Regional Appearance Modeling based on the Clustering of Intensity Profiles

François Chung, Hervé Delingette

▶ **To cite this version:**

François Chung, Hervé Delingette. Regional Appearance Modeling based on the Clustering of Intensity Profiles. Computer Vision and Image Understanding, 2013, 117 (6), pp.705-717. 10.1016/j.cviu.2013.01.011 . hal-00813880

HAL Id: hal-00813880

https://inria.hal.science/hal-00813880

Submitted on 17 Apr 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Regional Appearance Modeling based on the Clustering of Intensity Profiles

François Chung[a], Hervé Delingette[a,]

[a]*INRIA Sophia-Antipolis, Asclepios Team, 06902 Sophia-Antipolis, France*

Abstract

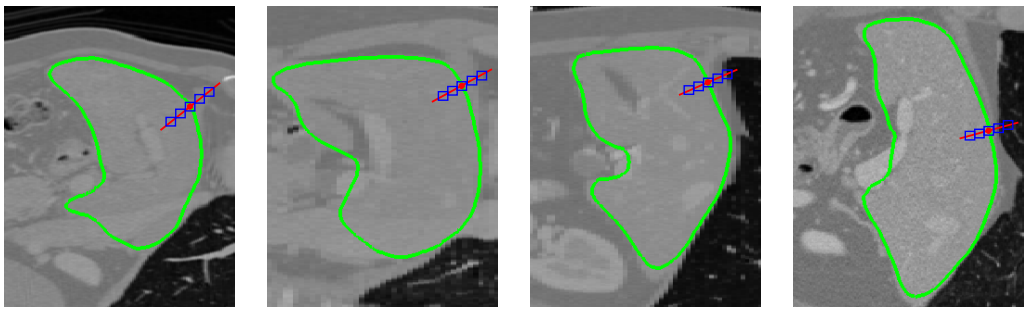
Model-based image segmentation is a popular approach for the segmentation of anatomical structures from medical images because it includes prior knowledge about the shape and appearance of structures of interest. This paper focuses on the formulation of a novel appearance prior that can cope with large variability between subjects, for instance due to the presence of pathologies. Instead of relying on Principal Component Analysis such as in Statistical Appearance Models, our approach relies on a multimodal intensity profile atlas from which a point may be assigned to several profile modes consisting of a mean profile and its covariance matrix. These profile modes are first estimated without any intra-subject registration through a boosted EM classification based on spectral clustering. Then, they are projected on a reference mesh whose role is to store the appearance information in a common geometric representation. We show that this prior leads to better performance than the classical monomodal Principal Component Analysis approach while relying on fewer profile modes.

Keywords: appearance modeling, unsupervised clustering, model-based image segmentation, medical imaging.

1. Introduction

Extracting anatomical structures of interest (*e.g.* organs, bones and tissues) from medical images is still an important topic in the medical imaging community. This process, known as *segmentation*, may be performed in several ways. Using a prior information about the shape, explicit model-based image segmentation has been extensively used in the literature. A mesh is deformed by means of both *internal* and *external* forces. Internal forces aim at ensuring a smooth deformation of the mesh during segmentation while external forces attract the mesh to image-based features. In the literature, such features have been widely based on image boundaries (*i.e.* local features) or image regions (*i.e.* global features). A review on appearance models is available in [27].

For instance, statistical models of appearance such as Active Shape Models (ASM) are widely used [16]. Intensity profiles are trained and both mean profile and principal modes of variation for each landmark are extracted using Principal Component Analysis (PCA). Gabor wavelets have been successfully used in face recognition and tracking applications [18]. Other local feature-based approaches worth mentioning are kNN-classifier and boosting methods [29]. The most popular global feature-based approach are the Active Appearance Models (AAM), where all intensities from the inner region of the mesh are used to create a large feature vector. Like in Active Shape Models, a PCA is built, though on textures this time, and both mean of textures and modes of variation are extracted. To overcome computational time or memory issues that may arise due to the feature vector size, only parts of the inner region may be used around specific landmarks, *e.g.* bounding



(a) (b) (c) (d)

Figure 1: Sampling of an intensity profile (blue squares) at the normal (red line) of the same vertex (red circle) from a reference mesh (green delineation) registered to four liver images {(a),(b),(c),(d)}. The reference mesh features $N = 11760$ vertices and has been registered to the four liver images to ensure point correspondence.

boxes [17].

PCA-based appearance model (PCAP). To model the appearance around structures of interest, typical PCA-based methods consist in computing the Gaussian distribution (*i.e.* mean $\boldsymbol{\mu}_i$ and covariance matrix $\boldsymbol{\Sigma}_i$) of profiles $\mathbf{x}_i$ at each vertex $\mathbf{p}_i$ of a reference mesh registered on all datasets, *i.e.* point correspondence must be ensured (see an illustration in Figure 1). As a similarity measure, Mahalanobis distance d_M may be computed as $d_M(\mathbf{y}) = \sqrt{(\mathbf{y}-\boldsymbol{\mu}_i)^T\boldsymbol{\Sigma}_i^{-1}(\mathbf{y}-\boldsymbol{\mu}_i)}$, where $\mathbf{y}$ is the current profile sampled during local search. A regularization of the covariance matrix $\boldsymbol{\Sigma}_i$ is usually necessary to avoid any singularities due to its inversion [42]. A solution consists in using an alternative approach proposed by Cootes [15], in which the Mahalanobis distance d_M is computed by relying on the eigenvectors associated with the largest eigenvalues of the covariance matrix.

Given a PCA performed on N profiles $\mathbf{x}_i$, expressed by the principal

matrix Φ_i, the m eigenvalues λ_{il} and the mean intensity profile $\boldsymbol{\mu}_i$, truncated Mahalanobis distance $d_{M'}$ is defined as $d_{M'}(\mathbf{y}) = \sum_{l=1}^{m} \sqrt{(b_{il}^2/\lambda_{il})}$, where $b_i = (b_{i1}, \ldots, b_{im})^T$ is the model parameter vector of the best fit $\hat{\mathbf{y}}$ of $\mathbf{y}$ given the PCA model $b_i = \Phi_i^T(\hat{\mathbf{y}} - \boldsymbol{\mu}_i)$. Using PCA enables to reduce the number of profile dimensions by working on a subspace that still retains most of the variability observed with the profiles. In practice, only top eigenvectors from an eigen decomposition performed on covariance matrix $\boldsymbol{\Sigma}_i$ are selected to reduce its complexity. For that, a ratio on normalized eigenvalues $\mathcal{R}$ defined in [0,1] is specified. Eigenvalues above this ratio $\mathcal{R}$ are kept, otherwise they are discarded. In the literature, external forces based on intensity profiles are usually computed using PCA with explicit deformable models [23, 43]. In addition to the Mahalanobis distance, other similarity measures have been proposed to perform the local search [36, 9, 38, 40, 39].

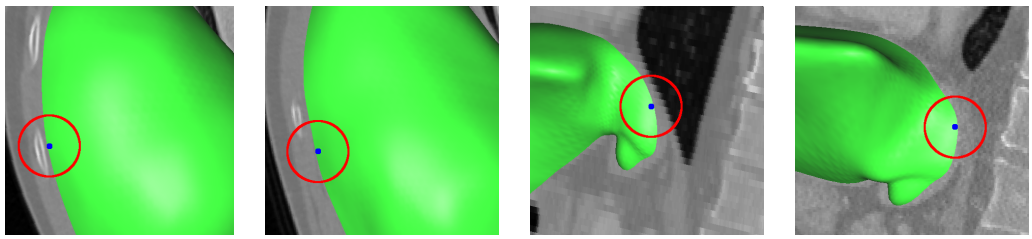
There are two main drawbacks with appearance models based on PCA, such as AAM and ASM. First, they require an accurate point-wise registration as the statistical analysis of appearance is performed at each point. Defining homologous points for 3D structures is difficult and therefore registering those points accurately is still considered challenging [27]. To cope with this difficulty, several authors have proposed to combine PCA shape priors with appearance force that do not rely on a training set. For instance, de Bruijne *et al.* [8] use information for adjacent slices to segment CTA images, while van Assen *et al.* [2] use fuzzy inference for feature detection.

Second, they are *monomodal*, *i.e.* they rely on the hypothesis that the probability density function is well described by one single Gaussian distribution. In practice, this means that, during local search, the current profile

is compared with only one prior profile. This hypothesis is often violated because shape is not necessarily fully correlated with appearance (see an illustration in Figure 2). Indeed, when comparing two liver images, points that can be matched based on shape similarity may not match based on intensity profile similarity, and conversely. The lack of correspondence between shape and appearance may also be amplified by the presence of pathological structures (*e.g.* liver tumors). For example, in the presence of a tumor signal at the border of the structure of interest, classical approaches assuming a single intensity profile distribution per landmark would lead to either Gaussian distribution priors with biased mean and increased variance (if such information is present in the training stage) or false border detection (if present in the segmentation stage). Instead of having one mode with large covariance, it is preferable for image segmentation or image detection to have several modes with lower covariance, in an attempt to reduce the number of outlier profiles but also to limit overfitting.

Multimodal Prior Appearance Model (MPAM). To overcome those limitations, we propose to estimate the intensity profile classes for each mesh and not for each point (*i.e.* without the need for any registration). Registration between subjects is only used to estimate the posterior probabilities on a reference mesh. This profile classification leads to the creation of *multimodal* priors, which may be then used during local search to compare the current profile with several prior profiles.

To do so, we proposed an alternative to PCA-based appearance models that is based on a multimodal prior [11]. This appearance prior, denoted as *Multimodal Prior Appearance Model (MPAM)*, is built upon an EM clustering



(a) (b) (c) (d)

Figure 2: Two examples illustrating that shape is not necessarily correlated with appearance: {(a),(b)} at rib level, and {(c),(d)} at lung level. In each example, a corresponding vertex (blue point) is displayed to compare the neighboring appearance. From {(a),(b)}, note how the rib close to the corresponding vertex in liver (a) is shifted in liver (b). From {(c),(d)}, note how the lungs close to the corresponding vertex in liver (c) are shifted in liver (d). The same reference mesh featuring $N = 11760$ vertices has been registered to the four liver images {(a),(b),(c),(d)} to ensure point correspondence.

of intensity profiles with model order selection to automatically select the number of profile classes, or *modes*. With the MPAM, each point may be associated with several profile modes and each profile mode is estimated on each subject and not across subjects. All mesh instances are then projected on a common reference framework.

We qualify this clustering as *regional* because it aims at finding clusters representing the appearance of a region, or a zone of a structure. Note that the term *regional* does not refer to the inner region enclosed by the structure of interest, as it is the case with AAM. In this paper, this *regional* approach is compared with the classical ASM approaches that rely on intensity profiles at landmarks, and which may therefore be considered as local appearance methods.

