Robust segmentation using non-parametric snakes with multiple cues for applications in radiation oncology

Jayashree Kalpathy-Cramer[1], Umut Ozertem[2], William Hersh[1], Martin Fuss[3], Deniz Erdogmus[4]

Departments of [1]Medical Informatics and Clinical Epidemiology, [3]Radiation Medicine, Oregon Health & Science University, Portland, OR; [2]Yahoo! Inc. Santa Clara CA; [4]Department of Electrical and Computer Engineering, Northeastern University, Boston, MA

ABSTRACT

Radiation therapy is one of the most effective treatments used in the treatment of about half of all people with cancer. A critical goal in radiation therapy is to deliver optimal radiation doses to the perceived tumor while sparing the surrounding healthy tissues. Radiation oncologists often manually delineate normal and diseased structures on 3D-CT scans, a time consuming task. We present a segmentation algorithm using non-parametric snakes and principal curves that can be used in an automatic or semi-supervised fashion. It provides fast segmentation that is robust with respect to noisy edges and does not require the user to optimize a variety of parameters, unlike many segmentation algorithms. It allows multiple cues to be incorporated easily for the purposes of estimating the edge probability density. These cues, including texture, intensity and shape priors, can be used simultaneously to delineate tumors and normal anatomy, thereby increasing the robustness of the algorithm. The notion of principal curves is used to interpolate between data points in sparse areas. We compare the results using a non-parametric snake technique with a gold standard consisting of manually delineated structures for tumors as well as normal organs.

Keywords: Segmentation, non-parametric snakes, principal curves, tumors.

1. Introduction

Cancer is one of the leading causes of death in the United States and in the world. Ionizing radiation can be used to kill cancer cells and shrink tumors to treat the cancer or relieve suffering [1]. Recent technological advances such as Intensity-modulated Radiation Therapy (IMRT) [1, 2] or 3-dimensional conformal radiation therapy allow radiation oncologists to tailor the dose profile in order to deliver appropriately high doses of radiation to the tumor while minimizing the dose delivered to nearby areas, resulting in fewer side effects and reducing toxicity. Thus, a critical aspect of the treatment is the planning stage during which the radiation oncologist determines the area to be treated, nearby organs at risk and the dose profile that will be delivered. During the planning stage for many solid tumors, multi-slice, three-dimensional computed tomography (3D CT) scans are obtained from the patient. Typically, the radiation oncologist manually delineates (or "contours") the tumor and the neighboring organs on each slice of the CT volume using a mouse or other pointing device, a time-consuming process.

Segmentation for medical applications is an active area of research. Since radiation oncologists routinely perform segmentations manually, an algorithm that can provide near real-time, robust segmentation with minimal supervision can have tremendous value in the clinical workflow. Segmentation is often based on color, intensity, textures and adjacency. Common image processing approaches to segmentation include those based on clustering, histograms, edge detection, region growing, level sets, and watershed. Model based segmentation approaches integrate prior knowledge about anatomical structures including their shapes, typical intensities and location, with image processing.

Although there are many algorithms available for automatic contouring, segmentation continues to be a challenge for many medical applications. Radiation oncologists often adjust the contours on CT slices using their knowledge of general anatomy, prior knowledge about the particular patient and data from other sources like Positron Emission Tomography (PET) scans. In addition, when tumors lie on the walls of organs or are present in organs with similar

Medical Imaging 2009: Image Processing, edited by Josien P. W. Pluim, Benoit M. Dawant,
Proc. of SPIE Vol. 7259, 72594S · © 2009 SPIE
CCC code: 1605-7422/09/$18 · doi: 10.1117/12.812712

density, the edge of the object can be occluded or hard to discern. These situations can degrade the performance of algorithms that rely on clear boundaries or sharp differences in pixel intensity.

Active contours or "snakes" [3] are a popular class of algorithms used for segmentation in medicine where the segmentation is defined as an energy minimization problem. Typically, the contours are subject to an internal and external energy and the contour minimizing the energy is defined in parametric terms. However, traditional snakes are sensitive to noise and can suffer from capture range issues. Their initialization must be close to boundary desired and their convergence can depend on the initial position. Gradient vector flow (GVF) snakes [4] solve many of these issues and are good at detecting shapes with boundary concavities. However, they are sensitive to the parameters of the algorithm and may need to be tuned for each application. They can also be quite slow as extracting the GVF field can be computationally expensive. Both these issues can be problematic for medical applications where the clinicians do not wish to tune algorithmic parameters and essentially real-time segmentation is required.

