

ProbExplorer: Uncertainty-guided Exploration and Editing of Probabilistic Medical Image Segmentation

Ahmed Saad^{1,2}, Torsten Möller¹, and Ghassan Hamarneh²

¹Graphics, Usability, and Visualization (GrUVi) Lab, ²Medical Image Analysis Lab (MIAL)
School of Computing Science, Simon Fraser University, Canada

Abstract

In this paper, we develop an interactive analysis and visualization tool for probabilistic segmentation results in medical imaging. We provide a systematic approach to analyze, interact and highlight regions of segmentation uncertainty. We introduce a set of visual analysis widgets integrating different approaches to analyze multivariate probabilistic field data with direct volume rendering. We demonstrate the user's ability to identify suspicious regions (e.g. tumors) and correct the misclassification results using a novel uncertainty-based segmentation editing technique. We evaluate our system and demonstrate its usefulness in the context of static and time-varying medical imaging datasets.

Categories and Subject Descriptors (according to ACM CCS): Computer Graphics [I.3.3]: Picture/Image Generation—Display algorithms

1. Introduction

Medical image segmentation is the procedure by which medical images are partitioned into disjoint regions of homogeneous properties. In 3D magnetic resonance imaging (MRI) or computed tomography (CT) images, the homogeneous regions are those with similar anatomical information. In dynamic positron emission tomography (dPET) or dynamic single photon emission computed tomography (dSPECT) images, the homogeneous regions are those with similar functional behavior. Image segmentation is often the precursor to quantification and visualization, which in turn aids in statistical analysis, diagnosis, and treatment evaluation.

Traditionally, domain experts were required to manually segment images. Even in 2D, this is a time consuming process, and can rapidly become infeasible for vector and tensor field images. In recent years, a variety of semi- and fully automatic techniques have been developed to address the segmentation problem [OS01]. However, the current state-of-the-art approaches fall short of providing a “silver bullet” for medical image segmentation. The majority of the existing segmentation methods rely on and are sensitive to user-initialized seeds, contours, and/or setting of low level parameters. Existing segmentation algorithms are yet to achieve full automation while producing completely correct results,

and hence cannot be relied upon in clinical settings without user intervention. Also, most segmentation techniques are problem-specific, and all have limitations due to the different image degradation factors, such as low signal-to-noise ratio and partial volume effect (PVE). Therefore, the resulting segmentation is imperfect, requiring further expert editing, which is usually done manually relying largely on visual assessment. The manual editing becomes very difficult for vector or tensor based images.

In direct volume rendering, the transfer function plays the role of the image segmenter by assigning optical properties (color and opacity) to different regions. Several methods have been proposed for transfer function design [EHK*06]. The most common ones are based on exploring a 2D data-derived feature layout. Those features might be corrupted due to different image degradation factors. Thus feature extraction becomes an error-prone process. Further, most of those techniques are not scalable with regard to encoding multiple features or incorporating higher levels of knowledge. Another major concern about volume rendering is the absence of quantitative evaluation of the underlying classification, which, for example, prohibits the provision of volumetric tumor measurements. This limits the usability of rendering in quantitative diagnosis procedures. The challenge is

to take the results of the segmentation and present it to the expert user for approval or improvement in an interactive manner, while allowing quantitative analysis of the underlying segmentation results.

In this work, we focus on analyzing the results of probabilistic segmentation techniques that provide an uncertainty measure for each voxel instead of a crisp labeling. We propose “ProbExplorer” to analyze and visualize the probabilistic fields and provide intuitive means for the user to influence the segmentation outcome. We allow the user to investigate different tissue interaction scenarios (Sec. 3.3.2). By tissue interaction we mean that there are multiple classes interacting and competing for the label of a particular voxel. For example, having probabilities close to $1/C$; where C is the number of classes; makes it uncertain to which class should the voxel be assigned. In addition to visualizing the uncertainty in the datasets, we allow the user to interact with the underlying uncertainty and edit the segmentation result in a systematic manner (Sec. 3.4). We will show the usability and efficacy of our framework within the context of segmenting static and time-varying medical imaging datasets (Sec. 4). This tool will allow the research clinicians to explore the uncertainty in the resulting segmentation. In addition, medical image analysis researchers can use this tool to analyze the behavior of their probabilistic segmentation algorithms.

2. Related work

Three main forms of user input have been adopted for interactive segmentation [OS01] by specifying: (1) a nearby complete boundary that evolves to the desired boundary [KWT88, MT96], (2) a small set of voxels belonging to the desired boundary, followed by the application of minimal path techniques to extract the complete boundary [MB98, BM97], and (3) few voxels belonging to the core of different regions to overcome the limitation of the ill-defined boundaries [BVZ01, APB07, Gra06], followed by the classification of all unlabeled voxels. In this paper, the proposed framework does not depend on a particular segmentation technique as long as it provides probabilistic results such as Gaussian mixture model [ZBS01], random walker [Gra06] and a variant of graph cuts [KT08]. Unlike most of the interactive segmentation techniques, we don’t require the repeated execution of the algorithm in order to obtain better results, which is expensive.

Few papers tackled the segmentation editing problem [OS01]. This can be due to the fact that scientific publications on segmentation mostly emphasize the automatic part while necessary manual corrections are considered flaws in the automated process. Kang et al. [KEK04] introduced a set of 3D segmentation editing tools based on morphological operators for hole filling and surface editing. The introduced tools require a good segmentation to start with. Grady & Funka-Lea [GFL06] formulated the editing task as an energy minimization problem. The user-supplied seeds, pre-

segmentation and image content all impact the segmentation. Our proposed approach is different from those methods in that we operate on the probabilistic fields instead of the crisp labeling. We leverage the uncertainty encoded into the segmentation results to highlight those problematic regions that need more attention from the user, as well as provide an uncertainty-driven segmentation editing scheme that is not possible using only the crisp labeling.

Different methods have been proposed for transfer function design in volume rendering [EHK*06]. For 3D medical volumes, data-driven features such as scalar, gradient values [KKH02], curvature [KWTM03], spatial information [RBS05, LLY06] are represented as multi-dimensional histograms. Also, transfer functions play an important role for visualizing time-varying datasets [JKM01]. The user is required to identify different regions in the feature space which might be problematic with noisy data or complex anatomical structures. Tzeng & Ma [TM04] used ISODATA hierarchical clustering to classify volumetric data. Tzeng et al. [TLM05] incorporated an artificial neural network and a support vector machine as a supervised learning mechanism for classification. As a supervised learning algorithm, only a small number of voxels was used for training and all the remaining voxels were used as a test set. Thus, a longer interaction time is needed for complex structures. Editing the classification result in all of those algorithms is based on the user knowledge of the data alone with no guidance to enhance the classification result more quickly.