However, this proposed method [11] only concentrated on the presenta-

tion and the creation of the MPAM as a novel appearance prior, not on its optimization for a segmentation procedure. This is why we propose in this paper to classify the profiles using spectral clustering, whose aim is to reduce the high dimensional space featured by the profiles. This dimensionality reduction resolves common issues from the EM algorithm and has proven in our experiments to be robust with the change of mesh resolution (*i.e.* the number of profiles). In a second step, the profile classification is optimized in an iterative way using a boosting procedure. The objective is to make sure that the MPAM is suitable for segmentation purposes.

As the title of this article indicates, the overall objective of this article is to focus on the description of our novel appearance model optimized for segmentation purposes, then to illustrate the performance of the generated external forces using a public database. Therefore, when comparing the performance of our MPAM with PCAP, focus will be kept on these external forces while internal forces will be generated from a very basic shape prior.

2. Clustering of intensity profiles

2.1. Overview

In Figure 3, we overview the consecutive steps our method requires to build the MPAM. The input is a set of P meshes $\mathcal{M}_p$ corresponding to the segmentation of the same structure whose appearance needs to be captured in different images. The meshes may have different number of vertices, or even different topologies. Unlike PCA-based approaches, our method does not require accurate pointwise correspondences. At each vertex $\mathbf{p}_i$ of $\mathcal{M}_p$, d regularly sampled intensities are extracted to build a d-dimensional intensity

profile along the normal direction, noted $\mathbf{x}_i^p$, that may extend inward, outward or in both sides. This part may be seen as the first step of the training, where the method is capturing raw appearance information throughout the datasets.

Then, we propose to automatically cluster intensity profiles from each mesh using spectral clustering, as we do not want to make any assumption about the profile classification. The number of classes (*i.e.* the number of appearance regions), a hyperparameter, is selected in an automatic fashion through an heuristic. To efficiently use the clustering of intensity profiles associated with the MPAM as an appearance prior for segmentation purposes, we propose the boosted clustering. The objective is to perform a clustering under a localization constraint, in the idea that a profile mode should be able to describe in a discriminative way the boundary between the inside and the outside of the structure. To build a MPAM out of P datasets, all P meshes are registered to the same reference mesh $\mathcal{M}^\star$. The objective is to project the classifications from P meshes into a reference mesh and to estimate the posterior probabilities associated with the MPAM. Each vertex $\mathbf{p}_i$ of $\mathcal{M}^\star$ is given a probability to belong to several intensity profile classes.

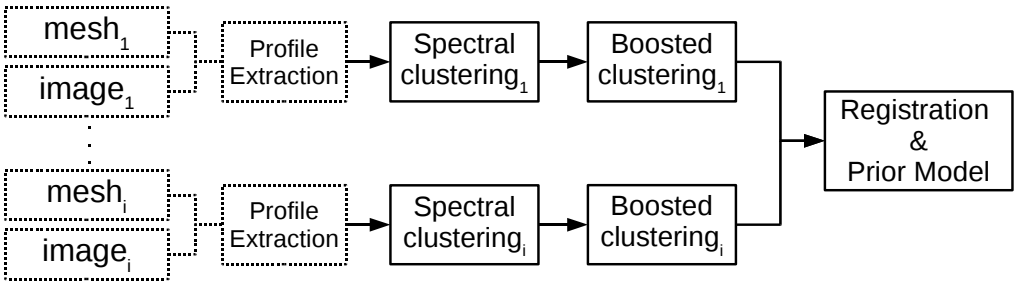


Figure 3: Pipeline for Multimodal Prior Appearance Model (MPAM) construction.

2.2. Intensity profiles

Intensity profiles were among the first image tools used to describe appearance for segmentation purposes. Sampled along the normal direction at each vertex, they can extend inward, outward or both sides, depending on the application. In computer vision, profiles act as feature vectors that could be replaced by any other local or global features such as isophote curvature, texture descriptors, oriented filters, ...

Cootes used intensity profiles to build Statistical Appearance Models [16, 14]. Intensity profiles are sampled in training images and both mean profile and its principal modes of variation are extracted for each landmark. During segmentation, the profile model associated with each landmark is compared with current intensity profiles sampled at a certain search window around the current landmark. This comparison is done using similarity measures like the Mahalanobis distance or the correlation coefficient. In their basic form, profiles are made of plain pixel/voxel intensity values sampled directly from the image. However, the gradient of those profiles may be used too, as well as the normalized version of both profiles.

For instance, intensity and gradient profiles were used to optimize image forces of deformable models [23]. The idea was to better discriminate organ contours in images by comparing intensity profiles, using two generic models and checking the similarity variation with the normalized cross-correlation. Normalized gradient profiles proved to be the best choice for face recognition [15]. For bone segmentation in radiographs, normalized intensity profiles gave the best results [4]. Combining these different types of profile into a larger feature vector has also been implemented [7]. A thorough study on

intensity profiles for 3D medical image segmentation can be found in [27].

2.3. Spectral clustering

Spectral clustering is used in computer vision, machine learning, pattern recognition and VLSI design [31, 32, 37, 1]. The idea is to perform clustering on a lower dimensional space built from the spectral analysis of similarity matrices. More precisely, it aims at creating a low dimensional space where the Euclidean distance between features roughly matches their geodesic distance when features lie on sub-manifolds rather than hyperplanes. Experiments in the literature have shown that such manifold learning approach has the advantage to reinforce the Gaussian hypothesis on input data points [31, 37] (see Figure 4). Another advantage is to ease the initialization issue, since data points are meant to be better represented with the similarity graph. Last but not least, curse of dimensionality is reduced. Indeed, data dimension is not related to profile length anymore, but rather to spectral data featuring a reduced number of dimensions.

The consecutive steps for spectral clustering are depicted in Figure 5. First, a similarity function needs to be defined between intensity profiles. This similarity function ensures the creation of a spectral graph, whose nodes represent the data points (*i.e.* intensity profiles). Two nodes are connected if corresponding data points are similar, *i.e.* if their similarity value is positive or larger than a threshold, and the connecting edge is weighted with the value returned by the similarity function. Spectral clustering consists in finding partitions of the graph such that edges between cluster of vertices have low weights (*i.e.* data points in different cluster are dissimilar from each other) and edges within a cluster have high weights (*i.e.* data points

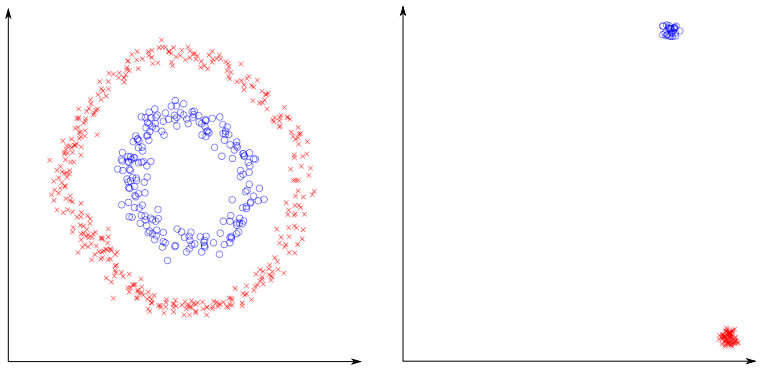


Figure 4: An example of spectral clustering applied on a set of 2D data points: representation of the dataset (a) before and (b) after spectral clustering (Source: [37]).

within the same cluster are similar from each other). Then, top eigenvectors are extracted from the affinity matrix associated with the spectral graph. The number of top eigenvectors may be computed in an automatic fashion with a model order selection based on a heuristic. These top eigenvectors are the new representation of data points, *i.e.* spectral data, which are classified with EM clustering.

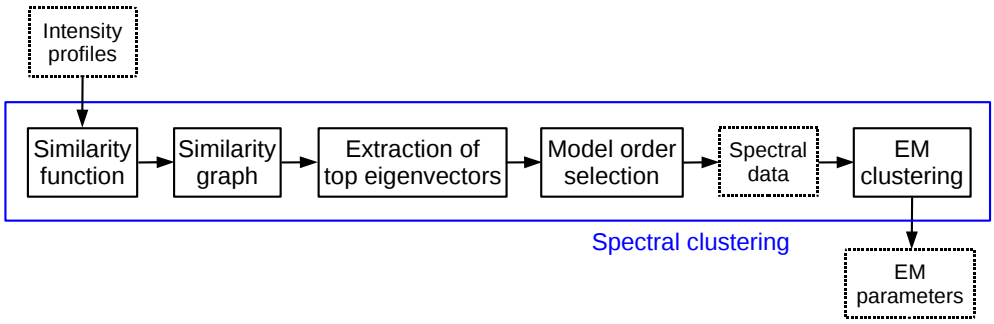
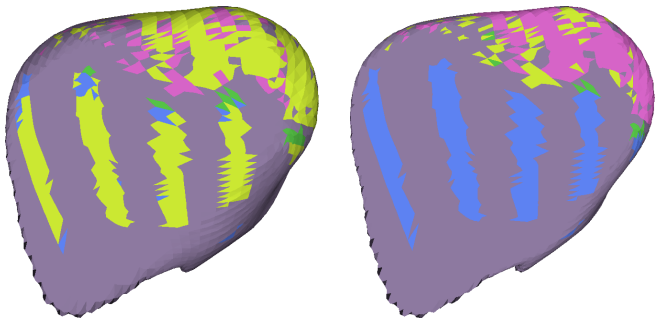


Figure 5: Pipeline for spectral clustering.

2.3.1. Similarity function

First step in spectral clustering consists in choosing the similarity function to determine how close two feature vectors are (*i.e.* how similar two data points are). In general, this function depends on the domain the data comes from [13, 3, 6, 31]. In the common case where feature vectors belong to the Euclidean space $\mathbb{R}^d$, the Gaussian kernel function is to be used. This is also our case because we want data points to be represented by a Gaussian Mixture Model. The Gaussian kernel function is defined as $S(\mathbf{x}_i, \mathbf{x}_j) = \exp(\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|^2}{(2\sigma^2)})$, where σ controls the scale of the neighborhoods. Parameter σ plays an important role in the graph construction, and thus on the spectral clustering. As there is no theory on how to determine σ, empirical experiments need to be done to optimize its value [34, 37, 21, 31].

