Image Segmentation Using Information Bottleneck Method

Anton Bardera, Jaume Rigau, Imma Boada, Miquel Feixas, and Mateu Sbert

***Abstract*—In image processing, segmentation algorithms constitute one of the main focuses of research. In this paper, new image segmentation algorithms based on a hard version of the information bottleneck method are presented. The objective of this method is to extract a compact representation of a variable, considered the input, with minimal loss of mutual information with respect to another variable, considered the output. First, we introduce a split-and-merge algorithm based on the definition of an information channel between a set of regions (input) of the image and the intensity histogram bins (output). From this channel, the maximization of the mutual information gain is used to optimize the image partitioning. Then, the merging process of the regions obtained in the previous phase is carried out by minimizing the loss of mutual information. From the inversion of the above channel, we also present a new histogram clustering algorithm based on the minimization of the mutual information loss, where now the input variable represents the histogram bins and the output is given by the set of regions obtained from the above split-and-merge algorithm. Finally, we introduce two new clustering algorithms which show how the information bottleneck method can be applied to the registration channel obtained when two multimodal images are correctly aligned. Different experiments on 2-D and 3-D images show the behavior of the proposed algorithms.**

***Index Terms*—Image registration, image segmentation, information bottleneck method, information theory.**

I. Introduction

THE main objective of image segmentation is to divide an image into regions that can be considered homogeneous with respect to a given criterion such as color or texture. Image segmentation is one of the most widely studied problems in image analysis and computer vision and, it is a significant step towards image understanding. Many different methods, such as thresholding, region growing, region splitting and merging, active contours, and level sets, have been proposed. Each one of these methods considers the segmentation problem from a different perspective and is suitable for solving a limited number of situations. For a survey of segmentation algorithms, see [11], [13].

The purpose of this paper is to introduce new segmentation algorithms using a hard version of the *information bottleneck method* [29]. The use of this method requires the definition of an information channel where a random variable *controls* the clustering of the other by preserving the maximum mutual information between them. That is, the objective of this method is to extract a compact representation of a random variable with minimal loss of mutual information with respect to another variable.

In this paper, the information bottleneck method will be applied to two different channels: i) the channel defined between the set of regions of a given image and its histogram bins, and ii) the channel built between the histogram bins of two multimodal registered images. From the first channel, both split-and-merge and histogram clustering algorithms are introduced and, from the second channel, both one-sided and two-sided histogram clustering algorithms are presented. While the splitting process is guided by the maximization of the mutual information gain, all the other processes (merging and clustering) are driven by the minimization of the mutual information loss.

The following information-bottleneck-based algorithms represent the main contributions of this paper.

- *Split-and-merge algorithm* (Section III). In the first phase, a top-down strategy is applied to partition an image into quasi-homogeneous regions using a binary space partition (BSP) or a quadtree partition. In the second phase, a bottom-up strategy is used to merge the regions whose histograms are more similar.
- *Histogram clustering algorithm* (Section IV). Neighbor bins of the histogram are clustered from a previously partitioned image. After assuming that the split-and-merge algorithm provides us with the structure of the image, our clustering algorithm tries to preserve the correlation between the clustered bins and the structure of the image.
- *Histogram clustering algorithms for two registered multimodal images* (Section V). Two different algorithms are presented. The first one segments just one image at a time, while the second one segments both simultaneously. The clustering process works by extracting from each image the structures that are more relevant to the other one. In these algorithms, each image is used to control the quality of the segmentation of the other.

The proposed methods have several advantages. In the split-and-merge algorithm, this channel makes the correspondence between the structure of the image and the histogram bins. This spatial information makes the method robust to texture analysis, without assuming any *a priori* intensity or texture distribution. The proposed histogram clustering algorithm considers the spatial distribution of the intensities to achieve a good representation of the colors of the image. The obtained segmentation tries to preserve with a given number of colors the maximum spatial

Manuscript received June 05, 2008; revised February 16, 2009. First published May 12, 2009; current version published June 12, 2009. This work was supported in part by Grants TIN2007-67982-C02 and TIN2007-68066-C04-01. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. John Kerekes.

The authors are with the Graphics and Imaging Laboratory, University of Girona, Spain (e-mail: anton.bardera@ima.udg.edu; rigau@ima.udg.edu; imma@ima.udg.edu; feixas@ima.udg.edu; mateu@ima.udg.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2009.2017823

information of the original image. Finally, the registration-based segmentation is able to segment one image from the information of another. For instance, this algorithm enables us to segment images of low quality from the information contained in high quality images. This technique could be used to segment intraoperative images using high quality preoperative ones. A global advantage of these methods is that they do not assume any *a priori* information about the images (e.g., intensity probability distribution). The results of our experiments show the feasibility of the information bottleneck method to deal with different 2-D and 3-D image segmentation techniques.

II. Previous Work

In this section, we review some basic concepts on image segmentation [13], information theory [8], and information bottleneck method [26], [29].

A. Image Segmentation

In image processing, grouping parts of an image into regions that are homogeneous with respect to one or more features results in a segmented image. Segmentation algorithms are generally based on one of two basic properties of intensity values: discontinuity and similarity. In the first category, the algorithm partitions the image based on abrupt changes in intensity, such as edges [6], [24]. The principal approaches in the second category are based on partitioning an image into regions that are similar according to a set of predefined criteria. Thresholding, region growing, histogram clustering, split-and-merge, and random fields are examples of methods of this category [1], [2], [10], [12], [13], [18], [30]. For our purposes, we briefly review the thresholding, histogram clustering, and split-and-merge algorithms.

Thresholding [13], [23], [28] is a basic technique of image segmentation with a significant degree of popularity, especially in applications where speed is an important factor. The thresholding algorithm provides a number of threshold levels, which determine the region in which each pixel belongs depending on its intensity value. In order to find these thresholds, almost all methods analyze the histogram of the image. In most cases, the optimal thresholds are found by either minimizing or maximizing an objective function, which depends on the positions of the thresholds. Thresholding is best suited for bimodal distribution, such as solid objects resting upon a contrasted background [23]. A similar approach is given by image clustering algorithms (e.g., k-means algorithm [15]), which discover groups of similar intensity values. These methods tackle the segmentation problem from a different perspective: instead of finding the levels which separate one group from the other (as thresholding techniques do), they group similar bins. This notion of similarity can be expressed in very different ways, according to the purpose of the segmentation, the domain-specific assumptions, and the prior knowledge of the problem. Image clustering is traditionally seen as part of unsupervised learning [14].

The split-and-merge algorithm [11], [13], [16], [28] is composed by two steps. First, the method subdivides the entire image into smaller regions following a dissimilarity criterion. To divide the image, different strategies can be adopted, such as a quadtree partition (where each region is subdivided into four equal regions) and a binary space partition (BSP) (where an optimal partition is selected to divide the region). Second, the neighbor regions obtained from the splitting step are merged if they verify a similarity criterion. These similarity and dissimilarity criteria can be based on an intensity range, gradient, contrast, region statistics, or texture. The combination of splitting and merging steps allows for the segmentation of arbitrary shapes, which are not constrained to vertical or horizontal lines, as occurs if only the splitting step is considered.

B. Information Theory

Let $\mathcal{X}$ be a finite set and X a random variable taking values x in $\mathcal{X}$ with distribution $p(x) = Pr[X = x]$. Likewise, let Y be a random variable taking values y in $\mathcal{Y}$. An information channel $X \rightarrow Y$ between the random variable X (input) and Y (output) is characterized by a *probability transition matrix* (composed of conditional probabilities) which determines the output distribution given the input [8].

The *Shannon entropy* $H(X)$ of a random variable X is defined by

$$H(X) = -\sum_{x\in\mathcal{X}} p(x)\log p(x). \tag{1}$$

It is also denoted by $H(p)$ and measures the average uncertainty of a random variable X. All logarithms are base 2 and entropy is expressed in bits. The convention $0\log 0 = 0$ is used. The *conditional entropy* is defined by

$$H(Y|X) = -\sum_{x\in\mathcal{X}} p(x)\sum_{y\in\mathcal{Y}} p(y|x)\log p(y|x) \tag{2}$$

where $p(y|x) = Pr[Y = y|X = x]$ is the conditional probability. The conditional entropy $H(Y|X)$ measures the average uncertainty associated with Y if we know the outcome of X. In general, $H(Y|X) \neq H(X|Y)$, and $H(X) \geq H(X|Y) \geq 0$.

The *mutual information* (MI) between X and Y is defined by

$$\begin{aligned} I(X,Y) &= H(X) - H(X|Y) && (3)\\ &= \sum_{x\in\mathcal{X}} p(x)\sum_{y\in\mathcal{Y}} p(y|x)\log\frac{p(y|x)}{p(y)} && (4)\end{aligned}$$

and measures the shared information between X and Y. It can be seen that $I(X,Y) = I(Y,X) \geq 0$ [8]. A fundamental property of MI is given by the *data processing inequality* which can be expressed in the following way: if $X \rightarrow Y \rightarrow Z$ is a Markov chain, i.e., $p(x,y,z) = p(x)p(y|x)p(z|y)$, then

$$I(X,Y) \geq I(X,Z). \tag{5}$$

This result demonstrates that no processing of Y, deterministic or random, can increase the information that Y contains about X.

