Stat 21 (1994), 224–270.

R.A. Lotufo, and A.X. Falcão, The ordered queue and the optimality of the watershed approaches, In Mathematical Morphology and its Applications to Image and Signal Processing, Vol. 18. Kluwer, Palo Alto (CA), 2000, pp. 341–350.

R.A. Lotufo, A.X. Falcão and F. Zampirolli, IFT-Watershed from Gray-Scale Marker, IEEE Comput Soc (2002), 146–152.

U. von Luxburg, A tutorial on spectral clustering, Stat Comp 17 (2007), pp. 395–416.

J.B. MacQueen, Some methods for classification and analysis of multivariate observations, University of California Press, Berkely, CA, 1967, pp. 281–297.

F. Meyer, Levelings, image simplification filters for segmentation, J Math Imaging Vision 20 (2004), 59–72.

MPEG-7, "MPEG-7: The Generic Multimedia Content Description Standard, Part 1." IEEE MultiMedia 09, 2002, pp. 78–87.

D.J. Newman and A. Asuncion, "UCI machine learning repository." UCI machine learning repository, Irvine, CA, 2007.

J.P. Papa, A.X. Falcão, C.T.N. Suzuki, and N.D.A. Mascarenhas, "A discrete approach for supervised pattern recognition," Combinatorial image analysis, In Lecture Notes in Computer Science, V.E. Brimkov, R.P. Barneva, and H.A. Hauptman (Editors), Springer, Berlin Heidelberg, Vol. 4958, 2008, pp. 136–147.

E. Persoon and K. Fu. Shape discrimination using fourier descriptors, IEEE Trans Sys Man Cybernetics 7 (1977), 170–178.

L.M. Rocha, A.X. Falcão, L.G.P. Meloni, "A robust extension of the mean shift algorithm," Image Analysis In Theory to Applications, R.P. Barneva and V.E. Brimkov (Editors), Research Publishing, Singapore, 2008, pp. 29–38.

P.K. Saha and J.K. Udupa, Relative fuzzy connectedness among multiple objects: Theory, algorithms, and applications in image segmentation., Comput Vision Image Understanding 82 (2001), 42–56.

P. Salembierand and J. Serra, Flat zones filtering, connected operators, and filters by reconstruction, IEEE Trans Image Processing 4 (1995), 1153–1160.

P. Salembier and L. Garrido, Connected operators based on region-tree pruning strategies, 15th Intl Conf on Pattern Recognition 03, 2000, pp. 3371.

P. Salembier, A. Oliveras and L. Garrido, Antiextensive connected operators for image and sequence processing. IEEE Trans Image Processing 7, 1998, pp. 555–570.

J. Shi and J. Malik, Normalized cuts and image segmentation. IEEE Trans Pattern Analysis Machine Intelligence 22, 2000, pp. 888–905.

S. Theodoridis and K. Koutroubas, Pattern recognition, Academic Press, New York, 1999.

R. Torres, A.X. Falcão, and L.F. Costa, A graph-based approach for multiscale shape analysis, Pattern Recognition 37, 2004, 1163–1174.

L. Vincent, Morphological Grayscale Reconstruction in Image Analysis, IEEE Trans Image Proc 2, 1993, pp. 176–201.

J. Wang, B. Thiesson, Y. Xu, and M. Cohen, Image and video segmentation by anisotropic kernel mean shift, Springer, Berlin/Heidelberg, 2004, pp. 238–249.

Z. Wu and R. Leahy, An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation, IEEE Trans Pattern Analysis Machine Intelligence 15, 1993, pp. 1101–1113.

C. Yang, R. Duraiswami, and L. Davis. Efficient mean-shift tracking via a new similarity measure, IEEE Comput Soc 2005, 176–183.

C.T. Zahn, Graph-theoretical methods for detecting and describing gestalt clusters, IEEE Trans Comput C-20 (1971), 68–86.

A.P. Zijdenbos and B.M. Dawant, Brain segmentation and white matter lesion detection in MR images, Crit Rev Biomed Eng 22 (1994), 401–465.