Non-parametric snakes provide a fast alternative by using a kernel density estimation (KDE) based approach that does not require the optimization of many parameters. In this paper, we use non-parametric snakes [5, 8] for segmentation of multi-slice volumes in near real-time. Our semi-supervised method incorporates the knowledge from the clinician along with the features of the image to provide a robust algorithm that can deal with noisy or somewhat occluded edges.

2. Methodology

In this paper, we present results on two-dimensional slices from 3D CT scans. However, the same approach can be used for volume segmentation in three dimensions. The overall approach starts with edge detection on the slice of interest. This is followed by a kernel density estimation based on the detected edge. Finally, the ridges of the edge KDE provide the object boundary. These ridges can be found using the fixed point algorithm [5] as well as using principal curves [7]. If available, initialization based on user input or from external sources is used.

For medical images, in some cases, the intensity provides sufficient information in order to estimate the edge, in other cases the edge is discerned on the basis of changes in texture while in other cases, both provide useful estimates of the edge. In the implementation used in this paper, we use multiple cues including intensity and texture to provide a good edge density estimate. As described in [8], we can also incorporate user input or prior knowledge or knowledge from another modality to improve this density estimate.

2.1 Edge detection:

Let I be the CT image volume in 3 dimensions such that $[x,y,z]^T$ is the location of a pixel in space and $I(x,y,z)$ is the pixel intensity of the image at that location. We consider a given slice, S, chosen by the user. We want to segment the object of interest (tumor or organ at risk) in this slice contained in the image $I(x,y)$ for a slice at a given z value.

The edge maps can be obtained using an edge detector with thresholds selected for the anatomical location of interest. These edge values can be binary or continuous.

The output of a binary edge detector can be stated as:

$$E(\mathbf{s}) = \begin{cases} E(\mathbf{s}) = 1 & \mathbf{s} \text{ is an edge pixel} \\ E(\mathbf{s}) = 0 & otherwise \end{cases}$$

Many algorithms that have been developed for edge detection for applications in image processing and computer vision. One of the most popular of these, the Canny edge detector, has been shown to be effective for noisy data. [9] The Canny edge detector was found to be useful in detecting edges for the images that we used for this study. Figure 1 shows CT slices of a lung tumor, lung and a liver tumor with the output of a Canny edge detector (using the pixel intensity) in blue. Note that the choice of thresholds for the Canny edge detector is critical and should be chosen depending on the organs to be segmented. Although these thresholds can be selected in an ad-hoc manner, we created an algorithm to automatically "learn" these thresholds based on training data consisting of manually contoured edges for the organs. The technique is briefly described in section 2.4.