A convex function f on the interval $[a,b]$ fulfils the *Jensen inequality*: $\sum_{i=1}^{n}\lambda_i f(x_i) - f(\sum_{i=1}^{n}\lambda_i x_i) \geq 0$, where $0 \leq \lambda \leq 1$, $\sum_{i=1}^{n}\lambda_i = 1$, and $x_i \in [a,b]$. For a concave function, the inequality is reversed. If f is substituted by the Shannon entropy, which is a concave function, we obtain the *Jensen–Shannon inequality* [5]

$$JS(\pi_1,\ldots,\pi_n;p_1,\ldots,p_n)\equiv H\left(\sum_{i=1}^{n}\pi_i p_i\right)-\sum_{i=1}^{n}\pi_i H(p_i)\geq 0 \tag{6}$$

where $JS(\pi_1,\ldots,\pi_n;p_1,\ldots,p_n)$ is the *Jensen–Shannon divergence* of probability distributions $\{p_1,\ldots,p_n\}$ with prior probabilities or weights $\{\pi_1,\ldots,\pi_n\}$ fulfilling $\sum_{i=1}^{n}\pi_i=1$. The JS-divergence measures how far the probabilities p_i are from their likely joint source $\sum_{i=1}^{n}\pi_i p_i$ and equals zero if and only if all the p_i are equal. It is important to note that the JS-divergence is identical to $I(X,Y)$ when $\{\pi_1,\ldots,\pi_n\}$ and $\{p_1,\ldots,p_n\}$ represent, respectively, the input distribution and the probability transition matrix of the channel $X\rightarrow Y$, where $n=|\mathcal{X}|$ and $m=|\mathcal{Y}|$. That is, $\forall i\in\{1\ldots n\}.\pi_i=p(x_i)$ and $\forall i\in\{1\ldots n\}.p_i=p(Y|x_i)$, where $p(Y|x_i)=\{p(y_1|x_i),\ldots,p(y_m|x_i)\}$ is the conditional probability disribution [5], [26].

C. Information Bottleneck Method

The information bottleneck method, introduced by Tishby *et al.* [29], extracts a compact representation of the variable X, denoted by $\widehat{X}$, with minimal loss of MI with respect to another variable Y (i.e., $\widehat{X}$ preserves as much information as possible about the relevant variable Y). Soft [29] and hard [25] partitions of X can be adopted. In the first case, every cluster $x\in\mathcal{X}$ can be assigned to every cluster $\hat{x}\in\widehat{\mathcal{X}}$ with some conditional probability $p(\hat{x}|x)$ (soft clustering). In the second case, every cluster $x\in\mathcal{X}$ is assigned to only one cluster $\hat{x}\in\widehat{\mathcal{X}}$ (hard clustering).

In this paper, we focus our attention on the *agglomerative information bottleneck method* [25]. Given a cluster $\hat{x}$ defined by $\hat{x}=\{x_1,\ldots,x_l\}$, where $x_k\in\mathcal{X}$, and given probability distributions $p(\hat{x})$ and $p(y|\hat{x})$ defined by

$$p(\hat{x})=\sum_{k=1}^{l}p(x_k) \tag{7}$$

$$p(y|\hat{x})=\frac{1}{p(\hat{x})}\sum_{k=1}^{l}p(x_k,y)\quad\forall y\in\mathcal{Y} \tag{8}$$

the following properties are fulfilled.

- The decrease in the mutual information from $I(X,Y)$ to $I(\widehat{X},Y)$ due to the merge of $x_1,\ldots,x_l$ is given by

  $$\delta I_{\hat{x}}=p(\hat{x})JS(\pi_1,\ldots,\pi_l;p_1,\ldots,p_l)\geq 0 \tag{9}$$

  where $\pi_k=(p(x_k)/p(\hat{x}))$ and $p_k=p(Y|x_k)$. An optimal clustering algorithm has to minimize $\delta I_{\hat{x}}$.
- An optimal merge of l components can be obtained by $l-1$ consecutive optimal merges of pairs of components.

Dhillon *et al.* [9] presented a co-clustering algorithm applied to word-document clustering that simultaneously clusters X and Y into disjoint or hard clusters. An optimal co-clustering algorithm has to minimize the difference $I(X,Y)-I(\widehat{X},\widehat{Y})$.

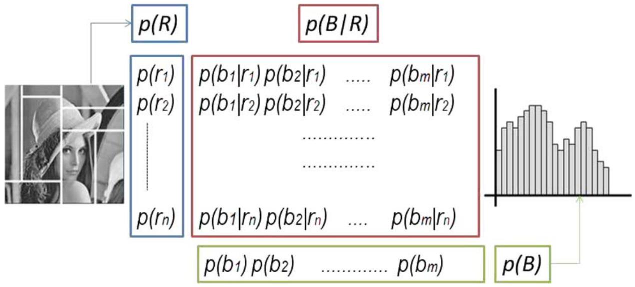
p(R)
p(B|R)
p(r1)
p(r2)
p(rn)
p(b1|r1) p(b2|r1) ..... p(bm|r1)
p(b1|r2) p(b2|r2) ..... p(bm|r2)
....
....
p(b1|rn) p(b2|rn) .... p(bm|rn)
p(b1) p(b2) .... p(bm)
p(B)

Fig. 1. Information channel between the regions of the images (R) and the intensity bins (B) for the split-and-merge algorithm. The reverse of this channel used in the histogram clustering algorithm.

III. Split-and-Merge Algorithm

In this section, we present an split-and-merge algorithm that is constructed from an information channel $R\rightarrow B$ between the random variables R (input) and B (output), which represent, respectively, the set of regions $\mathcal{R}$ of an image and the set of intensity bins $\mathcal{B}$ (see Fig. 1). This channel is defined by a conditional probability matrix $p(B|R)$ which expresses how the pixels corresponding to each region of the image are distributed into the histogram bins. Throughout this paper, the capital letters R and B as arguments of $p()$ will be used to denote probability distributions. For instance, while $p(R)$ will represent the input distribution of the regions, $p(r)$ will denote the probability of a single region r.

Given an image with N pixels, N_{r} regions, and N_{b} intensity bins, the three basic elements of the channel $R\rightarrow B$ are as follows.

- The conditional probability matrix $p(B|R)$, which represents the transition probabilities from each region of the image to the bins of the histogram, is defined by $p(b|r)=(n(r,b)/n(r))$, where $n(r)$ is the number of pixels of region r and $n(r,b)$ is the number of pixels of region r corresponding to bin b. Conditional probabilities fulfil $\sum_{b\in\mathcal{B}}p(b|r)=1,\forall r\in\mathcal{R}$.
- The input distribution $p(R)$, which represents the probability of selecting each image region, is defined by $p(r)=(n(r)/N)$ (i.e., the relative area of region r).
- The output distribution $p(B)$, which represents the normalized frequency of each bin b, is given by $p(b)=\sum_{r\in\mathcal{R}}p(r)p(b|r)=(n(b)/N)$, where $n(b)$ is the number of pixels corresponding to bin b.

From the data processing inequality (5) and the information bottleneck method (Section II-C), we know that any clustering or quantization over R or B, respectively represented by $\widehat{R}$ and $\widehat{B}$, will reduce $I(R,B)$. Thus, $I(R,B)\geq I(R,\widehat{B})$ and $I(R,B)\geq I(\widehat{R},B)$.

A. Splitting

The splitting phase of the algorithm is a greedy top-down procedure (see Fig. 2) which partitions an image in quasi-homogeneous regions. Our partitioning strategy takes the full image as the unique initial partition and progressively subdivides it [e.g., with vertical or horizontal lines in 2-D images (BSP)] chosen according to the maximum MI gain for each partitioning step. In our experiments, BSP and quad-tree strategies will be used.

```
Input
    Joint probability distribution: $p(x,y)$
    Number of clusters: $m \in \{1..|X|\}$
Output
    A partition of $X$ into $m$ clusters
Computation
    $\tilde{X} \leftarrow U$
    $\forall i \in \{1..|X|-1\}$.compute($\delta I_{\tilde{x}}(i)$) (see Eq. (12))
    while $|\tilde{X}| < m$ do
        $k \leftarrow \max_i(\delta I_{\tilde{x}}(i))$
        $\{x_k, x_{k+1}\} \leftarrow$ split($\tilde{x}$,$k$)
        $\tilde{X} \leftarrow (\tilde{X} - \tilde{x}) \bigcup \{x_k, x_{k+1}\}$
        Update $\delta I_{\tilde{x}}$ for $x_k$ and $x_{k+1}$
    end while
    return $\tilde{X}$
```

Fig. 2. Top-down bottleneck algorithm.