Uncertainty visualization is considered one of the top visualization research challenges [Joh04]. Different visual mapping methods have been proposed to convey the uncertainty information such as glyphs [PWL97, WPL96], opacity [DKLP02], color [Hen03], texture [RLBS03], and surface displacement [GR04]. Few research papers tackled the problem of visualizing the classification uncertainty given a probabilistic segmentation result. Kniss et al. [KUS*05] proposed a Bayesian risk minimization framework, in which the final classification can be altered by changing the risk of each class instead of applying the maximum-a-posteriori (MAP) estimate directly on the probabilistic result. Lundström et al. [LLPY07] applied different animation schemes to convey the uncertainty at each voxel. Although, we consider the methods of Kniss et al. [KUS*05] and Lundström et al. [LLPY07] to be the closest work to our contribution, ours is different in multiple aspects: instead of only trying to convey the uncertainty information in the rendered image, we provide a set of novel multivariate visual analysis widgets (e.g. graphs, spreadsheets, and 2D histograms) to summarize the underlying multivariate probabilistic field and their associated semantic information; we allow the user to highlight those probabilistic regions in 2D and 3D; and finally, the user can change the final classification by using a novel uncertainty-based editing technique instead of being solely a user-driven process.

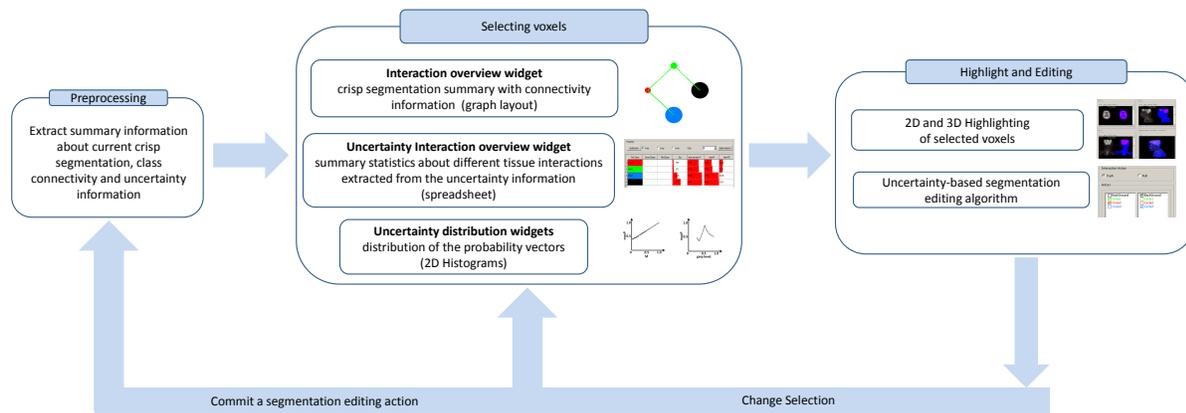


Figure 1: ProbExplorer consists of three main steps: preprocessing, voxel selection, and highlighting and editing.

3. Method

ProbExplorer consists of three main steps: preprocessing, selecting voxels, and highlighting and editing, that facilitate the exploration process (Fig. 1). In the preprocessing step, a number of quantities are extracted to represent the uncertainty in the segmentation results (Sec. 3.2). Then, in the voxel selection step (Sec. 3.3), the user specifies a region of interest (ROI) using a number of novel interaction widgets. These widgets are used to highlight different aspects of the multivariate probabilistic field in order to reveal a specific tissue interaction or a suspicious region. Finally, in the last highlighting and editing step (Sec. 3.4), the user applies a certain action over the selected voxels. The selected voxels are highlighted in a 2D slice and a 3D image. Further, the user corrects any misclassification by editing the probability field. This exploration process is iterative (Fig. 1), where the user can highlight or edit different regions multiple times.

3.1. Synthetic example

We introduce a synthetic example that simulates the common image generation model in medical imaging [BVZ01]. A grey-level image of size 128×128 consists of three circular materials representing three materials in addition to the background. We start with a piece-wise constant model in each region with the following grey-level values (30, 60, 90) while the background receives the grey-level of zero (Fig. 2(a)). We blur the image with a rotationally symmetric Gaussian lowpass filter of size 30 voxels with standard deviation 3, (Fig. 2(b)). Gaussian noise is added with a variance of 4, (Fig. 2(c)). Further, we add two smaller circles which represent two types of suspicious regions (e.g. different types of tumors) with grey-levels 17 and 100, respectively (Fig. 2(d)). We segment this image with a mixture of four Gaussians centered at the known means of the four main regions with variance of 56 to account for the PVE. Fig. 3(a) shows the unnormalized mixture of Gaussians used in the segmentation process, where the x-axis represents the

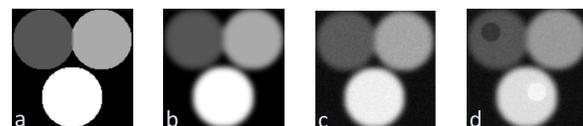


Figure 2: Synthetic example. a) piecewise constant image, b) blurred with a Gaussian kernel, c) Gaussian noise added, d) two-suspicious regions added.

grey-level value and the y-axis is the probability for each region marked with a distinct color. Fig. 3(b) shows the normalized version of the mixture when the probabilities sum to 1. The normalized probabilities will be the actual output of a probabilistic segmentation algorithm and the input for our tool. The intensity values of the two suspicious regions are shown as magenta and orange bars, respectively. Note that the intensity range of one of those suspicious regions falls in-between the intensity ranges of two other materials, whereas the intensities of the other outlier region falls near the tail of the distribution (Fig. 3(a)) and (Fig. 3(b)).

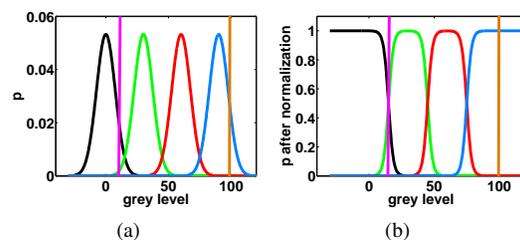


Figure 3: Mixture of Gaussians used to segment the image in Fig. 2(d). a) unnormalized mixture of Gaussians, b) normalized mixture of Gaussians. The magenta and orange bars represent the intensity values of the two small suspicious, tumor-like regions.