With intensity profiles, we decide to compute σ as the intensity standard deviation (STD) of the tissue that best represents the structure of interest, multiplied by the number of dimensions. For instance, we choose parenchyma as the tissue that features at best the liver and estimate the standard deviation of its intensity in CT images at $STD = 60$. For a profile of length 16, this means we have $\sigma = 16 \times 60 = 960$. Tests have shown that this value gives satisfactory results. In addition, we added a neighboring weight ω_{nei} specific to simplex meshes to spatially regularize the clustering. The objective is to account for the connectivity between profiles by giving a higher similarity to neighboring profiles. Should j be among the three neighbors of i, we decrease their pairwise distance by ω_{nei} such that $S(\mathbf{x}_i, \mathbf{x}_j)' = \omega_{nei}\ S(\mathbf{x}_i, \mathbf{x}_j)$, where $\omega_{nei} \in [0, 1]$. Tests showed that SVD computation features convergence issues for $\omega_{nei} < 0.8$. However, $\omega_{nei} = 0.8$ gives good results, leading to an



(a) (b)

Figure 6: Spectral clustering of liver profiles spanning both sides with respect to neighboring weight ω_{nei}: (a) $\omega_{nei} = 1.0$ (no spatial regularization) and (b) $\omega_{nei} = 0.8$. See how the thoracic cage appearance becomes spatially smoother.

increased local neighborhood (see an illustration in Figure 6).

2.3.2. Similarity graph

After choosing the appropriate similarity function, the next step consists in transforming data points into vertices of a similarity graph. In the literature, several types of graph have been proposed [31]: fully connected graph (all vertices are connected with each other), ε-neighborhood graph (two vertices are connected only if their pairwise distance is smaller than a threshold ε) and k-nearest neighbor graph (two vertices are connected only if one vertex is among the k-nearest neighbor of the other).

We tested the three methods for the clustering of intensity profiles. Depending on mesh resolution, fully connected graph requires lots of memory. We thus tested the ε-neighborhood graph, with varying values of ε to overcome this problem. However, this implementation is not stable enough because some profiles are dissimilar to many others. Depending on the value of ε, some vertices have many neighbors while others have few of them (*i.e.*

corresponding rows of the affinity matrix are almost full or empty). As a consequence, SVD computation is sometimes not able to find a solution. This is why we use a k-nearest neighbor graph, so that each vertex, and thus each row, has exactly the same number of neighbors. Empirical tests on liver and bone profiles showed that $k = 14\% \times N$ gives the optimal results. Below this value, SVD computation does not converge (*i.e.* there is not enough information to compute eigenvectors). Conversely, increasing k does not seem to bring more information in SVD computation (*i.e.* eigenvectors are not significantly changed).

Finally, affinity matrix A, which is computed from the similarity graph to extract eigenvectors, is defined as $A_{ij} = S(\mathbf{x}_i, \mathbf{x}_j)$, where $i \neq j$, $A \in \mathbb{R}^{N \times N}$, $i \in k$-nearest neighbors of j (and *vice versa*), and $A_{ii} = 0$.

2.3.3. Top eigenvectors extraction

To compute top eigenvectors, Laplacian matrix L needs first to be computed. Laplacian matrices are the main concept of spectral clustering and lots of research has been conducted to study those matrices [35, 45, 37, 12]. To cluster intensity profiles, we decide to use the normalized Laplacian matrix L_{sym} because the associated spectral clustering algorithm [37] features an additional normalization step (*i.e.* by creating matrix Y) that is suitable for our proposed initialization of the EM algorithm using Gray code (see section 2.3.5). The normalized Laplacian matrix is defined as $L_{sym} = D^{\frac{-1}{2}} A D^{\frac{-1}{2}}$ [37], where D is a diagonal matrix whose (i, i)-element is the sum of i-th row of A.

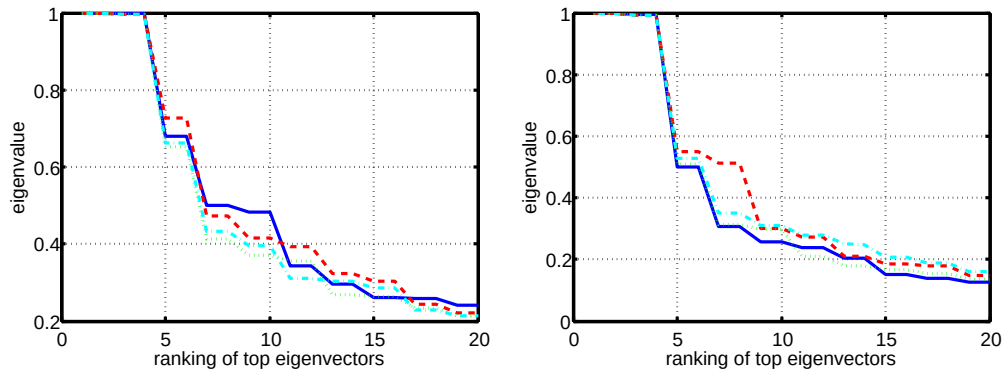
Next step consists in finding the K top eigenvectors of Laplacian matrix L. As there are as many eigenvectors as rows in L, computation requirement to

extract eigenvectors may be considerable when using fully connected graphs, especially for high resolution meshes. This is why ε-neighborhood and k-nearest neighbor are usually preferred, since they lead to sparse matrices that have the advantage to reduce the computation resources. Another point is that computing all eigenvectors is not necessary. Only top eigenvectors are needed for spectral clustering. The most popular methods to extract top eigenvectors from sparse matrices are the Krylov methods [25].

2.3.4. Model order selection

Instead of performing a model order selection on original data points, spectral clustering is aimed at using top eigenvectors for this operation. As previously explained, the advantage is that spectral data points feature a much stronger Gaussian distribution [31, 37]. An interesting property that is particularly suited for spectral clustering, and which is known as the *eigen-gap heuristic*, consists in combining a model order selection with the number of eigenvectors K. The objective is to select K as the number of eigenvalues $\lambda_1, \ldots, \lambda_k$ so that there is a significant gap δ between λ_k and λ_{k+1}. This heuristic has been justified in both perturbation and spectral graph theories [35, 12].

Advantage of such an heuristic is its invariance to eigenvalues, since the selection is not based on the eigenvalue *per se*, but rather on an eigenvalue difference. This feature is particularly interesting in our case because the number of intensity profiles may greatly vary depending on the mesh resolution. Our experiments showed that such a difference may have an impact on eigenvalues. They also showed that a gap $\delta = 0.15$ gives good results in selecting the number of modes for liver profiles (see a plot of eigenvalues in



(a) (b)

Figure 7: Plot of eigenvalues from 20 top eigenvectors, which are extracted from the profiles of 2 livers {(a),(b)} using the same reference mesh in four resolutions: $N = 3856$ (blue solid line), $N = 7586$ (red dashed line), $N = 11760$ (green dotted line), and $N = 35280$ (cyan dash-dot line). Note how eigenvalues feature the same steep decreases in the four resolutions (between the 4th and 5th top eigenvector).

Figure 7).

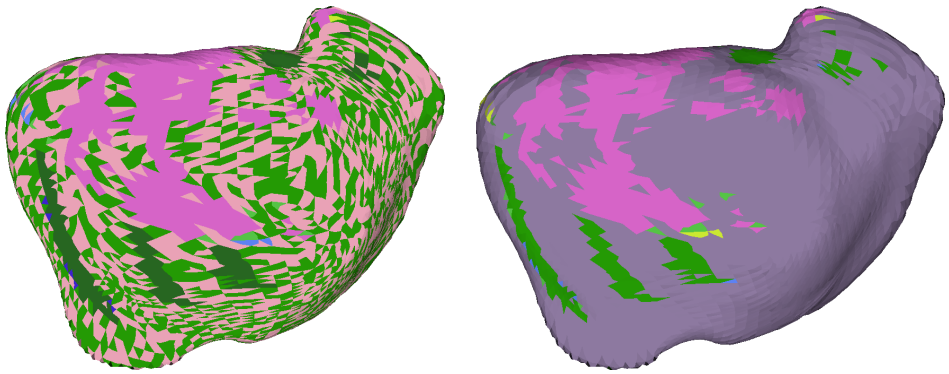
2.3.5. Clustering of spectral data

At this stage, we have found the K top eigenvectors of Laplacian matrix L. These K top eigenvectors have been stacked in columns to form matrix X. Since we are using the normalized Laplacian matrix L_{sym} (see section 2.3.3), the last transformation consists in normalizing matrix X. For that, matrix Y is formed by normalizing each row of matrix X to have unit length [37]: $Y_{ij} = X_{ij}/\sqrt{\sum_{j=1}^{N} X_{ij}^2}$. Matrix Y contains thus the new data points that are to be clustered. In the literature, this clustering is usually performed using K-Means [37]. In our case, we chose the EM algorithm as it is the most suited method to represent GMM. A specific care needs to be taken when initializing EM algorithm with spectral data points. Indeed, some eigenvalues

are likely to be similar. This is particularly the case with intensity profiles. Our experiments showed that corresponding eigenvectors are likely to lead to similar mode centers trapped in the same local minimum (see an illustration in Figure 8).

Since spectral data points are normalized when using Laplacian matrix L_{sym} (*i.e.* $X_{ij} \in \{0, 1\}$), we decide to initialize mode centers using the Gray code [41]. Gray code is a binary numerical system in which two successive values differ in only one bit. Doing so, we prevent mode centers from ending up in similar minima, since they are initialized orthogonal one from each other (*i.e.* unlike a classical random initialization where two mode centers may be close to each other). Caution needs to be taken though, as we have to ensure there are enough Gray codes for the number of mode centers K (*i.e.* $K < 2^d$, where d is the number of dimensions featured by the mode centers, which corresponds to the number of top eigenvectors in spectral clustering).

At the end of EM clustering, EM parameters from spectral clustering, which are referred to as $\hat{\pi}_k^p$, $\hat{\boldsymbol{\mu}}_k^p$ and $\hat{\boldsymbol{\Sigma}}_k^p$, may not be directly used with original data points because these parameters are computed in the spectral space (*i.e.* they do not have the same number of dimensions). However, posterior probabilities ${}^p\hat{\gamma}_i^k$ do not have this dimensional issue and may thus be used to recompute $\{\pi_k^p, \boldsymbol{\mu}_k^p, \boldsymbol{\Sigma}_k^p\}$ in one M-step on original data points $\mathbf{x}_i$. Mixing coefficients are computed as $\pi_k^p = \sum_{i=1}^N {}^p\hat{\gamma}_i^k / N$, means as $\boldsymbol{\mu}_k^p = \sum_{i=1}^N {}^p\hat{\gamma}_i^k \, \mathbf{x}_i / \sum_{i=1}^N {}^p\hat{\gamma}_i^k$ and covariance matrices as $\boldsymbol{\Sigma}_k^p = \sum_{i=1}^N {}^p\hat{\gamma}_i^k (\mathbf{x}_i - \boldsymbol{\mu}_k^p)(\mathbf{x}_i - \boldsymbol{\mu}_k^p)^T / \sum_{i=1}^N {}^p\hat{\gamma}_i^k$.