Note that similar algorithms have been introduced in the context of pattern recognition [22], learning [17], and DNA segmentation [4]. This splitting algorithm has been previously presented in [21].

The partitioning process is represented over the channel $\widetilde{R} \rightarrow B$, where $\widetilde{R}$ denotes that R is the variable to be partitioned. Note that this channel varies at each partition step because the number of regions is increased and, consequently, the marginal probabilities of $\widetilde{R}$ and the conditional probabilities of $\widetilde{R}$ known B also change. For a BSP strategy, the gain of MI due to the partition of a region $\tilde{r}$ in two neighbor regions r_1 and r_2, such that

$$p(\tilde{r}) = p(r_1) + p(r_2) \tag{10}$$

and

$$p(b|\tilde{r}) = \frac{p(r_1)p(b|r_1) + p(r_2)p(b|r_2)}{p(\tilde{r})} \tag{11}$$

is given by

$$\begin{aligned} \delta I_{\tilde{r}} &= I(R,B) - I(\widetilde{R},B) \\ &= p(\tilde{r}) JS\left(\pi_1, \pi_2; p(B|r_1), p(B|r_2)\right) \end{aligned} \tag{12}$$

where $\pi_1 = (p(r_1)/p(\tilde{r}))$ and $\pi_2 = (p(r_2)/p(\tilde{r}))$. The JS-divergence $JS(\pi_i, \pi_j; p(B|r_1), p(B|r_2))$ between two regions can be interpreted as a measure of *dissimilarity* between them respect to the intensity values. That is, when a region is partitioned, the gain of MI is equal to the degree of dissimilarity between the resulting regions times the size of the region. In our splitting algorithm, the optimal partition is determined by the the maximum MI gain $\delta I_{\tilde{r}}$.

The BSP partitioning algorithm can be represented by an evolving binary tree [22] where each leaf corresponds to a terminal region of the image. At each partitioning step, the tree gains information from the original image such that each internal node k contains the information I_k gained with its corresponding splitting. At a given moment, $I(R,B)$ can be obtained adding up the information available at the internal nodes of the tree weighted by $p(k)$, where $p(k) = (n(k)/N)$ is the relative area of the region associated with node k and $n(k)$ is the number of pixels of this region. Thus, the MI of the channel is given by

$$I(R,B) = \sum_{k=1}^{T} p(k) I_k \tag{13}$$

where T is the number of internal nodes. It is important to stress that the best partition can be decided locally. That is, the information gained I_k in a given node k is independent of the level of partitioning of the other regions of the image.

From the (3), the partitioning procedure can also be visualized as $H(B) = I(R,B) + H(B|R)$, where $H(B)$ is the histogram entropy and $I(B,R)$ and $H(B|R)$ represent, respectively, the successive values of MI and conditional entropy obtained after the successive partitions. The progressive acquisition of information increases $I(R,B)$ and decreases $H(B|R)$. This reduction of conditional entropy is due to the progressive homogenization of the resulting regions. Observe that the maximum MI that can be achieved is the histogram entropy $H(B)$, that remains constant along the process. The partitioning algorithm can be stopped using a ratio $MIR_{\mathrm{r}} = (I(R,B)/H(B))$ of mutual information gain or a predefined number of regions N_{r}.

Fig. 3(a) and (b) shows two test images used in our experiments. The first corresponds to the well-known Lena image and the second to a CT medical brain image with a hematoma lesion. In this paper, the segmentation of colored images is obtained using the luminance channel. The two curves in Fig. 3(c) indicate the behavior of MIR_{r} with respect to the number of partitions, which have been obtained using a BSP strategy for both test images. These plots show the concavity of the MIR_{r} function. It can be clearly appreciated that a big gain of MI is obtained with a low number of partitions. Thus, for instance, a 50% of MI is obtained with approximately 1% of the maximum number of partitions for the hematoma test image. Observe that in the hematoma image less partitions are needed to extract the same MIR_{r} than in the Lena image, due to the higher heterogeneity of the latter image. Note also that $MIR_{\mathrm{r}} = 1$ is achieved with approximately 50% of the regions in the hematoma image, since these are completely homogeneous.

Fig. 4 presents the results of partitioning the hematoma test image. We show the partitioned images corresponding to two different MIR_{r} for quadtree and BSP simplifications. Observe that, for the same quantity of extracted information, the BSP partition fits better to the image structure, due to the higher flexibility of this scheme. For instance, observe how the first BSP partitions of the hematoma image [Fig. 4(c)] try to separate the brain structure from the background. Despite these interesting results, they can not be used by themselves as a final segmentation and a merging process is needed to achieve a correct image segmentation. This merging process is explained in the next section.

B. Merging

From the agglomerative information bottleneck method [25] applied to the channel $R \rightarrow B$, we know that any clustering over R will not increase $I(R,B)$. Analogous to the MI gain

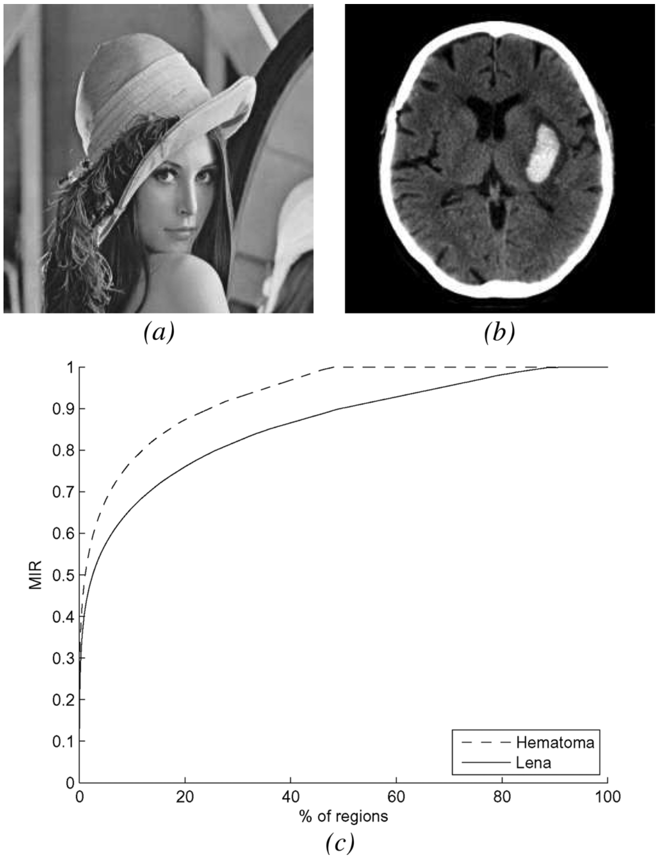

Fig. 3. Test images: (a) Lena and (b) hematoma. The two plots in (c) show the mutual information ratio (MIR_{r}) with respect to the number of regions for (a) and (b).

(12) obtained in the splitting phase, the loss of MI due to the clustering $\hat{r}$ of two neighbor regions r_1 and r_2 is given by

$$\begin{aligned}\delta I_{\hat{r}} &= I(R,B) - I(\widehat{R},B)\\ &= p(\hat{r}) JS\left(\pi_1, \pi_2; p(B|r_1), p(B|r_2)\right) \qquad (14)\end{aligned}$$

where $p(\hat{r}) = p(r_1) + p(r_2)$, $\pi_1 = (p(r_1)/p(\hat{r}))$, $\pi_2 = (p(r_2)/p(\hat{r}))$, and $p(b|\hat{r}) = (p(r_1)p(b|r_1) + p(r_2)p(b|r_2)/p(\hat{r}))$, and $\widehat{X}$ denotes that the variable X has been clustered.

As we have seen in the splitting phase, the JS-divergence between two regions can be interpreted as a measure of *dissimilarity* between them. The similarity will be maximum when the two regions have the same histogram: if $p(B|r_1) = p(B|r_2)$, then $\delta I_{\hat{r}} = 0$. Thus, if two regions are very similar (i.e., the JS-divergence between them is small) the channel could be simplified by substituting these two regions by their merging, without a significant loss of information. This is the principle that leads to the following merging algorithm.

From a given image partitioning, the algorithm merges successively the pairs (r_1, r_2) of neighbor regions such that $\delta I_{\hat{r}}$ is minimum (see Fig. 5). Thus, the number of regions decreases progressively together with the MI of the channel. Similarly to the splitting algorithm, the stopping criterion can be determined by the ratio $MIR_{\mathrm{r}} = (I(R,B)/H(B))$ or a predefined number of regions.