Fig. 4(a) shows the ground truth segmentation. The mix-

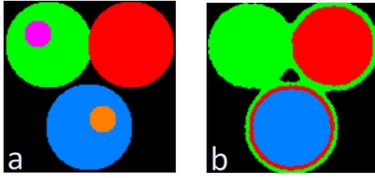


Figure 4: Segmentation of synthetic example. a) ground truth segmentation, b) crisp segmentation result showing different misclassification artifacts as well as inability to highlight the suspicious regions.

ture of Gaussians based segmentation gives erroneous results shown in Fig. 4(b) when applying the MAP principle on the resulting probabilistic segmentation. By analyzing the results shown in Fig. 4(b), we can identify different artifacts that are not specific to this example but rather common in medical imaging in general. The first artifact is the misclassification which appears as the green strip around the red circle and also the green and red strips around the blue circle. We need to allow the user to identify those implausible voxels and replace them with correct ones. For example, around the blue circle, we need to reclassify the green and red voxels to black (background) or blue. We will show that an uncertainty-based segmentation editing will allow us to do the reclassification operation in an intuitive way.

Another artifact is the misclassification of the two small circles. Note that we used only four regions during the segmentation which mimics the fact that during the segmentation, we usually have a-priori knowledge only about the expected normal regions (e.g. in our case three circles plus the background). This applies especially to common automatic segmentation techniques that rely on building intensity and shape priors from datasets of healthy subjects. In general, the suspicious regions are difficult to identify since they come from more challenging pathological cases. Pathological cases usually do not have well-defined models. The small magenta circle in Fig. 4(a) has a mean intensity value of 17 which falls in-between the intensity ranges of both the green and black regions even though it does not lie along the boundary between these two regions in the image domain Fig. 3(a). The small orange circle in Fig. 4(a) has a mean intensity value of 100 which lies in the intensity range of the tail of the probability distribution in Fig. 3(a). This region is uncertain with respect to the blue distribution but it is not similar to any other class considered in the segmentation. This leads to the conclusion that this region is considered very certain after the probability normalization as shown in Fig. 3(b). This necessitates a way to isolate this region by analyzing the probability field with additional information derived from the raw data.

3.2. Preprocessing

We assume that we have a probabilistic segmentation technique that produces a probabilistic vector field $P(\mathbf{x}) = [P_1(\mathbf{x}), P_2(\mathbf{x}), \dots, P_C(\mathbf{x})]$ where \mathbf{x} is a spatial location in \mathbb{R}^3 (i.e. $\mathbf{x} = [x \ y \ z]$) and $P_i(\mathbf{x})$ is the probability of the voxel at

location \mathbf{x} belonging to class i out of C classes such that $\sum_{i=1}^C P_i(\mathbf{x}) = 1$. The traditional procedure to obtain a crisp classification from a probabilistic result is by applying the MAP Bayesian principle [DHS00], where we assign a voxel located at \mathbf{x} to the class i with the maximum probability as the first best guess (FBG). Formally

$$P_{FBG}(\mathbf{x}) = \max_{i \in \{1..C\}} P_i(\mathbf{x}) \quad (1)$$

where $FBG = \text{argmax}_{i \in \{1..C\}} P_i(\mathbf{x})$. Note that throughout the paper, we interchangeably refer to P_{FBG} as maxP . In order to study regions of uncertainty, we study the *two-way interaction* at a point \mathbf{x} , where two classes are competing for the labeling of that voxel. This is defined by

$$M(\mathbf{x}) = P_{FBG}(\mathbf{x}) - P_{SBG}(\mathbf{x}) \quad (2)$$

where $P_{SBG}(\mathbf{x}) = \max_{i \in \{1..C\} \setminus \{FBG\}} P_i(\mathbf{x})$ is the second best guess (SBG) probability entry. Further, regions where more than two materials compete can also be studied. For example, the *three-way interaction*

$$M_{23}(\mathbf{x}) = P_{SBG}(\mathbf{x}) - P_{TBG}(\mathbf{x}) \quad (3)$$

where $P_{TBG}(\mathbf{x}) = \max_{i \in \{1..C\} \setminus \{FBG, SBG\}} P_i(\mathbf{x})$ is the third best guess (TBG) probability entry. In this paper, we did not consider any interaction beyond the three-way interaction but the underlying visual exploration framework is easily extensible to higher order interactions if needed.

A common assumption in medical image segmentation is that the image consists of piecewise homogeneous regions. Typically, this assumption is valid only in the tissue cores and not along their boundaries because of the PVE that causes the overlap between two or more materials. In order to properly analyze the tissue cores, we incorporate a Euclidean distance transform map D . This map is obtained by calculating the Euclidean distance from each voxel to the nearest boundary for each class i according to the MAP principle. D is zero along the boundary and increases gradually to a maximum, normalized to one, at the core.

3.3. Selecting voxels

In the following subsections, we introduce a set of widgets that highlight the uncertainty in the data and therefore allow a user-guided improvement of the segmentation results. In order to localize the effect of the analysis, the user has the ability to specify an axis-aligned ROI using sliders.

3.3.1. Interaction overview widget

The purpose of this widget is to give the user a summary of the segmentation inside the ROI. It shows a 2D layout of the different classes and their connectivity. It consists of an undirected weighted graph, where each node represents a specific class shown in a specific color associated with the class label. The size of each node is proportional to the number of voxels belonging to that class according to the MAP principle. The edges represent the connectivity inside the ROI

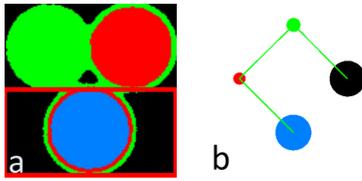


Figure 5: Interaction overview widget. a) ROI selection demarcated by a red border; b) interaction overview widget. Each node represents a specific class with size proportional to the number of voxels in that class in the image. The edges represent the connectivity between classes.

between different classes. The thickness of each edge is proportional to the strength of the spatial connectivity between these two classes. The connectivity strength is measured by the number of neighboring voxels between the two classes.

The user has the ability to select specific nodes for further analysis. For example, the user can select implausible voxel labels in the ROI by selecting the appropriate nodes in this widget. Fig. 5(a) shows the selection of the ROI for the lower half of the image. Fig. 5(b) shows the interaction overview of this ROI. It shows that the main (largest size) classes are the background (black) and the blue circle. It also shows that they are not directly connected to each other but rather through two other regions represented by the smaller red and green circles. This connectivity is a direct consequence of the PVE, where implausible regions (green and red) appear due to the overlapping tissues corresponding to the black and blue circles. Later, we will show how we can select the implausible regions and correctly relabel them.