(a) (b)

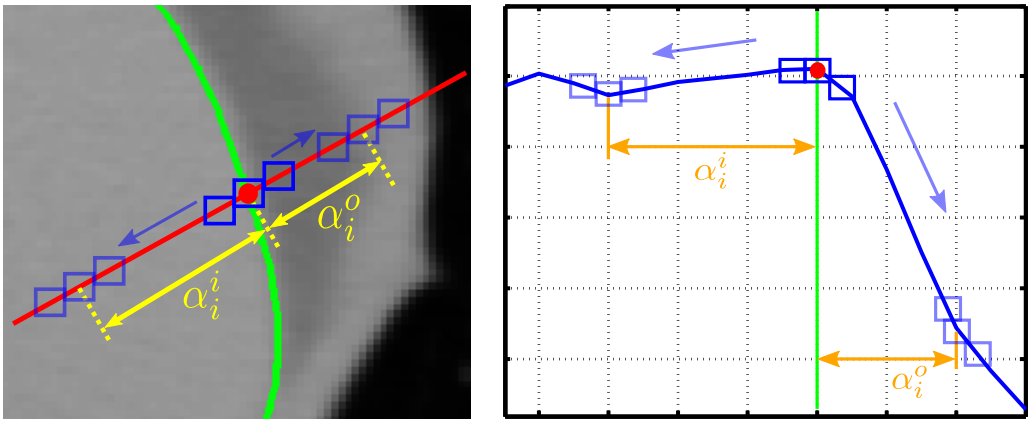
Figure 8: Spectral clustering of liver profiles spanning both sides and whose mode centers ($K = 9$ modes) are initialized: (a) using a random EM initialization, and (b) using Gray code. Note how the two main modes on the left (in green and light pink), whose centers are similar and trapped in the same local minimum, are separated on the right (main mode is in purple now) thanks to the Gray code initialization.

3. Regional appearance modeling

3.1. Overview

To efficiently use the clustering of intensity profiles as an appearance prior for segmentation, the clustering (set of centers and covariance matrices for each mode) must be suited for image segmentation. This is why we propose to perform a boosted clustering under a localization constraint, so that each profile mode describes in a discriminative way the boundary between the inside and the outside of the structure.

The boosted clustering relies on a classifier of intensity profiles. The objective is to tune an EM classification of profiles including a condition on offset α_i (see Figure 9), which states whether a profile is well localized by its associated modes. The condition on offset α_i is based on the hypothesis



(a) (b)

Figure 9: In boosted clustering, the search for the optimal offset α_i along the normal (red line) at a given vertex (red circle) consists in shifting a profile (blue squares) inward and outward (blue arrows) mesh surface (green line): (a) 2D slice of the CT image from which profiles are sampled, and (b) 2D plot of the corresponding intensities (α_i^i and α_i^o depict the optimal inward and outward offsets, respectively).

that at least one of the modes to which the profile belongs to must be able to localize the boundary along that profile. The objective is to encourage the creation of multiple modes with low variance, instead of modes with large covariance. The idea is to provide more discriminant modes to the MPAM during local search, *i.e.* when searching for the structure boundary during segmentation. Note, however, that with boosted clustering, the spatial smoothness of intensity profile clusters is no longer enforced.

First, we present the localization criterion on which the boosted clustering relies. Then, we describe three versions of the algorithm with increasing clustering results. Finally, we conclude with a discussion on the proposed algorithm.

3.2. Localization criterion

The objective of this localization criterion is to test if a given profile mode M_k (*i.e.* mean $\boldsymbol{\mu}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$) is able to unambiguously determine the boundary at a given vertex. More precisely, the offset which maximizes a similarity criterion (or minimizes a distance) between the current profile and the associated profile mode is computed and must be small enough.

First step consists in choosing an intensity similarity measure. As previously mentioned in the introduction, several measures have been proposed in the literature [36, 9, 38, 40, 39]. In our case, we decide to use *Mahalanobis distance*, *sum of absolute differences* and *linear criterion*. The linear criterion [36] is simply defined as the square correlation coefficient making the hypothesis of a linear relationship between intensity of two profiles.

The objective is to compute the offset α_i that maximizes a similarity value (*i.e.* linear criterion) or minimizes a distance (*i.e.* sum of absolute differences and Mahalanobis distance). In practice, the profile is regularly sampled along the normal (*e.g.* every millimeter) and profile offsets are obtained by shifting the profile inward and outward (see an illustration in Figure 9).

Let $\mathbf{x}(\mathbf{p}_i, \mathbf{n}_i)$ be the intensity profile at vertex $\mathbf{p}_i$ along normal $\mathbf{n}_i$, α_i be the offset of that profile along the normal as $\mathbf{x}(\mathbf{p}_i + \alpha_i \mathbf{n}_i, \mathbf{n}_i)$, and $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$ be, respectively, the mean and covariance matrix of profile mode M_k the profile $\mathbf{x}(\mathbf{p}_i, \mathbf{n}_i)$ belongs to.

Offset $\{\alpha_i\}_{\text{line}}$ needs to maximize the linear criterion:

$$\{\alpha_i\}_{\text{line}} = \arg\max_{\alpha_i,k} \left(\frac{(\mathbf{x}(\mathbf{p}_i + \alpha_i \mathbf{n}_i, \mathbf{n}_i) \cdot \boldsymbol{\mu}_k)^2}{(\mathbf{x}(\mathbf{p}_i + \alpha_i \mathbf{n}_i, \mathbf{n}_i)^2) \cdot (\boldsymbol{\mu}_k^2)} \right) \quad (1)$$

Both offsets $\{\alpha_i\}_{\text{diff}}$ and $\{\alpha_i\}_{\text{maha}}$ need to minimize the sum of absolute differences and the Mahalanobis distance, respectively:

$$\{\alpha_i\}_{\text{diff}} = \arg\min_{\alpha_i,k} \left(|\mathbf{x}(\mathbf{p}_i + \alpha_i \mathbf{n}_i, \mathbf{n}_i) - \boldsymbol{\mu}_k| \right) \quad (2)$$

$$\{\alpha_i\}_{\text{maha}} = \arg\min_{\alpha_i,k} \left((\mathbf{x}(\mathbf{p}_i + \alpha_i \mathbf{n}_i, \mathbf{n}_i) - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}(\mathbf{p}_i + \alpha_i \mathbf{n}_i, \mathbf{n}_i) - \boldsymbol{\mu}_k) \right) \quad (3)$$

A small α_i means there exists one mode that can well localize the boundary at that point. Otherwise, it means either that the current profile has no suitable feature to localize the structure boundary, or that profile mode M_k (*i.e.* mean $\boldsymbol{\mu}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$) is not suitable to represent it. Should α_i be greater than a defined threshold (*e.g.* 2mm), current point is considered as not being well represented by the current classification (*i.e.* considered as an outlier). Using offset α_i as a weak classifier, we thus enforce the classification associated with the MPAM to produce profile modes that are suitable for image segmentation.

3.3. Boosted clustering

3.3.1. Single pass boosted clustering

As depicted in Figure 10, this first version of the boosted clustering consists in applying a weak classifier on the intensity profiles after spectral clustering. This weak classifier is based on offset α_i. Profiles whose offset α_i is

greater than the threshold are rejected. The other profiles are kept. However, EM parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ must be updated to retrieve the influence of the rejected profiles on the classification. With these updated EM parameters, the weak classifier must be launched again to check whether kept profiles are still featuring an offset α_i lesser or equal to the threshold. If so, profiles are kept; otherwise, they are rejected. This procedure is iterated until a steady state is reached, *i.e.* when the number of rejected profiles becomes negligible (see Figure 10, blue arrows).

However, our experiments showed that the boosted clustering only keeps between 30% and 60% of the total number of profiles, which is clearly not enough for a further segmentation procedure. This is explained by the important number of profiles that are rejected, not only because their associated classification is not discriminant enough with respect to offset α_i at the first iteration, but also because their updated classification does not improve during the iterative procedure. In next section, we propose to iteratively re-estimate rejected profiles as new GMM in an attempt to increase the number of kept profiles.

3.3.2. Cascading boosted clustering

In this new version of the boosted clustering, depicted in Figure 11, the profiles that have been rejected are now considered as input to another iteration of the algorithm. At each iteration, rejected profiles are represented by a new GMM (*i.e.* new $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$). By cascading the weak classifier, the number of profiles that are classified while satisfying offset α_i is constantly increasing after each iteration (see Figure 11, red arrows). In theory, the boosted clustering should stop when offset α_i is met by all profiles. In practice, the

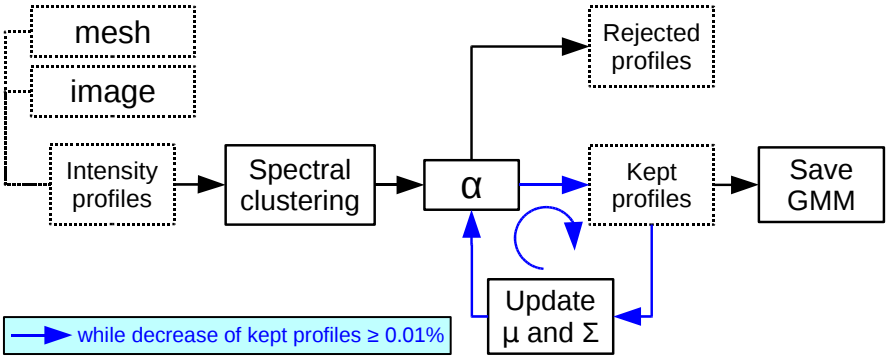


Figure 10: Pipeline for single pass boosted clustering.

algorithm stops when a steady state is reached, *i.e.* when the additional number of kept profiles is below a defined threshold. Each iteration of the weak classifier increases the number of profile modes that are found to be adequate to represent the appearance of the structure in the image.

Our experiments showed that cascading boosted clustering keeps around 50% to 70% of the total number of profiles. Although those results are better than spectral clustering algorithm described in previous section, they are still unsatisfactory and could be improved. This unsatisfactory result is likely to be due to the spectral clustering performance on a number of profiles that may decrease dramatically after each iteration. Also, there is no guarantee that the eigengap heuristic used for model order selection performs the same way with this decreased number of profiles. As a solution, we propose in the next section to change the model order selection based on the eigengap heuristic by a hierarchical approach.

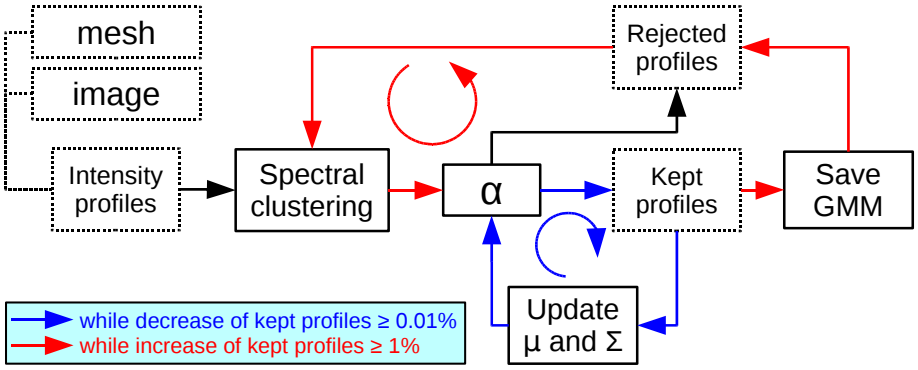


Figure 11: Pipeline for cascading boosted clustering.