Note that the clustering $\widehat{R}$ of all regions would give $I(B,\widehat{R}) = 0$. From (3), during the merging process

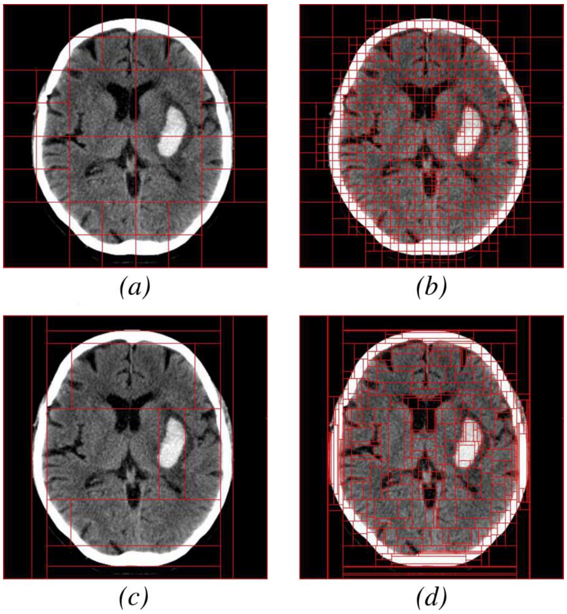

Fig. 4. Partition of the hematoma image [Fig. 3(b)] with two different MIR_{r} for (a–b) quadtree and (c–d) BSP simplifications. (a) $MIR_{\mathrm{r}} = 0.2$; (b)$MIR_{\mathrm{r}} = 0.4$; (c) $MIR_{\mathrm{r}} = 0.2$.

```
Input
    Joint probability distribution: $p(x,y)$
    Number of clusters: $m \in \{1..|X|\}$
Output
    A partition of $X$ into $m$ clusters
Computation
    $\widehat{X} \leftarrow X$
    $\forall i \in \{1..|X|-1\}$.compute($\delta I_{\hat{x}}(i)$) (see Eq. (9))
    while $|\widehat{X}| > m$ do
        $k \leftarrow \min_i(\delta I_{\hat{x}}(i))$
        $\hat{x} \leftarrow$ merge($x_k, x_{k+1}$)
        $\widehat{X} \leftarrow (\widehat{X} - \{x_k, x_{k+1}\}) \bigcup \{\hat{x}\}$
        Update $\delta I_{\hat{x}}$ for the neighbors of $\hat{x}$
    end while
    return $\widehat{X}$
```

Fig. 5. Bottom-up bottleneck algorithm.

$H(B) = I(B,\widehat{R}) + H(B|\widehat{R})$, where $I(B,\widehat{R})$ and $H(B|\widehat{R})$ represent, respectively, the successive values of MI and conditional entropy obtained after the successive mergings. Remember that $H(B)$ remains constant. Note also that $H(B|\widehat{R})$ is the average entropy of the regions, given by

$$\begin{aligned}H(B|\widehat{R}) &= -\sum_{r\in\mathcal{R}} p(r) \sum_{b\in\mathcal{B}} p(b|r)\log p(b|r)\\ &= -\sum_{r\in\mathcal{R}} p(r) H(B|r) \qquad (15)\end{aligned}$$

where $H(B|r)$ is the entropy of the normalized histogram of region r. If two regions are clustered

$$\delta I_{\hat{r}} = I(R,B) - I(\widehat{R},B) = H(B|\widehat{R}) - H(B|R). \qquad (16)$$

Thus, $H(B|\widehat{R})$ never decreases at any iteration due to the mixing of the histogram regions.

In Fig. 6, we show the results of merging the regions of the images of Fig. 3(a) and 3(b) obtained from the splitting phase with a $MIR_{\mathrm{r}} = 0.8$ in the BSP partition. For both images, the results

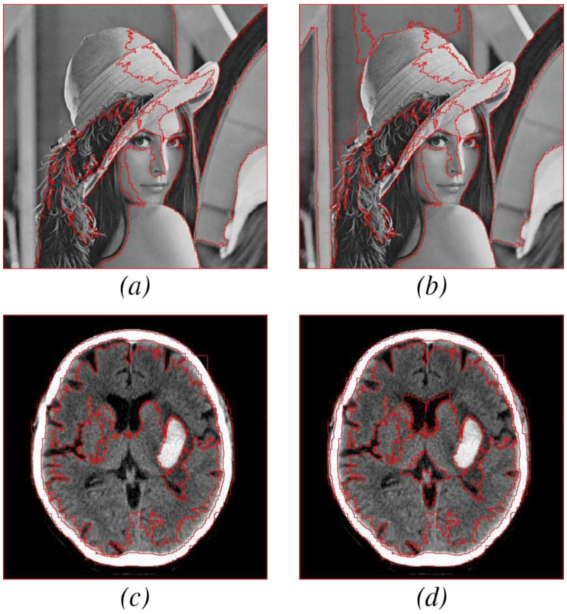

Fig. 6. Segmentation results of the split-and-merge algorithm for the Lena image [Fig. 3(a)] and hematoma image [Fig. 3(a)], where R represents the final number of regions of each image. (a) $R = 6$; (b) $R = 10$; (c) $R = 6$; (d)$R = 10$.

with six and ten different regions are shown. Observe that in this case the main structures of the image are separated, specially for the hematoma image, where the lesion, the skull, and internal brain structures, as the ventricles, are correctly identified. In the Lena image the main structures of the images are identified, but the illumination problem over the same object is not solved at all by the method. For instance, observe the incorrect segmentation of the hat. This is due to the fact that the method only deals with local intensities and not with other image features such as gradient or texture.

To evaluate our method, we compare it with a manual segmentation and the normalized cuts segmentation presented in [19]. For our experiments we use the 100 test images from the Berkeley database [19], which have been manually segmented. To compare the different segmentation results, we apply the LCE and the GCE error metrics proposed in [19]. These measures are defined as

$$LCE(S_1, S_2) = \frac{1}{n}\sum_i \min\left\{E(S_1, S_2, p_i), E(S_2, S_1, p_i)\right\} \tag{17}$$

and

$$GCE(S_1, S_2) = \frac{1}{n}\min\left\{\sum_i E(S_1, S_2, p_i), \sum_i E(S_2, S_1, p_i)\right\} \tag{18}$$

where

$$E(S_1, S_2, p_i) = \frac{\|R(S_1, p_i) \setminus R(S_2, p_i)\|}{\|R(S_1, p_i)\|}. \tag{19}$$

Here, the $R(S, p_i)$ represents the set of pixels corresponding to the region in segmentation S that contains pixel p_i, the symbol $\setminus$ denotes the set difference, and $\|x\|$ is the cardinality of the set x. These measures are tolerant to refinements, and, therefore, the importance of the level of detail of the segmentation has not high relevance.

TABLE I
Overall Segmentation Error for Humans, Split-and-Merge Algorithm, and NCuts for Both Same-Image Segmentation Pairs and Different-Image Segmentation Pairs. The NCuts Results are Obtained From [19], Computed in a Subset of the Database

	LCE		GCE	
	Same	Different	Same	Different
Humans	0.053	0.283	0.083	0.357
Split-and-merge	0.203	0.341	0.272	0.397
NCuts (from [19])	0.22	0.31	0.28	0.38

Since the computation of LCE and GCE requires a segmentation pair, we evaluate: 1) a pair of manual segmentations, and 2) a manual segmentation versus our method segmentation. The obtained results are shown in Table I, where each row corresponds to the mean distance value of each one of the evaluated situations. We compute each measure considering different segmentations of the same image and different segmentations of different images. In all the cases, the automated results have been obtained with three different segmentations with the same number of manual segmentation regions. In addition, we report the results presented in [19] when applying the normalized cuts (NCuts) segmentation algorithm [24]. Note that the results of this algorithm have been obtained from an early stage of the database, with less images and manual segmentations.

Observe that the similarity between the segmentation obtained with the split-and-merge method and the manual segmentation of the same image are clearly higher than the one with these methods from different images. Our method gives an overall error of 20% by LCE (compared to 5% for humans), and 27% by GCE (compared to 8% for humans). Observe also that the obtained results are better than the ones provided by the NCuts segmentation algorithm.

In Fig. 7, we depict the results of applying the split-and-merge algorithm to four images of the Berkeley database [19], where a given number of regions has been predetermined for each image. Note how our split-and-merge algorithm detects very well the homogeneity of the textured regions [such as the field in Fig. 7(a), the skin of the zebra in Fig. 7(b), the baboon hair in Fig. 7(c), and the sand in Fig. 7(d)]. This good behavior is due to the fact that the decision of splitting (and merging) is based on the divergence between the region histograms. In particular, two regions with the same texture have similar probability density function, and, therefore, the JS-divergence between them is very low. In the splitting phase, a region with the same texture will not be partitioned because the gain of MI would be very low [see (12)]. On the other hand, in the merging phase, those regions will be merged because the loss of MI is very low [see (14)]. Thus, ideally, each region will display a unique texture and only unconnected regions may have the same texture.