3.3.2. Uncertainty interaction overview widget

The purpose of this widget is to give the user a summary of the different interactions embedded into the probabilistic information. We used a spread-sheet to represent the uncertainty interaction. Each row represents a specific interaction (e.g. row 5 in Fig. 6 represents all the voxels that have red as the highest probability, green as the second best guess and blue as the third best guess). The first three columns represent the different classes as the first guess, second guess and third guess, respectively. The remaining columns represent summary statistics about this particular interaction. The fourth column displays the size, i.e. the number of voxels contained in this interaction. The fifth column (mean maxP) represents the mean value of the maximum probability (1) in this region. The sixth and seventh columns represent the mean M (2) and mean M_{23} (3) values, respectively. Each value is represented visually by a color bar as well as a numerical value. All the columns can be sorted by clicking on the column headings. Multiple-row selection is allowed to highlight multiple regions. The user has the ability to select specific types of interaction that involve a particular class (Fig. 7(a)). Given the ROI shown in Fig. 5, Fig. 7(b) shows the result of sorting all one-way interactions in an ascending order by the mean maxP (fifth column). It shows that red

and green circles have lower values as well as being smaller in size. Given our prior knowledge, this might be an indication for misclassification. Indeed, this results from the PVE in this case.

Filter Guess	Second Guess	Third Guess	Size	mean maxP (1)	mean M (2)	mean M ₂₃ (3)
Circle3	Circle2	Circle1	2631	0.970	0.940	0.600
Circle3	Circle2	Circle1	2631	0.970	0.940	0.600
Circle3	Circle1	BackGround	040	0.534	0.068	0.04
Circle3	Circle1	Circle3	2235	0.959	0.880	0.658
Circle3	Circle1	Circle3	2275	0.952	0.866	0.665
Circle3	Circle3	Circle1	1075	0.934	0.870	0.662
Circle3	Circle3	Circle1	1075	0.934	0.870	0.662
Circle3	BackGround	Circle3	3207	0.904	0.808	0.905
Circle3	BackGround	Circle3	3207	0.904	0.808	0.905
Circle3	Circle2	BackGround	1359	0.896	0.725	1.109
Circle3	Circle2	Circle3	026	0.524	0.052	0.04
Circle3	Circle2	Circle3	1385	0.880	0.762	1.116
Circle3	Circle2	Circle3	4592	0.896	0.794	1.101
BackGround	Circle1	Circle2	5811	0.964	0.927	0.606
BackGround	Circle1	Circle2	5811	0.964	0.927	0.606
BackGround	Circle1	Circle2	5811	0.964	0.927	0.606

Figure 6: The uncertainty interaction overview widget represents different tissue interactions with their summary statistics extracted from the probabilistic information. Each row represents a specific interaction. a) 1-way (unary) interaction where blue marks FBG, b) 2-way interaction where red marks FBG and green marks SBG, c) 3-way interaction where green marks FBG, black marks SBG and red marks TBG, d) row filtering using a specific tissue interaction (1-way, 2-way, and 3-way), e) row filtering using a specific class, f) summary statistics, such as number of voxels in that interaction, mean maxP, mean M, and mean M_{23} .

3.3.3. Uncertainty distribution widget

The uncertainty interaction overview widget, in the previous section, gives summary statistics over different interaction regions but it does not give a detailed overview of the distribution of the probability vectors inside the selected region. The detailed view is important to avoid missing small anomalies. In order to obtain that detailed view, we propose two different widgets. The first one is a 2D histogram between maxP (1) and M (2). This widget shows the distribution of the probability vectors to help differentiating the voxels with one dominant class vs. voxels with multiple interacting classes.

The motivation for this is demonstrated in Fig. 8 (note that $\max P = P_{FBG}$ by definition). The case when we have only one dominant class defined by $P_{FBG} = 1.0$ and all other probabilities are equal to zero is shown as point A. A two-way interaction is characterized by $1/2 \leq P_{FBG} \leq 1$ and $0 \leq P_{SBG} \leq 1/2$ and all other probabilities are zero. Alternatively, using $P_{FBG} + P_{SBG} = 1.0$ and substituting (2) yields $P_{FBG} = M/2 + 1/2$. This is shown as a blue line in the maxP-M graph in Fig. 8, connecting point A and B. Similarly, for a three-way interaction, we start with $P_{FBG} + P_{SBG} + P_{TBG} = 1$, which yields $P_{FBG} \leq M/2 + 1/2$ (line AB) after applying (2) once and $P_{FBG} \geq 2M/3 + 1/3$ (line AC) after applying (2) twice. For a constant P_{TBG} (e.g.

$P_{TBG} = 0.15$ for the red line in Fig. 8) we get the equality $P_{FBG} = M/2 + 1/2 - P_{TBG}/2$.

First Guess	Second Guess	Third Guess	Size	mean maximum P	mean M	mean M2
Circle3	Circle2		2631	0.970	0.625	0.030
Circle2	Circle1		2275	0.932	0.966	0.065
Circle2	Circle3		1075	0.934	0.870	0.062
Circle1	Circle2		1385	0.890	0.762	0.116

(a)

First Guess	Second Guess	Third Guess	Size	mean maximum P	mean M	mean M2
Circle2			562	0.730	0.485	0.246
Circle1			835	0.762	0.570	0.208
Circle3			2631	0.970	0.940	0.030
Background			4843	0.960	0.961	0.020

(b)

Figure 7: Uncertainty interaction overview filtering and sorting for the ROI shown in Fig. 5(a). a) uncertainty interaction overview showing only two-way interactions for the red material, b) sorting all one-way interactions in an ascending order by the mean maxP (fifth column).

We use this widget to analyze the green circle in Fig. 9. Fig. 9(a) shows the ROI selection, Fig. 9(b) shows the normalized distance transform map that corresponds to the green material. In order to restrict the analysis to the core of the region, we threshold the distance transform > 0.1 as shown in Fig. 9(c). This threshold will be used throughout the paper when needed. Note that the other classes are mapped using a grey-level map to maintain the context of those regions. Fig. 9(d) shows the 2D log normalized histogram between maxP and M, It shows that most voxels are constrained to (1,1) (i.e. the most reddish region) trailed by a line. We use a selection tool (Fig. 9(e)) to highlight this region which is shown in Fig. 9(f). This reveals the suspicious region which was not available in the crisp segmentation case. We used larger circular color glyphs for the non-zero entries of the histogram for illustration purposes but a normal grey-level 2D histogram will be used in Sec. 4.