3.3.3. Cascading boosted clustering with hierarchical approach

In Figure 12, we depict the cascading boosted clustering with hierarchical approach. Though the initialization is still performed using the eigengap heuristic, the model order selection performed at each iteration of the cascading boosted clustering is now replaced by a hierarchical approach [5]. Instead of systematically saving a new GMM after each iteration (see Figure 11, red arrows), the consistency of the GMM is now tested. To do so, the ratio of kept profiles (*i.e.* number of kept profiles with respect to the total number of profiles N) is computed.

If this ratio is higher than a threshold ρ (*e.g.* 3 % of the total number of profiles N), the GMM is saved (*i.e.* considered as consistent) and a new iteration of the cascading boosted clustering is launched (see Figure 12, red arrows). A low threshold ρ enables the creation of GMM composed of modes with a small number of profiles, which may be useful to capture the appearance of very small or very specific regions around the structure. Otherwise, kept profiles are rejected and model order selection based on the hierarchical

approach is launched (see Figure 12, green arrows). The objective is to classify rejected profiles using spectral clustering, but with a minimum number of modes K_{min} this time. At the next iteration, if kept profiles still do not reach threshold ρ, they are discarded again and re-classified using K_{min+1} modes. This procedure is performed in an iterative way until K_{max} modes are reached.

The number of modes is increased at each iteration so that the GMM receives more and more degrees of freedom to represent the data points (*i.e.* kept profiles). This is an important principle of unsupervised clustering: the more clusters, the better data are represented, but at a cost of a more complex model, which usually leads to overfitting. An appropriate K_{max} must thus be carefully chosen to avoid this undesirable effect. The cascading boosted clustering with hierarchical approach may end in two different ways. Either K_{max+1} modes have been reached at the end of the model order selection based on the hierarchical approach (*i.e.* meaning that K_{max+1} modes are not enough to ensure the classification of at least ρ% of kept profiles), or a steady state is reached (*i.e.* the additional number of kept profiles is below a defined threshold, similarly to cascaded boosted clustering). Cascading boosted clustering with hierarchical approach leads in average to 75% to 99% of kept profiles, which we consider as a satisfying number of kept profiles for a further segmentation procedure.

3.3.4. Discussion

The advantage of using a hierarchical approach is to test the GMM consistency at each iteration. Should one GMM not be consistent (*i.e.* when number of associated kept profiles is below threshold ρ), the classification is

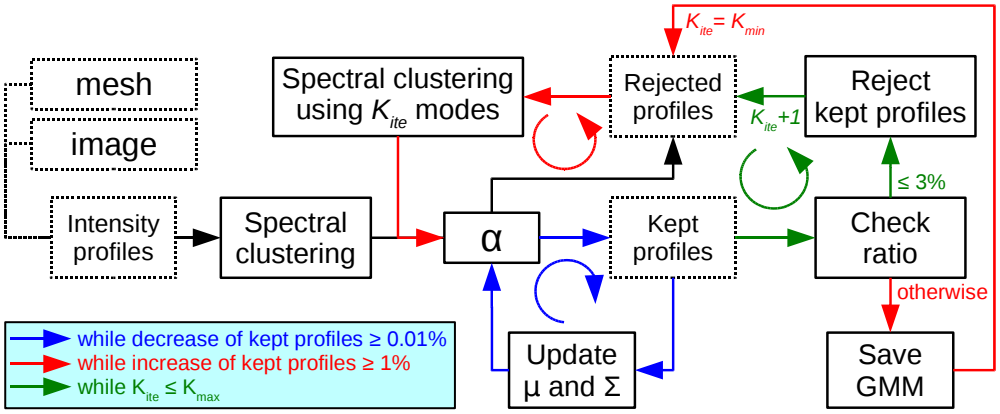


Figure 12: Pipeline for cascading boosted clustering with hierarchical approach.

optimized by increasing the number of modes K_{ite} (*i.e.* by giving the GMM more degrees of freedom to represent kept profiles). Our experiments showed that a $\rho = 3\%$ gives satisfactory results. Another advantage of this hierarchical approach is to enforce a minimum mode size using threshold ρ, which prevents the creation of tiny modes. The profiles that are not classified by the cascading boosting clustering correspond to small regions with a specific appearance but without enough profiles to robustly estimate a mean and covariance matrix. Those profiles could also be associated with regions with no salient features to describe the boundaries. At the end of the cascading boosted clustering on mesh $\mathcal{M}_p$, we obtain K_p modes M_k^p and a set of posterior probabilities ${}^p\gamma_i^k$ indicating the probability that a vertex $\mathbf{p}_i$ belongs to mode M_k^p.

4. Appearance model construction

To build the MPAM, a reference mesh $\mathcal{M}^\star$ is required to store the appearance information. The objective is to provide a geometric embedding

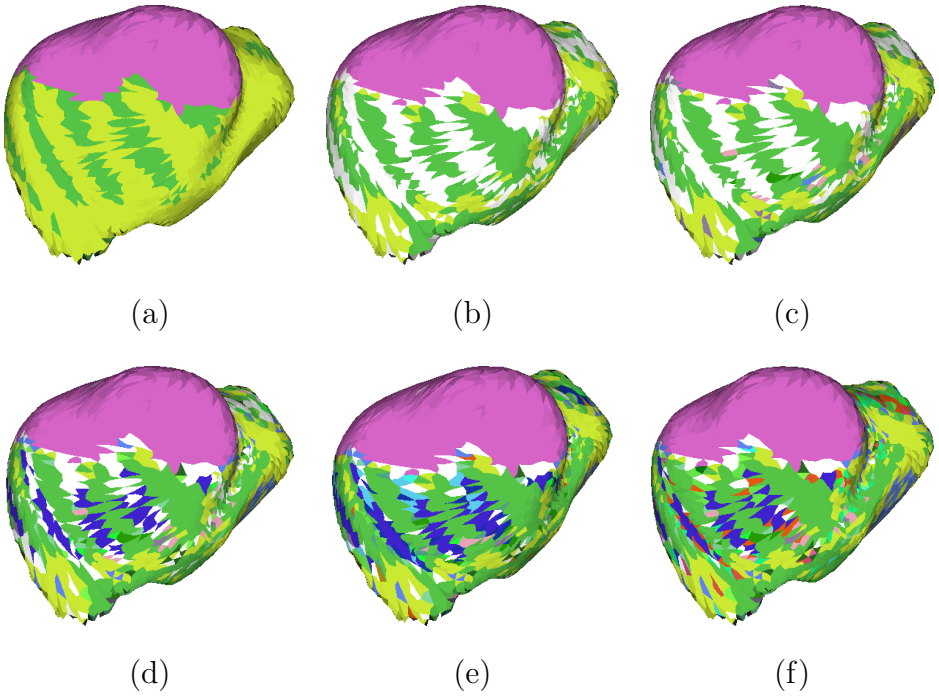
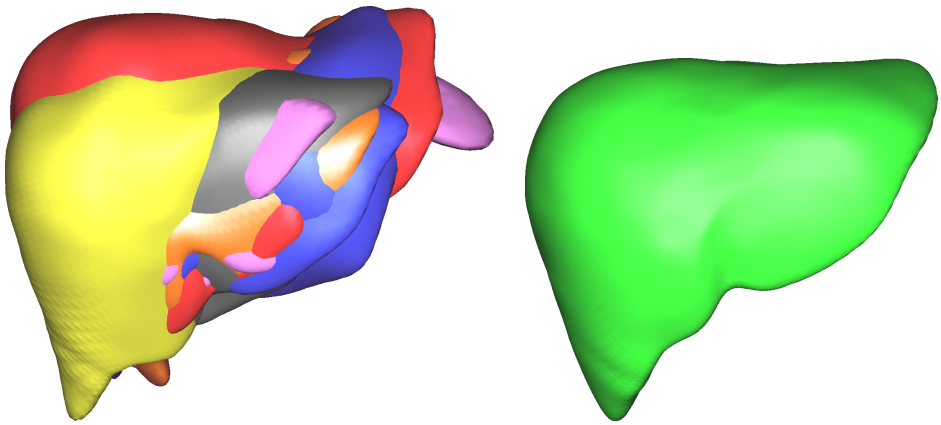


Figure 13: Boosted clustering applied on profiles sampled both sides of a liver mesh. The algorithm needed 5 global iterations to converge. First, (a) spectral clustering is applied as initialization step. Note that thoracic cage is represented by only one mode. After first global iteration, (b) only 58% of profiles are considered as well classified by the offset α_i. See how profiles representing the thoracic cage have been discarded (discarded profiles are highlighted by white vertices). Then, the cascading boosted clustering optimizes the classification to reach (c) 67%, (d) 77%, (e) 85%, and (f) 92% of well classified profiles. Note that thoracic cage is now represented by multiple modes.

for the independent modes, in which the P EM classifications need to be *registered*. More precisely, posterior probabilities ${}^{p}\gamma_i^k$ from the P datasets should be projected into this common reference framework. Because all profile modes are computed independently to any registration, and because the MPAM is multimodal (*i.e.* less dependent to the point correspondence), our approach is not as sensitive to registration errors as the classical PCA-based methods [27]. However, there is still a need to build a mapping between the reference mesh $\mathcal{M}^\star$ and each mesh instance $\mathcal{M}_p$.

Reference mesh $\mathcal{M}^\star$ construction. To build reference mesh $\mathcal{M}^\star$, a mesh-to-mesh registration is performed. The method consists in first randomly initializing the reference mesh as one of P meshes. As a first rigid transformation, we use a robust point set registration using Gaussian mixture models to perform a rigid registration without point correspondence [28]. Then, we register the current reference mesh on all meshes before finally recomputing the reference mesh as the mean of the deformed instances (see an illustration in Figure 14, left). This process is iterated several times until the mesh reaches a steady state, *i.e.* typically 3 iterations in our case (see an example in Figure 14, right).