IV. Histogram Clustering Algorithm

In this section, we present a greedy histogram clustering algorithm which takes as input a partitioned image and obtains a histogram clustering based on the minimization of the loss of MI. That is, we group the bins of the histogram so that the MI is

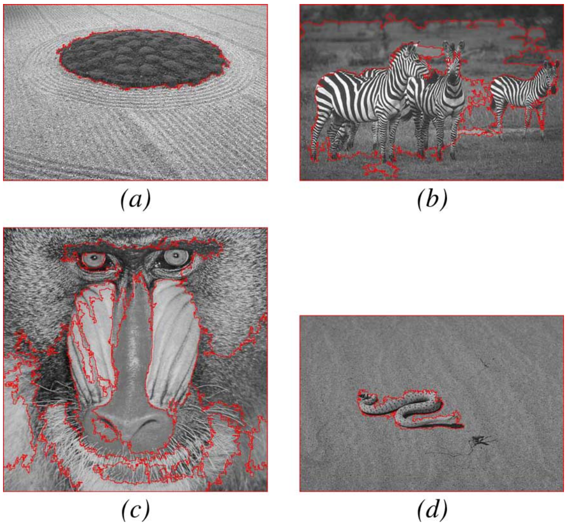

Fig. 7. Segmentation results of the split-and-merge algorithm for different images from the Berkeley database, where R represents the final number of regions of each image. (a) $R = 2$; (b) $R = 10$; (c) $R = 12$; (d) $R = 2$.

maximally preserved. From the perspective of the information bottleneck method, the binning process is controlled by a given partition of the image. This histogram clustering algorithm has been previously presented in [21].

Our clustering algorithm is based on the channel $B \rightarrow R$, which is a result of inverting the channel of the previous section. This channel is defined by a conditional probability matrix $p(R|B)$ which expresses how the pixels corresponding to each histogram bin are distributed into the regions of the image. Bayes' theorem, expressed by $p(b)p(r|b) = p(r)p(b|r)$, establishes the relationship between the conditional probabilities of both channels $B \rightarrow R$ and $R \rightarrow B$.

The basic idea underlying our histogram clustering algorithm is to capture the maximum information of the image with the minimum number of histogram bins. Analogous to the merging algorithm of the previous section, the loss of MI due to the clustering $\widehat{b}$ of two neighbor bins b_1 and b_2 is given by

$$\begin{aligned} \delta I_{\widehat{b}} &= I(B,R) - I(\widehat{B},R) \\ &= p(\widehat{b}) JS\left(\pi_1, \pi_2; p(R|b_1), p(R|b_2)\right) \end{aligned} \quad (20)$$

where $p(\widehat{b}) = p(b_1) + p(b_2)$, $\pi_1 = (p(b_1)/p(\widehat{b}))$, $\pi_2 = (p(b_2)/p(\widehat{b}))$, and $p(r|\widehat{b}) = (p(b_1)p(r|b_1)+p(b_2)p(r|b_2)/p(\widehat{b}))$. Thus, when two neighbor bins b_1 and b_2 are equally distributed in the regions of the image $(p(R|b_1) = p(R|b_2))$, their clustering results in $\delta I_{\widehat{b}} = 0$. In general, if two bins are very *similar* $(\delta I_{\widehat{b}} \approx 0)$, the channel can be simplified by substituting these two bins by their clustering, without a significant loss of information. Our algorithm proceeds by merging two neighbor bins so that the loss of MI is minimum (see Fig. 5). The stopping criterion is given by the ratio $MIR_{\mathrm{b}} = (I(\widehat{B},R)/I(B,R))$ or a predefined number of bins N_{b}.

Note that, during the clustering process $H(R) = H(R|\widehat{B}) + I(\widehat{B},R)$, where $H(R)$ is the entropy of $p(R)$, and $H(R|\widehat{B})$ and $I(\widehat{B},R)$ represent, respectively, the successive values of conditional entropy and MI obtained after the successive clusterings.

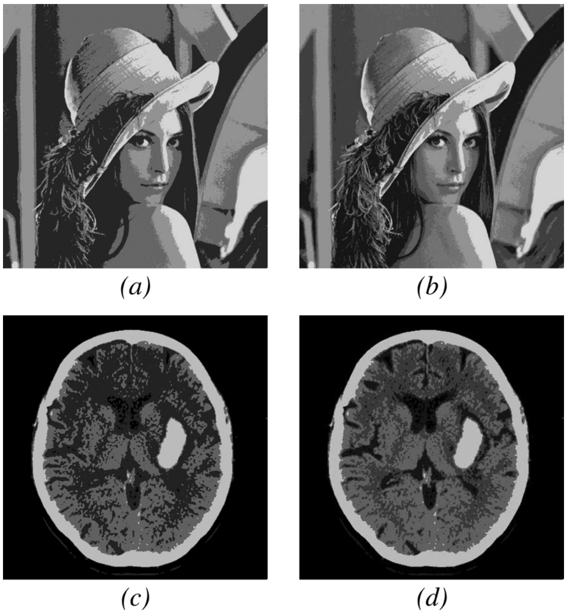

Fig. 8. Segmentation results of the histogram clustering algorithm for the Lena image [Fig. 3(a)] and hematoma image [Fig. 3(b)], where C represents the final number of intensity bins of each image. (a) $C = 4$; (b) $C = 6$; (c) $C = 4$; (d)$C = 6$.

Observe also that $H(R|\widehat{B})$ is the average entropy of the bins (i.e., a measure of the degree of dispersion of the bins in the set of regions) and increases (or remains constant) at each iteration.

In Fig. 8, we show the segmented images obtained from the partitions achieved with the split-and-merge algorithm with $MIR_{\mathrm{r}} = 0.8$ as stopping criterion of the splitting process and 100 regions for the merging one. For each image, the results obtained using 4 and 6 clusters are shown. For instance, observe how the internal structures of the brain are approximately preserved using only 6 clusters.

In Fig. 9, we plot LCE and GCE measures for the histogram clustering algorithm applied to the slice 80 of the T2 Brainweb image [see Fig. 11(ii.a)] considering different levels of noise. The number of clusters has been fixed to 6 and the experiment has been evaluated for different stopping criteria of the split-and-merge algorithm: while the MIR of the splitting phase has been set to 0.7 for all the cases, the number of regions of the merging phase takes the values 40, 60, 100, and 200. Observe that the best results are achieved for 60 regions in most of the cases. This is due to the fact that with a too high number of regions the spatial information is partially lost in the detail, while with a too low number of regions the spatial distribution is not much informative, not being able to capture any detail.

V. Registration-Based Segmentation

In this section, two histogram clustering algorithms based on the channel established between two registered images A and B are introduced. The main idea behind our algorithms is that the segmentation of image A is obtained by extracting the structures that are most relevant for image B. In this case, any previous segmentation is required. These histogram clustering algorithms have been introduced in [3].

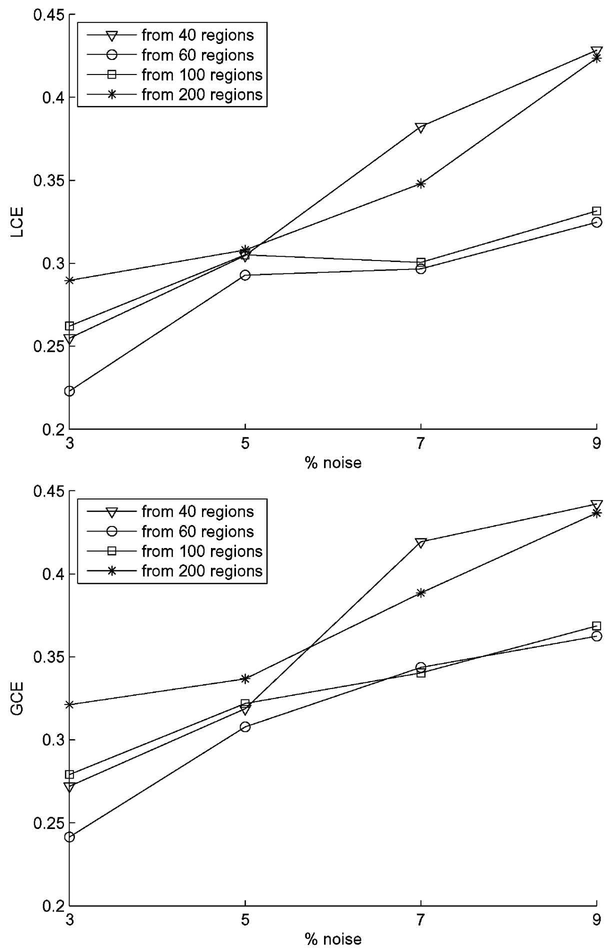

Fig. 9. Two plots representing the LCE and GCE measures of the segmentation results of the slice 80 of the T2 Brainweb image using six clusters for different levels of noise and from different level of image partition.

A. One-Sided Clustering Algorithm

We present a greedy hierarchical clustering algorithm that clusters the histogram bins of image A by minimizing the loss of MI between A and B. First of all, in a preprocessing step, images A and B have to be registered, establishing an information channel $X \rightarrow Y$, where X and Y denote, respectively, the histograms of A and B. From the data processing inequality (5) and the information bottleneck method (see Section II-C), we know that any clustering over X (for instance, merging neighbor histogram bins x_1 and x_2), denoted by $\widehat{X}$, will reduce $I(X,Y)$.