The suspicious region in the lower half of the image (mean value = 100) is different as it does not exist in the overlapping between two classes but rather in the tail of the distribution of the blue circle (Fig. 3(a)). After normalizing the distribution (Fig. 3(b)), it becomes very certain to belong to the blue class. This case is rather difficult to detect as the provided uncertainty information by the segmentation technique is already normalized. Hence, we need another feature besides the uncertainty information to detect those regions. In our case, we use the grey-level value for static images and the grey-level value for an average time frame for time-varying datasets. We construct a 2D log normalized histogram between maxP and the grey-level value. Fig. 10(d) shows an example of such a histogram. It shows two peaks near the line of $maxP = 1$, the more reddish (more voxels) peak corresponds to the majority of the voxels inside the region while the less reddish region is selected to highlight the suspicious region (Fig. 10(f)).

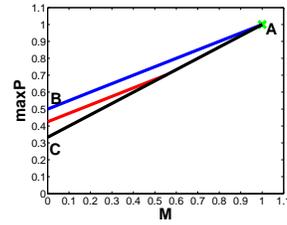


Figure 8: MaxP vs. M model. The green cross on the top right represents the one-way interaction. The blue line represents the two-way interaction. The red line represent the three-way interaction for a constant third best guess.

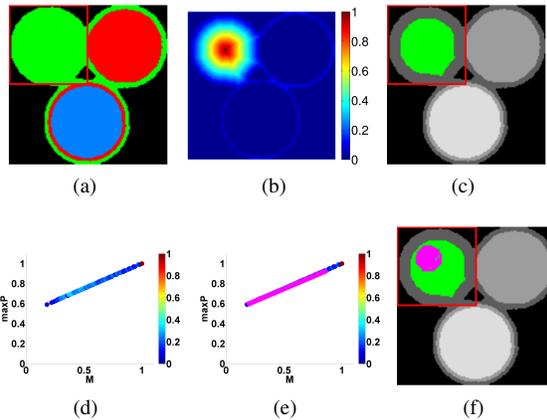


Figure 9: Suspicious region highlighting. a) ROI over the upper left quarter, b) normalized distance transform map with respect to the green material, c) thresholding the distance transform map to obtain the core of the organ, d) 2D log normalized histogram of maxP vs M representing the normalized number of voxels with a colormap, e) selection in magenta, f) suspicious region pops out in magenta.

3.4. Highlighting and editing

In the previous section, a subset of voxels is selected using our visual analysis widgets. There are two main actions we can apply over this subset: we can highlight them visually or we can edit their classification.

For highlighting in 2D, we use the common three slice views, one for each axis of the 3D data (see Fig. 13(a) where each view is split into a raw data view and the current crisp segmentation results). For segmentation results, we use a distinct color for each class. When a particular tissue interaction is selected using one of the overview widgets introduced in Sec. 3.3.1 or Sec. 3.3.2, we move the unselected classes to a grey-level map to maintain the context as shown in Fig. 9(c). When the user selects a subregion in the 2D feature layout introduced in Sec. 3.3.3, we use a highlight color for those voxels (Fig. 9(f)). In 3D, we use volume rendering similar to [KUS*05] with the same color mapping as for 2D and an opacity map determined by simple sliders. Fig. 13(c) shows a 3D example of a highlighted tumor.

Due to the large anatomical and functional variability between individuals, segmentation results are usually not perfect and need further editing to allow accurate analysis. Usually segmentation editing is done manually or utilizing morphological operators that act only on the crisp results. In this paper, we provide a simple interface to allow segmentation editing by operating on the probability field. We tackle two main problems that are common in segmentation, over-segmentation and under-segmentation. In over-segmentation, too many voxels are assigned to a particular class whereas in under-segmentation, too few voxels are assigned to a particular class. In order to overcome these problems, we provide two actions *push* and *pull* that the user can apply over the probabilistic field. The push action moves the voxels from a set of classes (source class set SCS) to another set of classes (destination class set DCS). The moving of voxels is done by interchanging the probability entries for the affected classes. The priority of replacing a particular voxel with another depends on the probability vector associated with this voxel. The pull action is the reverse of the push action. Details are shown in Alg. 1. Our approach requires only one slider for the editing process versus one slider per class [KUS*05]. We enable changing of the decision boundary between multiple classes simultaneously.

We apply our editing algorithm to remove the red and green strips surrounding the blue circle. Fig. 11(b) shows the selection of the two regions in the interaction overview widget. Fig. 11(c) shows selection of the push action as well as the SCS and DCS. Fig. 11(d) shows how the user changes the δ threshold to act as an input for Alg. 1. Fig. 11(e) and Fig. 11(f) show the result of changing δ to 0.7 and 1.0 respectively removing the red and green strips completely. The user has the ability to commit that change and start a new editing process.

Algorithm 1 uncertainty-based segmentation editing

Input: (i) Probabilistic segmentation $P(\mathbf{x})$, (ii) uncertainty threshold $\delta \in [0, 1]$, (iii) Required action (push or pull), (iv) ROI

Result: Modified probabilistic segmentation result $\hat{P}(\mathbf{x})$.

```

for all locations  $\mathbf{x} \in \text{ROI}$  do
    if required action = push then
        if  $P_{FBG}(\mathbf{x}) \in \text{SCS}$  then
            if  $\min_{i^*, j^*} |P_{i \in \text{SCS}} - P_{j \in \text{DCS}}| \leq \delta$  then
                swap( $P_{i^*}, P_{j^*}$ )
            end if
        end if
    else if required action = pull then
        if  $P_{FBG}(\mathbf{x}) \in \text{DCS}$  then
            if  $\min_{i^*, j^*} |P_{i \in \text{SCS}} - P_{j \in \text{DCS}}| \leq \delta$  then
                swap( $P_{i^*}, P_{j^*}$ )
            end if
        end if
    end if
end for
    
```

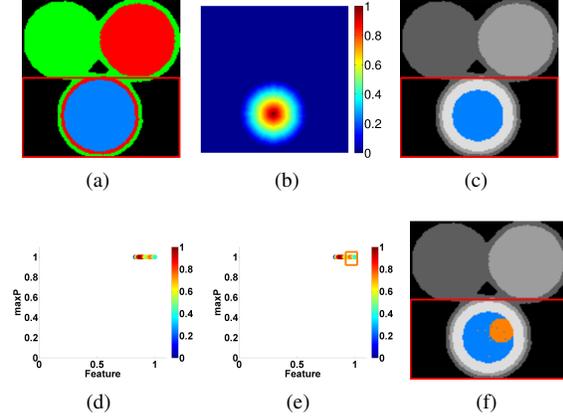


Figure 10: Suspicious region highlighting. a) ROI selected, b) normalized distance transform map with respect to the blue material, c) thresholding the distance transform map to obtain the core of the organ, d) 2D log normalized histogram between maxP vs. grey-level representing the normalized number of voxels with a colormap, e) selection in orange, f) suspicious region pops out in orange.