Registration between $\mathcal{M}^\star$ and $\mathcal{M}_p$. After defining a reference mesh $\mathcal{M}^\star$, a registration between the reference mesh $\mathcal{M}^\star$ and the P instance meshes needs to be performed. This registration may be done in both ways: deforming each instance towards the reference mesh (see an illustration in Figure 15, left), or deforming the reference mesh towards each mesh (see an illustration in Figure 15, right). In both cases, we propose to use a mesh-to-mesh registration based on currents [24, 46], which have been successfully used on



(a) (b)

Figure 14: Creation of the reference mesh $\mathcal{M}^\star$ from 16 liver meshes using a mesh-to-mesh registration based on currents: (a) registration of all instance meshes towards the current reference mesh, and (b) creation of the final reference mesh $\mathcal{M}^\star$ as the mean of all registered instance meshes.

brain [20] and heart [33] meshes. This approach has the advantage to be non-parametric, since shapes are represented by currents. The mesh deformation during registration is smooth and a coarse-to-fine framework may be used by changing the width of the Gaussian kernel used to represent both currents and diffeomorphisms. The former parameter is a fit parameter that attracts $\mathcal{M}_p$ to $\mathcal{M}^\star$ while the latter is a rigidity parameter that tries to keep $\mathcal{M}_p$ as smooth as possible.

Transfer of appearance information. Last step consists in an interpolation method required to transfer appearance information from $\mathcal{M}_p$ to $\mathcal{M}^\star$ (*i.e.* the clustering associated with $\mathcal{M}_p$ needs to be projected into $\mathcal{M}^\star$). For this purpose, we search for each vertex $\mathbf{p}_i$ of the reference mesh its closest points $Cl(\mathbf{p}_i)$ on mesh $\mathcal{M}_p$, and also consider its neighbors. We then assign in $\mathbf{p}_i$ the

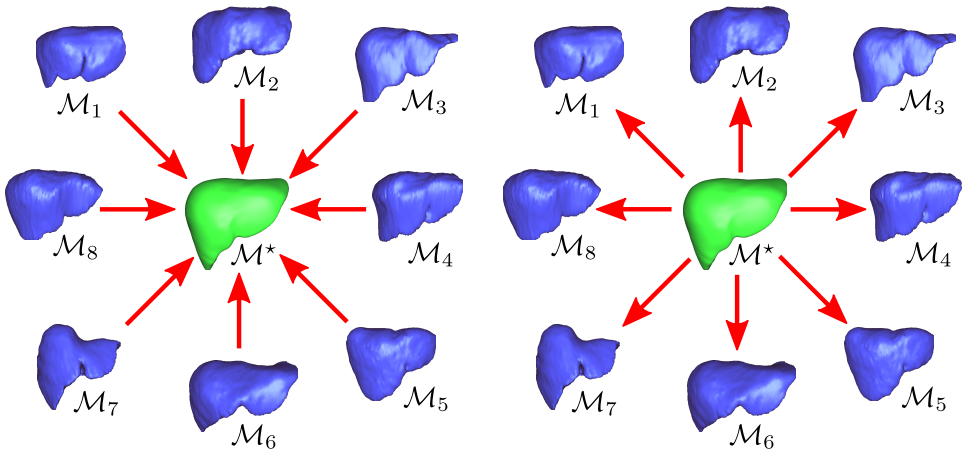
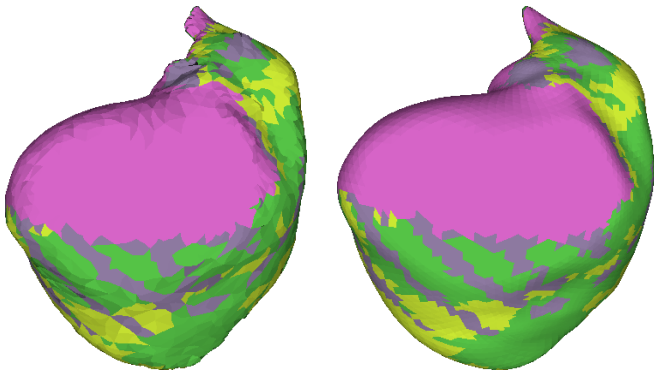


Figure 15: The registration between the reference mesh $\mathcal{M}^\star$ and 8 instance meshes $\{\mathcal{M}_1, \ldots, \mathcal{M}_8\}$ may be done in both ways: (a) deforming each instance mesh towards the reference mesh, or (b) deforming the reference mesh towards each instance mesh.

modes of $Cl(\mathbf{p}_i)$ and its neighbors and interpolate the posterior probabilities, the interpolation being inversely proportional to the distance. This interpolation not only compensates any misregistration that may have occurred but also smooths the clustering projection. To ensure that the clustering information from all P meshes $\mathcal{M}_p$ is captured by reference mesh $\mathcal{M}^\star$, the latter should be finer in terms of resolution (*i.e.* number of vertices) than the finest mesh $\mathcal{M}_p$. This way, there is always at least one vertex from $\mathcal{M}^\star$ to capture the associated clustering information from $\mathcal{M}_p$ vertices (see an illustration in Figure 16). Finally, for each vertex $\mathbf{p}_i$ of $\mathcal{M}^\star$, we compute the posterior probability $\tilde{\gamma}_i^m$ by summing and normalizing the posterior probabilities associated to each mode such that $\tilde{\gamma}_i^m = \sum_{p=1}^{P} {}^p\gamma_i^{\eta(p,m)} / \sum_{m=1}^{K} \sum_{p=1}^{P} {}^p\gamma_i^{\eta(p,m)}$, where P is the number of meshes $\mathcal{M}_p$, K is the number of modes associated



(a) (b)

Figure 16: Transfer of appearance information from (a) instance mesh $\mathcal{M}_p$ ($N = 4530$ vertices) to (b) reference mesh $\mathcal{M}^\star$ ($N = 11760$ vertices) using a closest point approach. Note how similar is the clustering despite the change of mesh resolution.

with $\mathcal{M}^\star$ and $\eta(p, m)$ is the equivalence table described in previous section. In practice, this approach leads to sparse posterior probabilities where only a few modes have non-zero posterior probabilities (*i.e.* non-negligible values). By filtering negligible posterior probabilities, we reduce the number of modes associated to each vertex during the local search performed by the regional external forces, which greatly improves the computational time.

4.1. Regional external forces based on the MPAM

External forces attract the mesh to image-based features in order to fit a desired region of interest. Those forces, which are expressed as a displacement field, compute the displacement of each vertex using information coming from the MPAM. For intensity profiles, a local search is performed (see an illustration in Figure 9). The new vertex position $\mathbf{p}(\mathbf{x}_i)'$, or *target position*, is computed as $\mathbf{p}(\mathbf{x}_i)' = \mathbf{p}(\mathbf{x}_i) + (\alpha_i\ \mathbf{n}_i)$, where $\mathbf{n}_i$ is the normal at vertex $\mathbf{p}(\mathbf{x}_i)$ and α_i is the offset distance from $\mathbf{p}(\mathbf{x}_i)$ and along $\mathbf{n}_i$ for which the

profile maximizes its similarity measure with one of the modes M_k profile $\mathbf{x}_i$ belongs to.

In this paper, we propose regional external forces based on the MPAM. Though only one offset per vertex is considered, the comparison is done with multiple modes during local search. Indeed, since the MPAM is multimodal, more than one mode may be assigned to $\mathbf{x}_i$. More precisely, every current profile sampled during local search is compared with every mode M_k, hence the multimodality of the MPAM. In practice, only modes whose posterior probability is higher than a threshold (*e.g.* 10^{-3}) are considered. This may be done because EM classification leads to unfuzzy posterior probabilities whose values are either very high or very low (*i.e.* quite below 10^{-3}).

To compare profile $\mathbf{x}_i$ with multiple modes, we used the three similarity measures presented in section 3.2. For the Mahalanobis distance, only the diagonal term σ_k of covariance matrix $\mathbf{\Sigma}_k$ may be considered. This has the advantage to reduce computational time, which may be considerable since the comparison is multimodal. Finally, regional external force ${}^R f_{ext}(\mathbf{p}(\mathbf{x}_i))$ drives the vertex $\mathbf{p}(\mathbf{x}_i)$ in the direction of the target position $\mathbf{p}(\mathbf{x}_i)'$ such that ${}^R f_{ext}(\mathbf{p}(\mathbf{x}_i)) = \beta(\mathbf{p}(\mathbf{x}_i)' - \mathbf{p}(\mathbf{x}_i))$, where β is the global weight controlling the strength of the external forces.

5. Results

5.1. Data

To compute significant statistical results, we use a database of 35 liver meshes, coming from both 3Dircadb1[1] and SLIVER07[2] databases. 3Dircadb1 is a database of 20 liver meshes provided by IRCAD, the French Research Institute against Digestive Cancer[3]. SLIVER07 is a database of 20 segmented liver images provided by the Workshop on 3D Segmentation in the Clinic that was held in conjunction with MICCAI 2007 conference[4]. 3Dircadb1 contains meshes with different number of vertices (*i.e.* no point correspondence). SLIVER07 contains binary images as segmentation and the Marching Cubes algorithm [30] has thus been used to generate meshes from these binary images. Since mesh generation from Marching Cubes algorithm depends on mesh volume and shape, generated meshes from SLIVER07 database also feature different number of vertices.

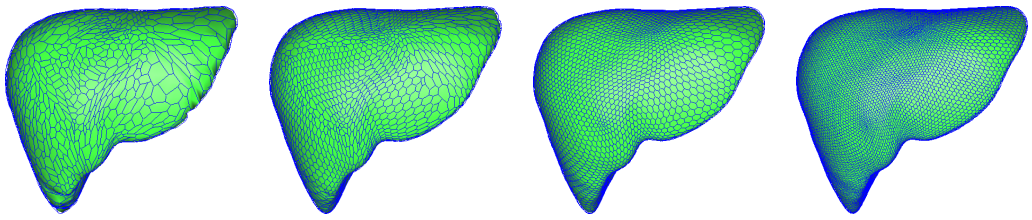
To have a meaningful comparison between datasets, a reference mesh $\mathcal{M}^\star$ ($N = 7586$ vertices) is created using the method presented in section 4. More precisely, one liver mesh from 3Dircadb1 is randomly selected as the reference mesh $\mathcal{M}^\star$ and registered to 16 other liver meshes from 3Dircadb1 using the mesh-to-mesh registration based on currents [24, 46]. Then, an average mesh is computed from the 16 registered meshes (using the average position at each vertex) and constitutes the new version of the reference mesh $\mathcal{M}^\star$. The

[1] http://www.ircad.fr/softwares/3Dircadb/3Dircadb1

[2] http://sliver07.isi.uu.nl

[3] http://www.ircad.fr

[4] http://www.miccai2007.org



(a) (b) (c) (d)

Figure 17: Four resolutions of reference mesh $\mathcal{M}^\star$: (a) $\mathcal{M}_3^\star$ ($N = 3856$ vertices), (b) $\mathcal{M}_2^\star$ ($N = 7586$ vertices), (c) $\mathcal{M}_1^\star$ ($N = 11760$ vertices), and (d) $\mathcal{M}_0^\star$ ($N = 35280$ vertices).

whole registration procedure is performed twice to reach a steady state (*i.e.* a reference mesh $\mathcal{M}^\star$ that is stable between two iterations). Finally, reference mesh $\mathcal{M}^\star$ is registered to the 35 liver meshes of our database using the mesh-to-mesh registration based on currents. Those registrations serve as ground truth for our experiments and are denoted as *ground truth liver meshes* in the remainder of this section.