At the initial stage of our algorithm (see Fig. 5), only one intensity value is assigned to each histogram bin of X. Then, the algorithm proceeds greedily by merging two neighbor clusters so that the loss of MI is minimum. This procedure merges the two clusters which are more similar from the perspective of B. Note the constraint that only neighbor bins can be merged. The cardinality $|\widehat{X}|$ goes from $|X|$ to 1 in the extreme case.

The efficiency of this algorithm can be greatly improved if the reduction of MI due to the merging of bins x_1 and x_2 is computed by

$$\delta I_{\hat{x}} = p(\hat{x}) JS\left(\pi_1, \pi_2; p(Y|x_1), p(Y|x_2)\right) \qquad (21)$$

where $p(\hat{x}) = p(x_1) + p(x_2)$, $\pi_i = (p(x_1)/p(\hat{x}))$, $\pi_2 = (p(x_2)/p(\widehat{x}))$, and $p(Y|x_1)$ and $p(Y|x_2)$ denote, respectively, the corresponding rows of the conditional probability matrix of the information channel [25]. The evaluation of $\delta I_{\hat{x}}$ for each pair of clusters is done in $O(|Y|)$ operations and, at each iteration of the algorithm, it is only necessary to compute the $\delta I_{\hat{x}}$ of the new cluster with its two corresponding neighbors. All the other precomputed $\delta I_{\hat{x}}$ values remain unchanged [25].

```
Input
    Joint probability distribution: $p(x,y)$
    Number of clusters: $m \in \{1..|X|+|Y|\}$
Output
    A partition of $(X,Y)$ into $m$ clusters
Computation
    $(\widehat{X},\widehat{Y}) \leftarrow (X,Y)$
    $\forall i \in \{1..|X|-1\}$.compute($\delta I_{\hat{x}}(i)$) (see Eq. (9))
    $\forall j \in \{1..|Y|-1\}$.compute($\delta I_{\hat{y}}(j)$) (see Eq. (9))
    while $|\widehat{X}|+|\widehat{Y}| > m$ do
        $k \leftarrow \min_{i,j}(\delta I_{\hat{x}}(i), \delta I_{\hat{y}}(j))$
        if $k$ indexes $\widehat{X}$then
            associate $(Z,V)$ to $(\widehat{X},\widehat{Y})$
        else
            associate $(Z,V)$ to $(\widehat{Y},\widehat{X})$
        $\hat{z} \leftarrow$ merge($z_k, z_{k+1}$)
        $\widehat{Z} \leftarrow (Z - \{z_k, z_{k+1}\}) \bigcup \{\hat{z}\}$
        Update $\delta I_{\hat{z}}$ for the neighbors of $\hat{z}$
        Update all $\delta I_v$
    end while
    return $(\widehat{X},\widehat{Y})$
```

Fig. 10. Co-clustering algorithm.

Similar to the algorithms of Sections III and IV, clustering can be stopped using several criteria: a fixed number of clusters, a given ratio $MIR_{\rm b} = I(\widehat{X},Y)/I(X,Y)$, or a variation $\delta I_{\hat{x}}$ greater than a given ϵ. The $MIR_{\rm b}$ ratio is considered as a quality measure of the clustering.

B. Co-Clustering Algorithm

Let us now consider a simultaneous clustering of images A and B. Unlike the algorithm presented by Dhillon [9] for word-document clustering, which alternatively clusters the variables $\widehat{X}$ and $\widehat{Y}$, our algorithm (see Fig. 10) chooses at each step the best merging of one of the two images (i.e., the one that entails a minimum reduction of MI). The similarity between the two images is being symmetrically exploited. Thus, each clustering step benefits from the progressive simplification of the images. One of the main advantages of this algorithm is the great reduction of sparseness and noise of the joint probability matrix. As we will see with the experimental results, the simultaneous merging over the images A and B obtain better results than with the one-sided algorithm.

From the data processing inequality (5), $I(\widehat{X},\widehat{Y})$ is a decreasing function with respect to the reduction of the total number of clusters $|\widehat{X}|+|\widehat{Y}|$. Thus, $I(\widehat{X},\widehat{Y}) \leq I(X,Y)$. Like the one-sided algorithm, the stopping criterion can be given by a predefined number of bins, a given ratio $MIR = I(\widehat{X},\widehat{Y})/I(X,Y)$ or a variation $\delta I_{\widehat{x}}$ (or $\delta I_{\widehat{y}}$) greater than a given ϵ. Similarly to the above one-sided algorithm, the reduction of MI can be computed from the JS-divergence (21). But in the co-clustering algorithm, for each clustering of $\widehat{X}$ (or

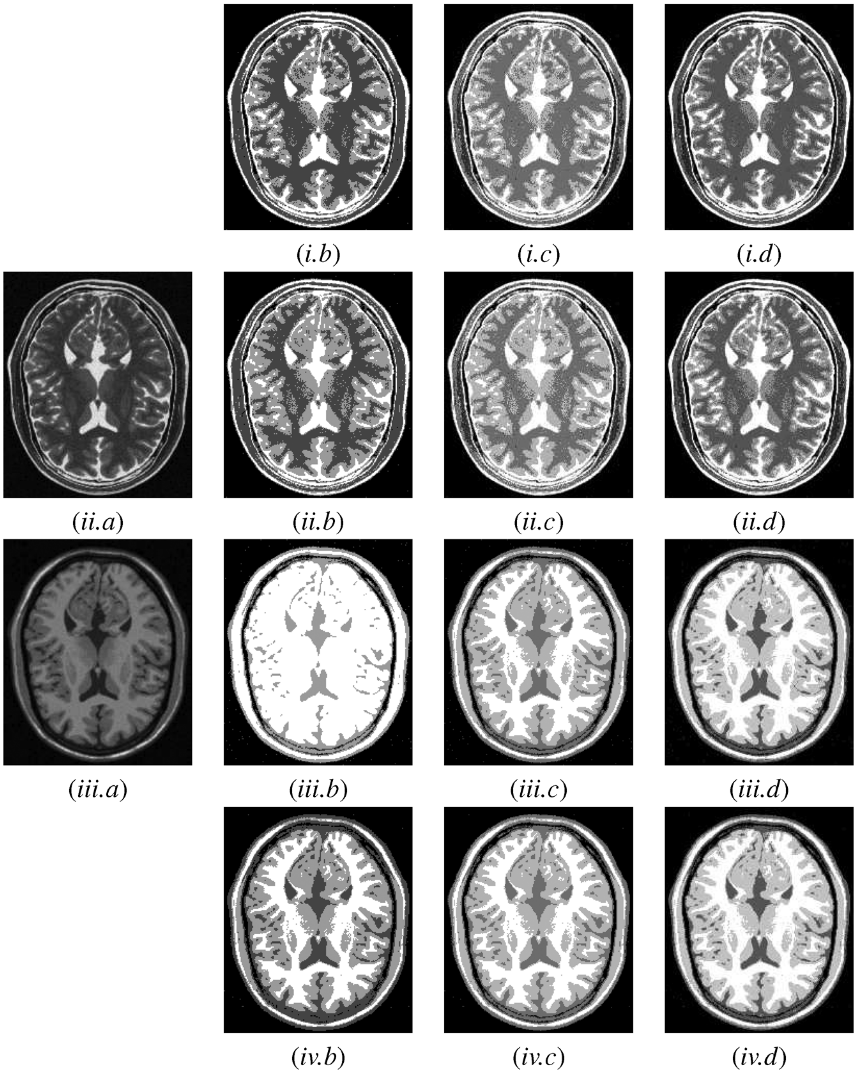

Fig. 11. (a) Original images from the Brainweb database with 3% of noise. (b)–(d) Images segmented using 4, 5, and 6 bins, respectively. (i,iv) Images obtained with the one-sided algorithm. (ii,iii) Images obtained with the co-clustering algorithm.

$\widehat{Y}$), it is necessary to recompute all the $\delta I_{\widehat{y}}$ (or $\delta I_{\widehat{x}}$). Fig. 10 shows the co-clustering algorithm where the stopping criterion is given by the total number of clusters.

C. Results

To evaluate the performance of the two registration-based segmentation algorithms, we have used both synthetic and real images. The first test images are a set of synthetic magnetic resonance T1 (MR-T1) and T2 (MR-T2) image modalities from the Brainweb database [7]. These images are obtained synthetically from a phantom and they can be generated with different levels of image noise. These two image modalities are acquired exactly in the same spatial position, and, therefore, the preprocessing registration step is not required. The second test images are real data from a patient from the Vanderbilt database [20]. This dataset is composed of MR and CT image modalities. The resolution of the MR and CT is $256 \times 256 \times 26$ and $512 \times 512 \times 28$, respectively. These MR and CT images have been registered using the NMI measure [27].