4. Results

In this section, we demonstrate the effectiveness of our framework to analyze and visualize different probabilistic segmentations of real static anatomical and time-varying functional datasets. We will show how we can highlight suspicious regions as well as misclassifications.

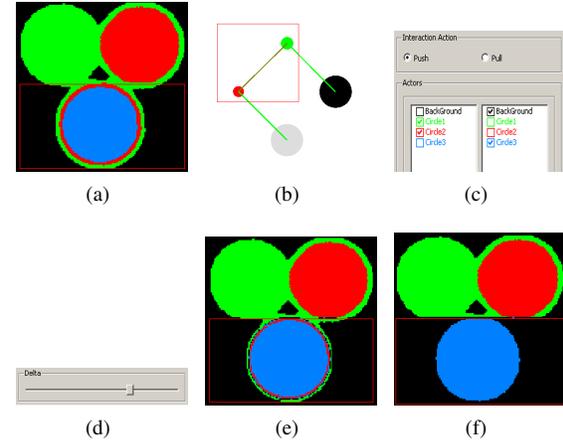


Figure 11: Uncertainty-based segmentation editing. a) ROI over the lower half, b) selecting the implausible regions, c) source class set (SCS) on the left and destination class set (DCS) on the right with push action, d) changing the editing threshold δ , e) $\delta = 0.7$, f) $\delta = 1.0$.

A renal dynamic SPECT study results in a time series of 3D images. From these images, the clinician's goal is to extract information about the spatio-temporal behavior of a radioactive tracer (e.g. $^{99m}\text{technetium}$ –

diethylenetriaminepentaacetic), which in turn is used to assess the renal system function. We used a 4D image of size $64 \times 64 \times 32$ with 48 time steps with an isotropic voxel size of $(2 \text{ mm})^3$. A 2D coronal slice is shown in Fig. 12(a). A mixture of Gaussians [ZBS01] combined with the random walker [Gra06] technique is used to segment the data into four regions (background, abdomen, left kidney, and right kidney). Analyzing the uncertainty interaction overview information for two-way interactions reveals (third row of Fig. 12(b)) that the SBG for the right kidney is the left kidney. This is not surprising as the two kidneys functions should behave similar in healthy individuals. The first two rows show that the SBG for the left kidney is split between the right kidney and the abdomen. Fig. 12(d) shows the selection of row 2 in the ROI shown in Fig. 12(c). Fig. 12(e) shows the selection of row 1 which reveals the lower third of the left kidney. Our clinical collaborators confirmed that there is indeed abnormality.

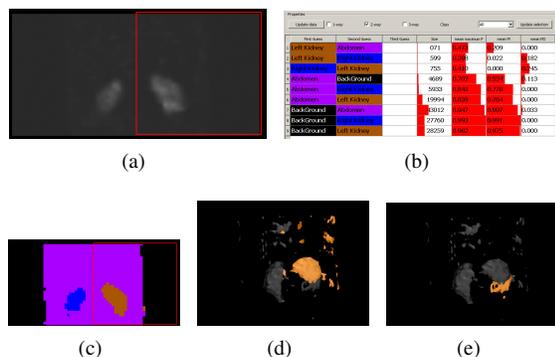


Figure 12: Dynamic SPECT case study with abnormal renal behavior in the lower third of the left kidney. a) 2D coronal slice out of the 4D dataset, b) uncertainty interaction overview widget, c) ROI over the left kidney, d) voxels that are labeled left kidney and SBG is right kidney that correspond to row 2, e) voxels that are labeled left kidney and SBG is abdomen that correspond to row 3.

In dPET imaging, a series of 3D images are reconstructed from list-mode data obtained by Gamma coincidence detectors. Kinetic modeling is the process of applying mathematical models to analyze the temporal tracer activity, in order to extract clinically or experimentally relevant information. We will analyze the probabilistic segmentation result obtained from the application of a kinetic modeling based K-means algorithm [SSHM07]. The 4D $[^{11}\text{C}]$ Raclopride dPET dataset size is $128 \times 128 \times 63$ with 26 time steps and voxel size of $2.11 \times 2.11 \times 2.42 \text{ mm}^3$. The dataset is segmented into six regions (background, skull, white matter, grey matter, cerebellum and putamen). It is clear from Fig. 14(a) that the dataset is suffering from low signal-to-noise ratio as well as severe PVE. This results in an oversegmentation of the putamen which is crucial for diagnosis of Parkinson’s disease. We use our editing technique to correct the misclassification. We need to apply a push action where the SCS con-

sists of the putamen only. In order to determine the DCS, we study the two-way interaction rows for the putamen (green) in Fig. 14(d). It is clear from row 4 that the white matter is the SBG, the most confident (mean maxP = 0.4 and mean M = 0.22) and it has the largest size (2310). Fig. 14(e) shows a highlight of that region with most voxels correctly classified. Due to the space limitation, we can not show other interactions, however, all are located in the upper part of the brain or near the cerebellum away from the putamen. We put the background, skull, grey matter, and cerebellum voxels into the DSC, then apply the push action. Due to the fact that some voxels from the vascular structure of the brain have similar behavior to the putamen, it was trivial to exclude them using the ROI widget. The result of the second editing step is shown in Fig. 14(f) which was verified clinically.

A static $[^{18}\text{F}]$ FDG-PET data from Osirix (<http://pubimage.hcuge.ch/>) is used in this experiment. The image size is $73 \times 87 \times 73$ with an isotropic voxel size of $(1 \text{ mm})^3$. A mixture of Gaussians [ZBS01] combined with the random walker [Gra06] technique is used to segment the data into three regions (background, active, and inactive). It is normal for this tracer to travel to the brain, so we will focus our analysis on the neck area (Fig. 13(a)). By analyzing the two-way interactions in Fig. 13(b), we note that most of the inactive voxels have background as the SBG but there are a number of voxels that have the active region as the SBG with a mean maxP = 0.6. Selecting this row (2) (Fig. 13(c)), clearly shows a tumor in the neck area. Further analysis of the tumor shows interesting patterns in the 2D uncertainty layout widget. We note two lines in the maxP-M graph, as well as a Gaussian like shape in the maxP-grey-level graph. Selecting those patterns demonstrate that the tumor is not homogeneous but rather decomposed into two (internal and external) functionally distinct regions. This conclusion was also verified clinically.