5.2. Clustering of intensity profiles

The boosted clustering of intensity profiles based on spectral clustering is expected to achieve two main goals. First, a reasonable percentage of kept profiles for every dataset should be reached to get a good representation of all appearance regions around the structure. Second, the clustering should behave similarly for different mesh resolutions in terms of percentage of kept profiles and number of modes.

We have launched the boosted clustering with four resolutions of reference mesh from 3856 to 35280 vertices as shown in Figure 17. In Table 1, the average percentage of kept profiles per dataset ($\overline{\%}$) and the average number of modes per dataset after boosted clustering ($\overline{K}$), *i.e.* $\overline{K} = (\sum_{p=1}^{P} K_p)/P$ has

	$\mathcal{M}_3^\star$		$\mathcal{M}_2^\star$		$\mathcal{M}_1^\star$		$\mathcal{M}_0^\star$	
	$\overline{\%}$	$\overline{K}$	$\overline{\%}$	$\overline{K}$	$\overline{\%}$	$\overline{K}$	$\overline{\%}$	$\overline{K}$
[SingPassBoost]	41	5	41	5	43	5	43	5
[CascBoost]	55	14	56	45	56	12	53	11
[CascBoostHier]	91	40	91	40	91	40	91	41

Table 1: Results of the boosted clustering for four resolutions of reference mesh $\mathcal{M}^\star$: $\mathcal{M}_3^\star$, $\mathcal{M}_2^\star$, $\mathcal{M}_1^\star$ and $\mathcal{M}_0^\star$ (see Figure 17): average of the percentage of kept profiles ($\overline{\%}$) and average number of modes per dataset ($\overline{K}$). The three versions of the boosted clustering are tested: single pass boosted clustering [SingPassBoost], cascading boosted clustering [CascBoost], and cascading boosted clustering with a model order selection based on a hierarchical approach [CascBoostHier].

been evaluated for the three versions of the boosted clustering presented in section 3.3 : single pass boosted clustering, the cascading boosted clustering and the cascading boosted clustering with a model order selection based on a hierarchical approach.

From this table, it is clear that the single pass clustering has a stable number of modes through all resolutions but is very limited in terms of appearance description since with very few modes it can localize only around 42% of the intensity profiles per datasets. The cascading boosted clustering performs a bit better in terms of segmentation ability (with around 55 % of kept profiles) but is not stable accross mesh resolutions with $11 \leq \overline{K} \leq 45$. Finally, the cascading clustering with a hierarchical approach leads to much better results with a stable number of classes accross resolutions ($\overline{K} \approx 40$) and excellent segmentation performances with nearly 91% of kept profiles.

5.3. Segmentation results

The initialization is the first step of any explicit model-based segmentation procedure. The mesh may be approximately initialized into the image because many structures of interest have a shape and location into the body that are well known. In the literature, different initialization techniques have been proposed, depending on the model used [27, 22, 44, 19].

In our case, the ground truth binary segmentation of 12 liver meshes from our database was first meshed with the Marching Cubes algorithm [30] and then the reference mesh was non-rigidly deformed onto that mesh with a mesh-to-mesh registration based on currents [24, 46]. The initialization for the segmentation was obtained by computing the optimal similarity transformation (in the least-square sense) between the reference mesh and the deformed reference mesh, knowing the point correspondence between the two meshes (see an illustration in Figure 18).

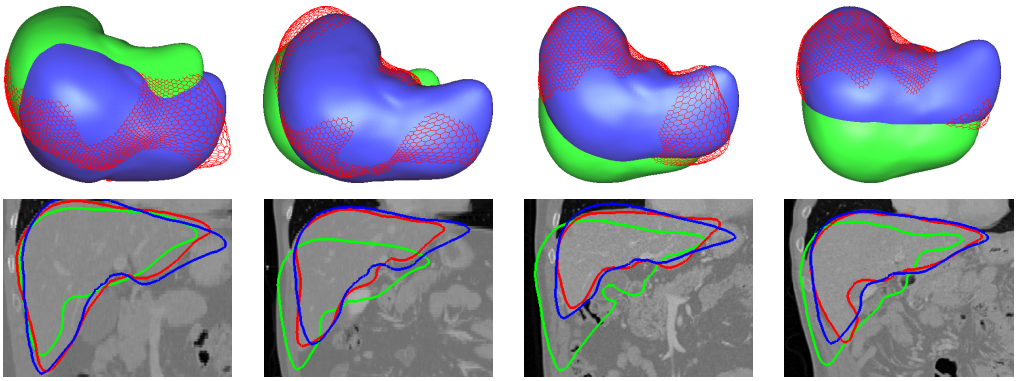
To model the regional appearance around livers, intensity profiles extending from 9 mm inward to 6 mm outward are sampled every mm for every ground truth liver mesh of our database. These profile parameters are chosen because they have been successfully used for liver segmentation in the literature [27]. Then, intensity profiles are classified using spectral clustering and the eigengap heuristic is used to automatically select the number of modes (*i.e.* after the extraction of 20 top eigenvectors whose eigenvalues are used by the eigengap heuristic). Finally, the classification associated with each mesh is optimized using the boosted clustering with a hierarchical approach and projected on reference mesh $\mathcal{M}^\star$.

However, the classification associated with each of those 12 livers is not

included into both the MPAM and PCAP (*i.e.* using a Leave-One-Out cross validation), so that external forces coupled with this initialization stage are fully studied without bias (*i.e.* without using appearance information from the datasets that are segmented). We use the different combinations of appearance priors and similarity measures that gave the best results in our experiments, namely MPAM with Mahalanobis distance, PCAP with sum of absolute differences (PCAP+diff), PCAP with linear criterion (PCAP+line) and PCAP with ratio on eigenvalues $\mathcal{R} = 0.9$ (PCAP+maha). Mean and standard deviation of segmentation results assessed with ASSD, VOE, MSD and SRVD [26] may be found in Table 2.

For both MPAM and PCAP, internal forces are generated using the shape memory associated with simplex meshes [36], which is a very basic shape prior that we used to focus on the external forces generated from our appearance model. The deformations used for segmentation are performed in 100 iterations, the strength of both internal and external forces are set to $\alpha = 0.3$ and $\beta = 0.1$, respectively, and mesh rigidity is set to $\eta = 10$. Those segmentation parameters have been chosen because they give reasonable results in terms of mesh regularity in our experiments.

Results assessed by ASSD, VOE and MSD in Table 2 show that our MPAM outperforms the PCAP, despite having fewer modes (*i.e.* $\overline{K} = 1250$ for MPAM whereas $N = 7586$ for PCAP). SRVD seems to indicate that all methods slightly undersegment, which is probably due to the influence of the larger inner part of the profile compared to its outer part, *i.e.* profiles are extending from 9 mm inward to 6 mm outward. Also, the inner part of the liver (*i.e.* the parenchyma) features a less variance, which makes it more



(a) (b) (c) (d)

Figure 18: Registration of reference mesh $\mathcal{M}^\star$ ($N = 7586$ vertices) towards four liver meshes {(a),(b),(c),(d)} using a similarity transformation. Results are depicted at mesh (top) and image (bottom) level. Original reference mesh is in green, target liver mesh in red and registered reference mesh in blue.

discriminant. For more information about the segmentation results, we refer the reader to [10].

6. Conclusion

In this paper, we proposed the construction of Multimodal Prior Appearance Models (MPAM) from the regional clustering of intensity profiles as a novel way to model the appearance around structures of interest in medical images. The clustering is considered as *regional* because intensity profiles are classified for each mesh, and not for each vertex (*i.e.* over a population of meshes). Unlike PCA-based methods that need an accurate point correspondence, our approach creates modes without requiring an accurate pointwise registration. This is because the statistical analysis of appearance is performed at each mesh, and not at each point. One advantage of this

	ASSD [mm]	VOE [%]	MSD [mm]	SRVD [%]
[MPAM+maha]	2.24 ± 1.08	12.99 ± 5.04	25.74 ± 8.85	-5.66 ± 5.59
[PCAP+diff]	3.74 ± 1.56	20.53 ± 7.17	31.74 ± 7.00	-6.20 ± 11.5
[PCAP+line]	4.14 ± 1.43	22.24 ± 5.76	33.26 ± 7.13	-1.52 ± 6.78
[PCAP+maha]	4.14 ± 1.07	22.76 ± 4.52	29.90 ± 6.58	-2.00 ± 11.7

Table 2: Mean ± STD of the segmentation results performed on 12 livers and assessed with ASSD, VOE, MSD and SRVD when using MPAM and PCAP with different similarity measures: MPAM with Mahalanobis distance (MPAM+maha), PCAP with sum of absolute differences (PCAP+diff), PCAP with linear criterion (PCA+line), and PCAP with ratio on eigenvalues $\mathcal{R} = 0.9$ (PCAP+maha).

feature is that a meaningful MPAM may be built with very few datasets (in fact one dataset suffices), which makes it well suited for a bootstrapping strategy. Since multimodal, the MPAM is able to cope with large variations of appearance including pathologies. This is not the case with PCA-based methods, which rely on the hypothesis that the probability density function is well described by one single Gaussian distribution. As we explained, this hypothesis is often violated by the presence of pathologies but also by the fact that shape is not necessarily correlated with appearance.

To classify intensity profiles, we proposed to perform spectral clustering. To do so, we explained how to transform intensity profiles (*i.e.* original data points) into vertices of a similarity graph using a similarity function, which leads to an affinity matrix. In fact, the new representation of data points used by spectral clustering comes from top eigenvectors extracted from this affinity matrix and we explained how to use sparse matrices to deal with high resolution meshes. To automatically determine the number of top

eigenvectors, we described the eigengap heuristic. Finally, the classical EM algorithm is used with those top eigenvectors as input data and we proposed to initialize centers with the Gray code to deal with similar eigenvalues. From our experiments, we noticed that spectral clustering is much more faster than original EM algorithm. This is because model order selection is not launched for every possible K, but performed once using the eigengap heuristic. Also, the use of sparse matrices makes the spectral clustering faster.