Fig. 11 shows the results of the proposed one-sided and co-clustering algorithms for the MR-T1 and MR-T2 Brainweb 3-D images with a 3% of noise. These images are simulated from a synthetic atlas and they are perfectly registered since the same process is applied to achieve both images. The original MR-T2 and MR-T1 images are depicted in Fig. 11(ii.a) and (iii.a), respectively. Columns Fig. 11(b)–(d) show the segmented images with 4, 5, and 6 clusters, respectively. The results obtained with the one-sided algorithm applied on the MR-T1 and MR-T2 images are shown in Fig. 11(i.b)–(d) and (iv.b)–(d), respectively. The results obtained with the co-clustering algorithm are shown for the MR-T2 image in Fig. 11(ii.b)–(d) and for the MR-T1 in Fig. 11(iii.b)–(d).

Observe the good segmentation results achieved with both methods for the MR-T2 image. For both methods, the images obtained with only four clusters distinguish between background (black), white matter (dark gray), gray matter (light gray), and ventricles and cerebral fluids (white), which are the main structures of brain anatomy. The results are similar for the MR-T1 image and the one-sided algorithm, but they

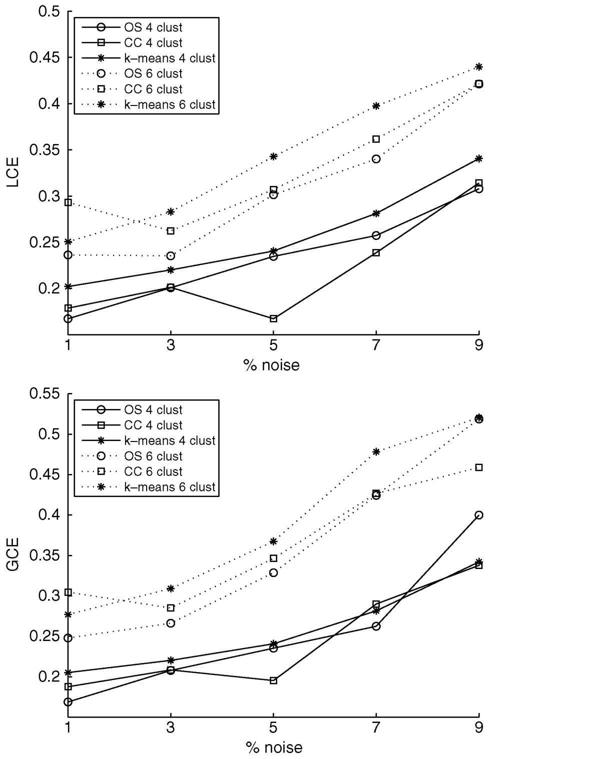

Fig. 12. Two plots representing the LCE and GCE measures of the segmentation results of the T2 Brainweb image for different levels of noise. The three curves represent the one-sided and co-clustering results with the T1 image with 3% of noise as a control variable and the k-means algorithm results.

are not so satisfactory for the co-clustering one. In this case, the background is split into two clusters while gray and white matter are considered in the same cluster. This might be due to the higher background probability in comparison with any other region of the image. This undesired behavior disappears when 5 or 6 clusters are considered.

In Fig. 12, we plot LCE and GCE measures for the co-clustering (represented with squares), the one-sided (represented with circles), and the k-means [15] (represented with stars) algorithms applied to the T2 Brainweb image for different levels of noise. For each algorithm, we evaluate two different number of clusters: 4 (represented as continuous lines) and 6 (represented as dotted lines). For the co-clustering and the one-sided algorithms, the T1 image with 3% of noise has been used as a control variable. Since the tested images have been obtained from a phantom, we use this phantom as a ground truth in order to compute the LCE and GCE measures.

Note that for four clusters, the behavior of the three algorithms is similar, even though the iterative structure of the k-means leads to the optimal solution and the greedy structure of our algorithms does not. Despite this, the results of the proposed algorithms are slightly better for the LCE measure. For 6 clusters, the proposed methods achieve better results than the k-means, specially when the noise increases. In these cases, the control variable, which is not influenced by this noise, helps to improve the segmentation.

With the next experiment we want to simulate the case where one image of low quality is segmented considering a high quality image, similar than the preoperative and intraoperative images. In order to study this situation, we have considered the MR-T1 Brainweb image with 1% of noise to be a high quality image and a MR-T2 Brainweb image with 7% of noise to be a low quality image. In this situation only the one-sided algorithm is considered, taking as a control variable the high quality image [MR-T1, Fig. 13(a)] and segmenting the low quality image [MR-T2, Fig. 13(b)]. The results of the one-sided algorithm with 4, 5 and 6 clusters are plotted in Fig. 13(c)–(e), respectively.

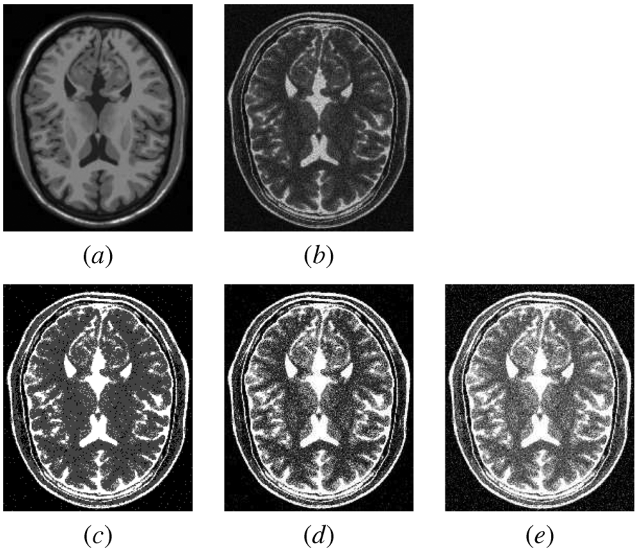

Fig. 13. (a) Original control image MR-T1 with 1% of noise. (b) Original image MR-T2 with 7% of noise. (c)–(e) Results of segmenting (b) using 4, 5, and 6 bins, respectively.

As we can observe in these images, in spite of the low quality of the original one, the segmentation results try to separate correctly the main parts of the brain image: background, ventricles, white matter and gray matter. This is because the control variable of the segmentation method is very accurate and tries to achieve the maximum relationship between the input image and the resulting segmentation.

In Fig. 14, we show the results obtained with the one-sided and co-clustering algorithms applied on the CT [Fig. 14(ii.a)] and MR [Fig. 14(iii.a)] original image of the Vanderbilt dataset. The composition of Fig. 14 is similar to the one in Fig. 11. Columns *b–d* show the segmented images with 2, 4, and 6 clusters, respectively. The results obtained with the one-sided algorithm applied on the CT and MR images are shown in Fig. 14(i.b)–(d) and Fig. 14(iv.b)–(d), respectively. The results obtained with the co-clustering algorithm are shown for the CT image in Fig. 14(ii.b)–(d) and for the MR in Fig. 14(iii.b)–(d).

If we compare the original unsegmented images with the resulting segmented images, we can see that the best results are obtained with the co-clustering algorithm [Fig. 14(ii–iii.b)–(d)]. There is clear evidence that hidden structures of the image are more precisely recovered. Compare, for instance, the images for an equal number of clusters of Fig. 14(i.c) and (ii.c). This better behavior can be explained because in the co-clustering case we make use of all bidirectional information obtained with the progressive simplification of both images. For both algorithms, results appear much better when segmenting the CT images than

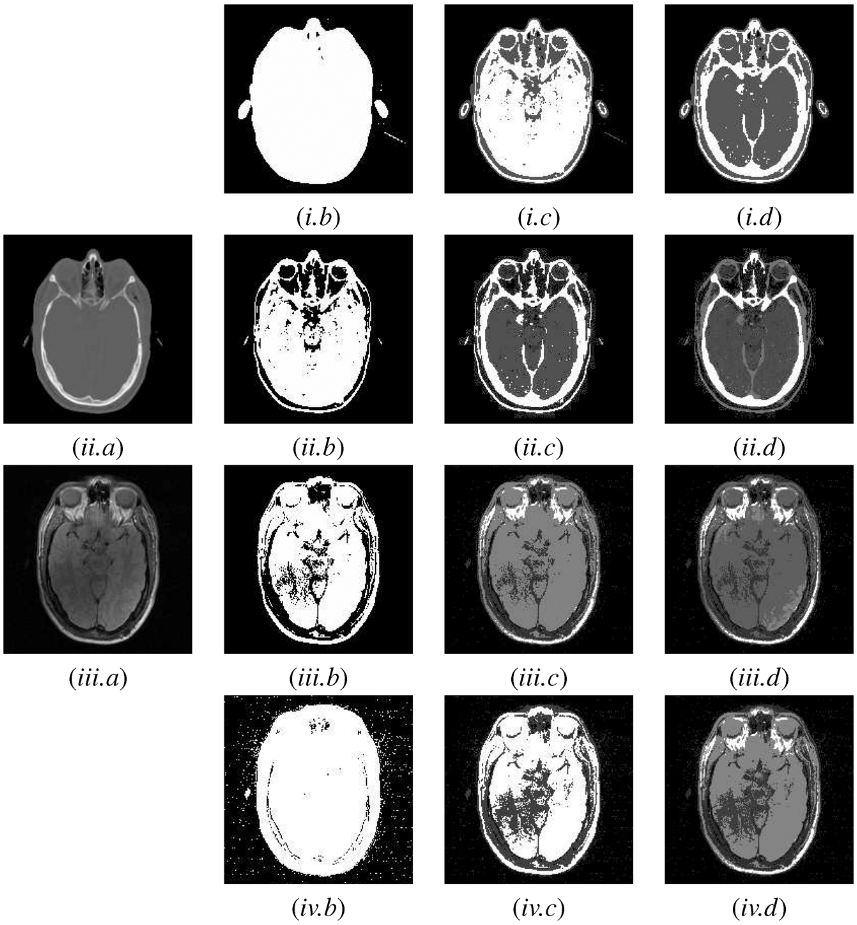

Fig. 14. (a) Original dataset images. (b)–(d) Images segmented using 2, 4, and 6 bins, respectively. (i,iv) Images obtained with the one-sided algorithm. (ii,iii) Images obtained with the co-clustering algorithm.

the MR ones. This is due to the fact that the segmentation of the CT images benefits a lot from the precise information contained in the MR histogram.