Our last study shows a simulated probabilistic field from the brainweb database (<http://www.bic.mni.mcgill.ca/brainweb/>). It is a brain MRI dataset with multiple sclerosis lesions. In addition to a main line along the two-way interaction line, Fig. 15(a) shows another stray line underneath. Fig. 15(b) and Fig. 15(c) show the white matter highlighted in a 2D slice and a 3D image, respectively. Highlighting the stray region (Fig. 15(d)) reveals the multiple sclerosis lesion in the white matter region in 2D (Fig. 15(e)) and in 3D (Fig. 15(f)).

5. User evaluation

Our clinical collaborators have already started using our software system to identify suspicious regions, as well as to correct segmentation misclassifications [HSC*09]. They showed how ProbExplorer can be used to achieve highly accurate segmentation from a very noisy dSPECT renal study, similar to the one shown in Fig. 12. The resulting segmentations have been incorporated into a novel dSPECT image reconstruction algorithm and have shown to improve the reconstruction [HSC*09]. Although ProbExplorer allowed our

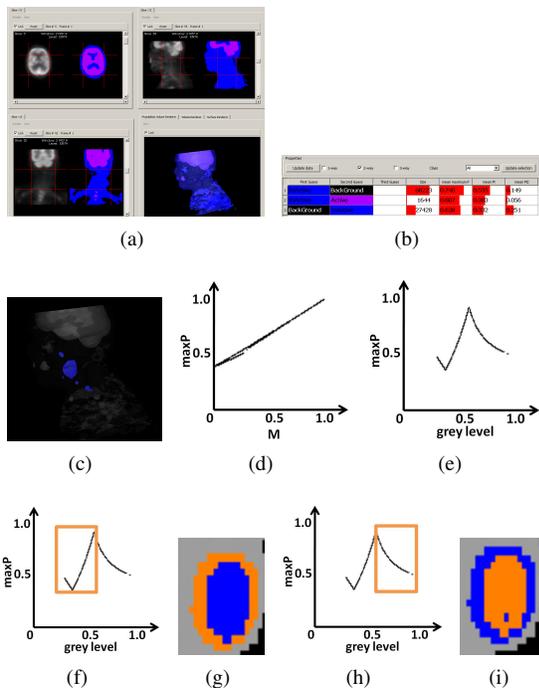


Figure 13: PET study showing a cervical tumor. a) ROI around the neck area, b) two-way interaction overview, c) voxels labeled inactive that have SBG active, d) M vs $maxP$, e) grey-level vs. $maxP$, f) selection, g) zoomed-in version of the outer shell of the tumor, h) selection, i) zoomed-in version of the inner core of the tumor.

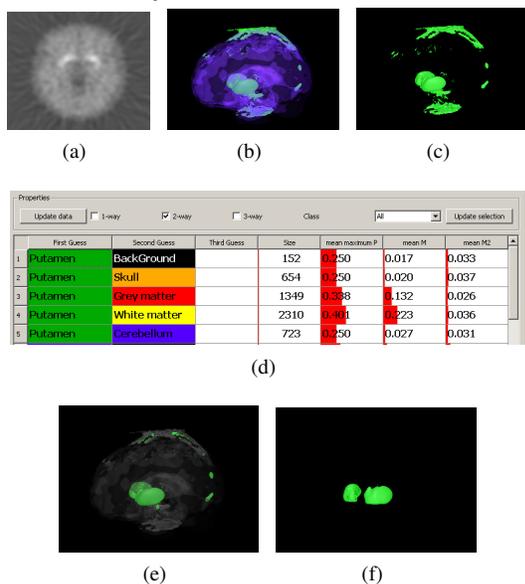


Figure 14: Dynamic PET case study with over-segmented putamen showing the efficiency of segmentation editing. a) 2D axial slice of the 4D dataset, b) a 3D image, c) over-segmented putamen, d) uncertainty interaction overview widget, e) voxels that have white matter as SBG, f) classification after two editing iterations.

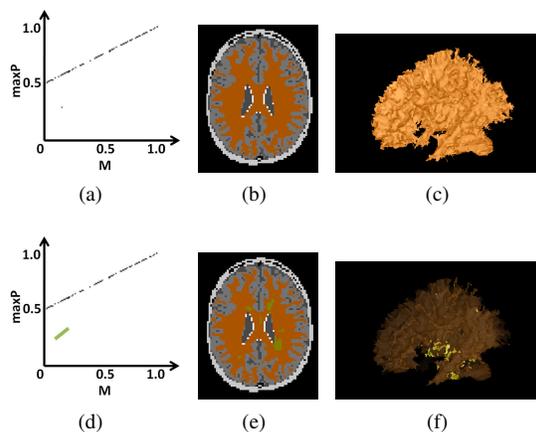


Figure 15: Simulated brain MRI data from the brainweb database multiple sclerosis lesion. a) $maxP$ vs. M of the white matter, b) white matter highlighted in a 2D slice, c) a 3D image of the white matter, d) selecting the outlier pattern from the main two-way interaction line, e) a 2D slice of the white matter with the lesion highlighted in green, f) a 3D image of the white matter with the lesion highlighted.

clinical collaborators to achieve accurate segmentation instead of performing manual editing, the tool would be more valuable if it could learn from previous segmentation edits. This way the user will not have to perform repeated editing actions when working with novel but similar datasets.

6. Conclusion and Discussion

In this paper, we presented ProbExplorer, a framework for the analysis and visualization of probabilistic segmentation results. We provided a number of visual data analysis widgets to reveal the different class interactions that are usually hidden by a simple MAP (i.e. crisp) visualization. We demonstrated the ease in which we can highlight suspicious regions which often correspond to easily missed pathological cases. Also the misclassification due to low signal-to-noise ratio and PVE can be edited efficiently with a novel uncertainty-based segmentation editing technique. We demonstrated the efficiency of the algorithm in the context of segmenting multiple simulated and real medical image datasets. Our uncertainty-based segmentation editing technique focused on changing the probability vector of a specific voxel without taking into account its neighboring voxels. We are working on a variation of our method that no longer assume spatial independence between voxels during interactive segmentation. Also, we plan to investigate the behavior of the resulting probabilistic results from different segmentation techniques (e.g. graph cuts and levelsets).

7. Acknowledgements

This work has been supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC). We would like to thank our clinical collaborators at the department of Radiology and Medical Physics, UBC, Canada for their valuable comments and suggestions.