To improve MPAM for segmentation purposes, we presented regional external forces based on the boosted clustering. In addition to learning appearance through datasets, the boosted clustering tunes the clustering of profiles during an iterative local search, which provides some guarantee about the boundary localization by the profile modes. During this iterative procedure, the spectral clustering and a hierarchical approach are used by the cascading boosted clustering to optimize the classification of profiles. We presented a localization criterion to decide whether the clustering associated to a vertex is able to localize well the boundary along the profile. If not, the profile associated to the vertex is rejected and possibly clustered at a next iteration. At the end of the iterative procedure, if the cascading boosted clustering is still not able to localize well the boundary along corresponding profiles, associated vertices are considered as outliers. The objective is to avoid their influence during the segmentation. Finally, all mesh instances are projected on a common reference framework.

Segmentation results from comparative tests on liver profiles show that our MPAM outperforms PCAP despite the fact that fewer profile modes are used. However, these comparative tests were done using only external

forces (internal forces were generated using simple shape priors, *i.e.* shape memory associated with simplex meshes). The results of a large number of segmentation methods applied on the same public database have been published after the MICCAI 2007 Grand Challenge workshop [26]. Though our results lie in the same range of these methods (*i.e.* ASSD = 2.24 mm $\in$ [0.8, 3.6], VOE = 12.99 % $\in$ [5.2, 14.3] and MSD = 25.74 mm $\in$ [15.7, 49.2]), our approach does not perform the best because our appearance prior is not combined with strong shape priors. The next step of this work would be to test our proposed MPAM when combined with robust shape priors, *e.g.* using a statistical analysis of shape based on PCA [27, 43].

[1] Alpert, C.J., Yao, S.Z., 1994. Spectral partitioning: The more eigenvectors, the better, in: Proc. ACM/IEEE Design Automation Conf, pp. 195–200.

[2] van Assen, H.C., Danilouchkine, M.G., Frangi, A.F., Ordas, S., Westenberg, J.J.M., Reiber, J.H.C., Lelieveldt, B.P.F., 2006. Spasm: A 3d-asm for segmentation of sparse and arbitrarily oriented cardiac mri data. Medical Image Analysis 10, 286–303.

[3] Balcan, M.F., Blum, A., Vempala, S., 2007. A theory of similarity functions for clustering. Technical Report. Carnegie Mellon University.

[4] Behiels, G., Maes, F., Vandermeulen, D., Suetens, P., 2002. Evaluation of image features and search strategies for segmentation of bone structures in radiographs using active shape models. Medical Image Analysis 6, 47–62.

[5] Bishop, C.M., 2007. Pattern recognition and machine learning. Springer-Verlag.

[6] Blum, A., 2007. A theory of similarity functions for learning and clustering, in: DS'07 - Proceedings of the 10th International Conference on Discovery Science, pp. 39–39.

[7] Brejl, M., Sonka, M., 2000. Object localization and border detection criteria design in edge-based image segmentation: automated learning from examples. IEEE Transactions on Medical Imaging 19, 973–985.

[8] de Bruijne, M., van Ginneken, B., Viergever, M.A., Niessen, W., 2004. Interactive segmentation of abdominal aortic aneurysms in cta images. Medical Image Analysis 8, 127–138.

[9] Buzug, T., Weese, J., 1998. Voxel-based similarity measures for medical image registration in radiological diagnosis and image guided surgery. Journal of computing and information technology 6, 165–179.

[10] Chung, F., 2011. Regional appearance modeling for deformable model-based image segmentation. Ph.d. thesis. Mines ParisTech.

[11] Chung, F., Delingette, H., 2009. Multimodal prior appearance models based on regional clustering of intensity profiles, in: Medical Image Computing and Computer-Assisted Intervention (MICCAI'09), Part II, Springer. pp. 1051–1058.

[12] Chung, F.R.K., 1996. Spectral Graph Theory (CBMS Regional Conference Series in Mathematics, No. 92). American Mathematical Society.

[13] Claussen, M., 1985. Estimation of the monin-obukhov similarity functions from a spectral model. Boundary-Layer Meteorology 33, 233–243.

[14] Cootes, T.F., Hill, A., Taylor, C.J., Haslam, J., 1993. The use of active shape models for locating structures in medical images, in: IPMI '93 - Proceedings of the 13th International Conference on Information Processing in Medical Imaging, Springer-Verlag. pp. 33–47.

[15] Cootes, T.F., Taylor, C.J., 1993. Active shape model search using local grey-level models: A quantitative evaluation, in: BMVC 1993 - Proceedings of the 4th British Machine Vision Conference, BMVA Press. pp. 639–648.

[16] Cootes, T.F., Taylor, C.J., 1994. Using grey-level models to improve active shape model search, in: ICPR 1994 - Proceedings of the 12th IAPR International Conference on Pattern Recognition, pp. 63–67.

[17] Cristinacce, D., Cootes, T., 2008. Automatic feature localisation with constrained local models. Pattern Recogn. 41, 3054–3067.

[18] Daugman, J., 1988. Complete discrete 2-D Gabor transform by neural networks for image analysis and compression. IEEE Trans. ASSP 36, 1169–1179.

[19] Delingette, H., 1998. Initialization of deformable models from 3d data, in: ICCV'98 - Proceedings of the 6th International Conference on Computer Vision, pp. 311–316.

[20] Durrleman, S., Pennec, X., Trouvé, A., Ayache, N., 2009. Statistical

models on sets of curves and surfaces based on currents. Med Image Anal 13, 793–808.

[21] Feremans, C., Labbe, M., Laporte, G., 2004. The generalized minimum spanning tree problem: Polyhedral analysis and branch-and-cut algorithm. Networks 43, 71–86.

[22] Fripp, J., Crozier, S., Warfield, S., Ourselin, S., 2005. Automatic initialization of 3d deformable models for cartilage segmentation, in: DICTA'05 - Proceedings of the Australian Conference on Digital Image Computing: Techniques and Applications.

[23] Gilles, B., 2007. Anatomical and kinematical modelling of the musculoskeletal system from MRI. Phd thesis. University of Geneva.

[24] Glaunes, J., 2005. Modeles déformables en appariement de formes. Transport par difféomorphismes de points, de mesures et de courants pour la comparaison de formes et l'anatomie numérique. Phd thesis. Université Paris 13.

[25] Golub, G.H., Loan, C.F.V., 1996. Matrix computations. Johns Hopkins University Press.

[26] Heimann, T., van Ginneken, B., Styner, M., Arzhaeva, Y., Aurich, V., Bauer, C., Beck, A., Becker, C., Beichel, R., Bekes, G., Bello, F., Binnig, G., Bischof, H., Bornik, A., Cashman, P., Chi, Y., Cordova, A., Dawant, B., Fidrich, M., Furst, J., Furukawa, D., Grenacher, L., Hornegger, J., Kainmuller, D., Kitney, R., Kobatake, H., Lamecker, H., Lange, T., Lee, J., Lennon, B., Li, R., Li, S., Meinzer, H.P., Nemeth, G., Raicu,

D., Rau, A.M., van Rikxoort, E., Rousson, M., Rusko, L., Saddi, K., Schmidt, G., Seghers, D., Shimizu, A., Slagmolen, P., Sorantin, E., Soza, G., Susomboon, R., Waite, J., Wimmer, A., Wolf, I., 2009. Comparison and evaluation of methods for liver segmentation from ct datasets. IEEE Transactions on Medical Imaging 28, 1251–1265.

[27] Heimann, T., Meinzer, H.P., 2009. Statistical shape models for 3d medical image segmentation: A review. Medical Image Analysis 13, 543–563.

[28] Jian, B., Vemuri, B.C., 2005. A robust algorithm for point set registration using mixture of gaussians, in: Proceedings of ICCV '05, pp. 1246–1251.

[29] Li, S., Zhu, L., Jiang, T., 2004. Active shape model segmentation using local edge structures and adaboost, in: in MIAR, 3150, pp. 121–128.

[30] Lorensen, W.E., Cline, H.E., 1987. Marching cubes: A high resolution 3d surface construction algorithm. ACM SIGGRAPH Computer Graphics 21, 163–169.

[31] Luxburg, U., 2007. A tutorial on spectral clustering. Statistics and Computing 17, 395–416.

[32] Malik, J., Belongie, S., Leung, T., Shi, J., 2001. Contour and texture analysis for image segmentation. Int. J. Comput. Vision 43, 7–27.

[33] Mansi, T., 2010. Image-based physiological and statistical models of the heart: Application to tetralogy of fallot. Phd thesis. Mines ParisTech.

[34] Meila, M., Shi, J., 2000. Learning segmentation by random walks, in: NIPS 2000 - Proceedings of the 14th Annual Conference on Neural Information Processing Systems, pp. 873–879.

[35] Mohar, B., 1997. Some applications of laplace eigenvalues of graphs, in: Graph Symmetry: Algebraic Methods and Applications, volume 497 of NATO ASI Series C, pp. 227–275.

[36] Montagnat, J., 1999. Modèles déformables pour la segmentation et la modélisation d'images médicales 3D et 4D. Phd thesis. Université de Nice-Sophia Antipolis.

[37] Ng, A.Y., Jordan, M.I., Weiss, Y., 2001. On spectral clustering: Analysis and an algorithm, in: Advances in Neural Information Processing Systems 14, MIT Press. pp. 849–856.

[38] Penney, G., Weese, J., Little, J., Desmedt, P., Hill, D., Hawkes, D., 1998. A comparison of similarity measures for use in 2d-3d medical image registration, in: MICCAI'98 - Proceedings of the 1st International Conference on Medical Image Computing and Computer Assisted Intervention, pp. 1153–1161.

[39] Roche, A., Malandain, G., Pennec, X., Ayache, N., 1998a. The correlation ratio as a new similarity measure for multimodal image registration, in: MICCAI'98 - Proceedings of the 1st International Conference on Medical Image Computing and Computer Assisted Intervention, pp. 1115–1124.

[40] Roche, A., Malandain, G., Pennec, X., Ayache, N., 1998b. Multimodal Image registration by maximization of the correlation ratio. Rapport de Recherche 3378. INRIA - Institut National de Recherche en Informatique et Automatique.

[41] Savage, C., 1997. A survey of combinatorial gray codes. SIAM Review 39, 605–629.

[42] Schäfer, J., Strimmer, K., 2005. A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. Statistical Applications in Genetics and Molecular Biology 4.

[43] Schmid, J., Magnenat-Thalmann, N., 2008. Mri bone segmentation using deformable models and shape priors, in: MICCAI 2008 - Proceedings of the 11th International Conference on Medical Image Computing and Computer Assisted Intervention, pp. 119–126.

[44] Shen, D., Zhan, Y., Davatzikos, C., 2003. Segmentation of prostate boundaries from ultrasound images using statistical shape model. IEEE Transactions on Medical Imaging 22, 539–551.

[45] Shi, J., Malik, J., 2000. Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 888–905.

[46] Vaillant, M., Glaunes, J., 2005. Surface matching via currents, in: IPMI 2005 - Proceedings of the International Conference on Information Processing in Medical Imaging, p. 381.