VI. Conclusion

We have presented a general framework for image segmentation based on a hard version of the information bottleneck method. Three different segmentation algorithms have been introduced: a split-and-merge, a histogram clustering and a registration-based clustering. For the two first algorithms, an information channel between the regions of the image and the histogram bins has been defined. Based on the preservation of mutual information, the spatial distribution and the histogram bins are maximally correlated. For the third algorithm, a channel between two multimodal images is defined, allowing to segment one image preserving the maximum information given by the other one. The main advantages of these methods are that do not assume any *a priori* information about the images (e.g., intensity probability distribution) and that take into account the spatial distribution of the samples. Different experiments on both natural and medical images and comparisons with standard methods have shown the good behavior of the proposed algorithms.

Further investigation on stopping criteria is needed to determine the optimal number of both regions and clusters. On the other hand, new segmentation channels could be tested, taking into account other kind of information, such as color, texture, or gradient. We also plan to explore the application of these methods to image fusion and level-of-detail applications.

References

[1] R. H. Y. Chung, N. H. C. Yung, and P. Y. S. Cheung, "An efficient parameterless quadrilateral-based image segmentation method," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 9, pp. 1446–1458, Sep. 2005.

[2] D. H. Ballard and C. M. Brown*, Computer Vision*. Englewood Cliffs, NJ: Prentice-Hall, 1982.

[3] A. Bardera, M. Feixas, I. Boada, J. Rigau, and M. Sbert, "Registration-based segmentation using the information bottleneck method," in *Proc. Iberian Conf. Patern Recognition and Image Analisys*, June , vol. II, pp. 190–197.

[4] P. Bernaola, J. L. Oliver, and R. Román, "Decomposition of DNA sequence complexity," *Phys. Rev. Lett.*, vol. 83, no. 16, pp. 3336–3339, Oct. 1999.

[5] J. Burbea and C. R. Rao, "On the convexity of some divergence measures based on entropy functions," *IEEE Trans. Inf. Theory*, vol. 28, no. 3, pp. 489–495, May 1982.

[6] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, no. 6, pp. 679–698, Jun. 1986.

[7] C. Cocosco, V. Kollokian, R.-S. Kwan, and A. Evans, "Brainweb: Online interface to a 3D MRI simulated brain database," *NeuroImage*, vol. 5, no. 4, 1997.

[8] T. M. Cover and J. A. Thomas*, Elements of Information Theory*, ser. Wiley Series in Telecommunications. New York: Wiley, 1991.

[9] I. S. Dhillon, S. Mallela, and D. S. Modha, "Information-theoretic co-clustering," in *Proc. 9th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, New York, 2003, pp. 89–98.

[10] D. A. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*. Upper Saddle River, NJ: Prentice-Hall, 2003.

[11] J. Freixenet, X. Muñoz, D. Raba, J. Martí, and X. Cufí, "Yet another survey on image segmentation: Region and boundary information integration," in *Proc. Eur. Conf. Computer Vision*, Copenhagen, Denmark, May 2002, pp. 408–422.

[12] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and the bayesian restoration of images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 6, no. 6, pp. 721–741, Jun. 1984.

[13] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Upper Saddle River, NJ: Prentice-Hall, 2002.

[14] N. Grira, M. Crucianu, and N. Boujemaa, Unsupervised and Semi-Supervised Clustering: A Brief Survey Réseau d'Excellence MUSCLE (6ePCRD), July 2004, Tech. Rep.

[15] J. A. Hartigan and M. A. Wong, "A k-means clustering algorithm," *Appl. Statist.*, vol. 28, no. 1, pp. 100–108, 1979.

[16] S. L. Horowitz and T. Pavlidis, "Picture segmentation by a tree traversal algorithm," *J. ACM*, vol. 23, no. 2, pp. 368–388, 1976.

[17] S. R. Kulkarni, G. Lugosi, and S. S. Venkatesh, "Learning pattern classification—A survey," *IEEE Trans. Inf. Theory*, vol. 44, no. 6, pp. 2178–2206, Jun. 1998.

[18] S. Li, *Markov Random Field Modeling in Image Analysis*. New York: Springer, 2001.

[19] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. 8th Int. Conf. Computer Vision*, Jul. 2001, vol. 2, pp. 416–423.

[20] National Institutes of Health, Retrospective Image Registration Evaluation Vanderbilt University. Nashville, TN, USA, 2003, project number 8R01EB002124-03, Principal Investigator J. M. Fitzpatrick [Online]. Available: http://www.vuse.vanderbilt.edu/~image/registration/

[21] J. Rigau, M. Feixas, and M. Sbert, "An information theoretic framework for image segmentation," presented at the IEEE Int. Conf. Image Processing, Singapore, Oct. 2004.

[22] I. K. Sethi and G. Sarvarayudu, "Hierarchical classifier design using mutual information," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 4, no. 4, pp. 441–445, Jul. 1982.

[23] M. Sezgin and B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," *J. Electron. Imag.*, vol. 13, no. 1, pp. 146–168, Jan. 2004.

[24] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

[25] N. Slonim and N. Tishby, "Agglomerative information bottleneck," in *Proc. NIPS-12*, 2000, pp. 617–623, MIT Press.

[26] N. Slonim and N. Tishby, "Document clustering using word clusters via the information bottleneck method," in *Proc. 23rd Annu. Int. ACM SIGIR Conf. Research and Development in Information Retrieval*, 2000, pp. 208–215, ACM Press.

[27] C. Studholme, "Measures of 3D Medical Image Alignment," Ph.D. dissertation, Univ. London, London, U.K., Aug. 1997.

[28] J. Suri, K. Setarehdan, and S. Singh, *Advanced Algorithmic Approaches to Medical Image Segmentation*. London, U.K.: Springer-Verlag, 2002, Inst. Computer Graphics, Vienna University of Technology, Austria.

[29] N. Tishby, F. C. Pereira, and W. Bialek, "The information bottleneck method," in *Proc. 37th Annu. Allerton Conf. Communication, Control and Computing*, 1999, pp. 368–377.

[30] Y.-T. Wu, F. Y. Shih, J. Shi, and Y.-T. Wu, "A top-down region dividing approach for image segmentation," *Pattern Recognit.*, vol. 41, no. 6, pp. 1948–1960, 2008.

Anton Bardera received the M.Sc. degree in telecommunications engineering in 2002 from the Universitat Politècnica de Catalunya, Spain, and the Ph.D. degree from the University of Girona, Spain, in 2008.

Currently, he is an Assistant Professor in the Department of Informàtica i Matemàtica Aplicada, University of Girona. The main fields of his research interests are image processing, information theory, and biomedical applications.

Jaume Rigau received the M.Sc. degree in computer science and the Ph.D. degree from the Technical University of Catalonia, Spain.

He is an Associate Professor of computer science at the University of Girona, Spain. His research is based on the application of information theory to computer graphics and image processing.

Imma Boada received the M.Sc. degree in computer science from the Universitat Autònoma de Barcelona, Spain, in 1992, and the Ph.D. degree from the Universitat Politècnica de Catalunya in 2001.

Since 1992, she has been with the University of Girona, Spain, where she is Associate Professor of computer science. Her research interests focus on scientific visualization applied to medicine.

Miquel Feixas received the M.Sc. degree in theoretical physics from the Universitat Autònoma de Barcelona, Spain, in 1979, and the Ph.D. degree in computer science from the Universitat Politècnica de Catalunya, Spain, in 2002.

He is an Associate Professor of computer science at the University of Girona, Spain. His research is focused on the application of information theory techniques to radiosity, global illumination, viewpoint selection, and image processing.

Mateu Sbert received the Ph.D. degree in computer science from the Technical University of Catalonia, Spain.

He is a Professor of computer science at the University of Girona, Spain. His research interests include the application of Monte Carlo and information theory to computer graphics and image processing.