References

- [APB07] ARMSTRONG C. J., PRICE B. L., BARRETT W. A.: Interactive segmentation of image volumes with live surface. *Journal of Computers And Graphics* 31 (2007), 212–229.
- [BM97] BARRETT W. A., MORTENSEN E. N.: Interactive live-wire boundary extraction. *Medical Image Analysis* 1 (1997), 331–341.
- [BVZ01] BOYKOV Y., VEKSLER O., ZABIH R.: Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 23, 11 (2001), 1222–1239.
- [DHS00] DUDA R. O., HART P. E., STORK D. G.: *Pattern classification*. Wiley-Interscience Publication, 2000.
- [DKLP02] DJURCILOV S., KIM K., LERMUSIAUX P., PANG A.: Visualizing scalar volumetric data with uncertainty. *Journal of Computers and Graphics* 26 (2002), 239–248.
- [EHK*06] ENGEL K., HADWIGER M., KNISS J. M., REZK-SALAMA C., WEISKOPF D.: *Real-time volume graphics*. A. K. Peters, Natick, MA, USA, 2006.
- [GFL06] GRADY L., FUNKA-LEA G.: An energy minimization approach to the data driven editing of presegmented images/volumes. In *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention* (2006), pp. 888–895.
- [GR04] GRIGORYAN G., RHEINGANS P.: Point-based probabilistic surfaces to show surface uncertainty. *IEEE Transaction on visualization and computer graphics* 10, 5 (2004), 564–573.
- [Gra06] GRADY L.: Random walks for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28 (2006), 1768–1783.
- [Hen03] HENGL T.: Visualization of uncertainty using the hsi colour model: computations with colours. In *Proceedings of the 7th International Conference on GeoComputation* (2003), pp. 8–17.
- [HSC*09] HUMPHRIES T., SAAD A., CELLER A., HAMARNEH G., MÖLLER T., TRUMMER M.: Segmentation-based regularization of dynamic spect reconstructions. In *IEEE Nuclear Science Symposium Conference Record (NSS/MIC)* (2009), pp. 2849–2852.
- [JKM01] JANKUN-KELLY T. J., MA K.-L.: A study of transfer function generation for time-varying volume data. In *Proceedings Volume Graphics Workshop* (2001), pp. 51–65.
- [Joh04] JOHNSON C.: Top scientific visualization research problems. *IEEE Transactions on Computer Graphics and Applications* 24, 4 (2004), 13–17.
- [KEK04] KANG Y., ENGELKE K., KALENDER W. A.: Interactive 3D editing tools for image segmentation. *Medical Image Analysis* 8 (2004), 35–46.
- [KKH02] KNISS J., KINDLMANN G., HANSEN C.: Multi-dimensional transfer functions for interactive volume rendering. *IEEE Transactions on visualization and computer graphics* 8, 3 (2002), 270–285.
- [KT08] KOHLI P., TORR P.: Measuring uncertainty in graph cut solutions. *Comput. Vis. Image Underst.* 112, 1 (2008), 30–38.
- [KUS*05] KNISS J. M., UITERT R. V., STEPHENS A., LI G.-S., TASDIZEN T., HANSEN C.: Statistically quantitative volume visualization. In *Proceedings of IEEE Visualization* (2005), pp. 287–294.
- [KWT88] KASS M., WITKIN A., TERZOPOULOS D.: Snakes: Active contour models. *International Journal of Computer Vision* 1, 4 (1988), 321–331.
- [KWTM03] KINDLMANN G. L., WHITAKER R. T., TASDIZEN T., MÖLLER T.: Curvature-based transfer functions for direct volume rendering: Methods and applications. In *Proceedings of IEEE Visualization* (2003), pp. 513–520.
- [LLPY07] LUNDSTRÖM C., LJUNG P., PERSSON A., YNNERMAN A.: Uncertainty visualization in medical volume rendering using probabilistic animation. *IEEE Transaction on visualization and computer graphics* 13, 6 (2007), 1648–1655.
- [LLY06] LUNDSTRÖM C., LJUNG P., YNNERMAN A.: Local histograms for design of transfer functions in direct volume rendering. *IEEE Transactions on visualization and computer graphics* 12, 6 (2006), 1570–1579.
- [MB98] MORTENSEN E. N., BARRETT W. A.: Interactive segmentation with intelligent scissors. *Graph. Models Image Process.* 60, 5 (1998), 349–384.
- [MT96] MCINERNEY T., TERZOPOULOS D.: Deformable models in medical image analysis: A survey. *Medical Image Analysis* 1, 2 (1996), 91–108.
- [OS01] OLABARRIAGA S., SMEULDERS A.: Interaction in the segmentation of medical images: A survey. *Medical Image Analysis* 5 (2001), 127–142.
- [PWL97] PANG A., WITTENBRINK C., LODHA S.: Approaches to uncertainty visualization. *The Visual Computer* 13, 8 (1997), 370–390.
- [RBS05] ROETTGER S., BAUER M., STAMMINGER M.: Spatialized transfer functions. In *Proceedings of IEEE/Eurographics Symposium on Visualization (EuroVis)* (2005), pp. 271–278.
- [RLBS03] RHODES P. J., LARAMEE R. S., BERGERON R. D., SPARR T. M.: Uncertainty visualization methods in isosurface rendering. In *Proceedings Eurographics, Short Papers* (2003), pp. 83–88.
- [SSH07] SAAD A., SMITH B., HAMARNEH G., MÖLLER T.: Simultaneous segmentation, kinetic parameter estimation, and uncertainty visualization of dynamic PET images. In *Lecture Notes in Computer Science, (MICCAI)* (2007), pp. 726–733.
- [TLM05] TZENG F.-Y., LUM E., MA K.-L.: An intelligent system approach to higher-dimensional classification of volume data. *IEEE Transactions on visualization and computer graphics* 11, 3 (2005), 273–284.
- [TM04] TZENG F.-Y., MA K.-L.: A cluster-space visual interface for arbitrary dimensional classification of volume data. In *Proceedings of Joint Eurographics-IEEE TVCG Symposium on Visualization* (2004), pp. 17–24.
- [WPL96] WITTENBRINK C. M., PANG A. T., LODHA S. K.: Glyphs for visualizing uncertainty in vector fields. *IEEE Transactions on visualization and computer graphics* 2, 3 (1996), 266–279.
- [ZBS01] ZHANG Y., BRADY M., SMITH S.: Segmentation of brain MR images through a hidden Markov random field model and the expectation-maximization algorithm. *IEEE Transactions on Medical Imaging* 20 (2001), 45–57.