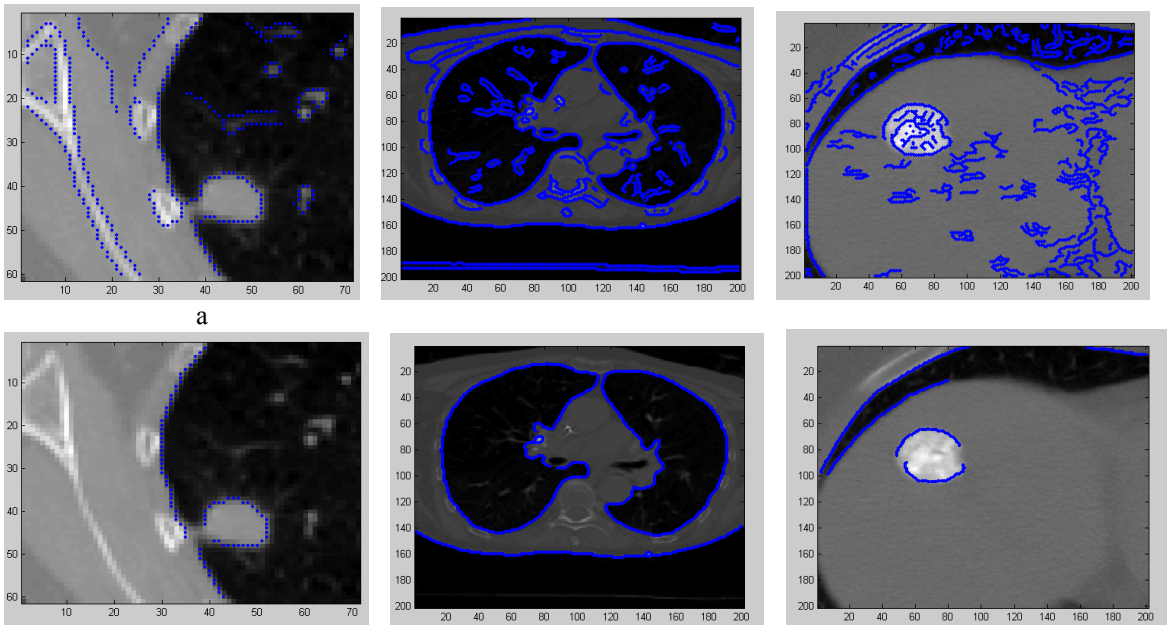

Figure 1 Output of Canny edge detector on lung tumors, lungs and liver tumors with two choices of thresholds.

The top row of Figure 1 contains the output of the Canny detector using default settings while the bottom row displays superior edge detection using the optimized thresholds.

Edges can also be estimated based on changes in texture. We experimented with a variety of texture filters starting with simple range, entropy and standard deviation filters. Figure 2 presents the output using range and standard deviation filters for the images shown in Figure 1. Edge detection can be applied to these images to detect the texture based edges.

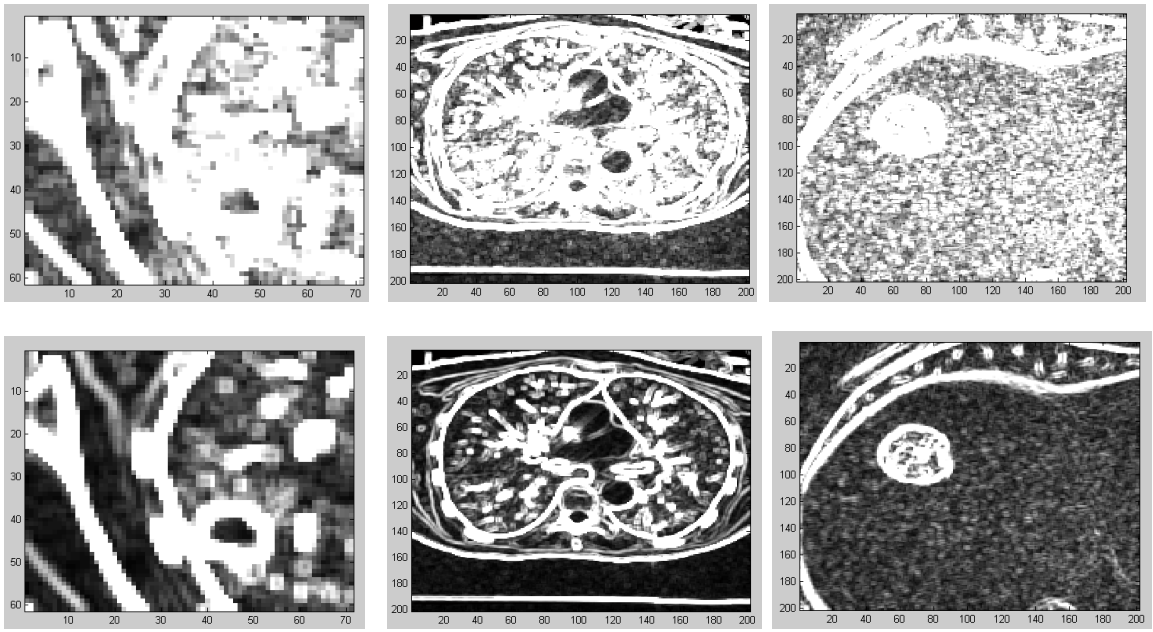

Figure 2 Effect of range (top row) and standard deviation (bottom row) filters

In recent years, more complex filter banks have been developed for texture analysis [10, 11, 12]. We evaluated the use of the Schmidt filter bank, a set of rotationally invariant filters of the Gabor-like filters [13].

A sample image and responses to the filters can be seen in figure 3.

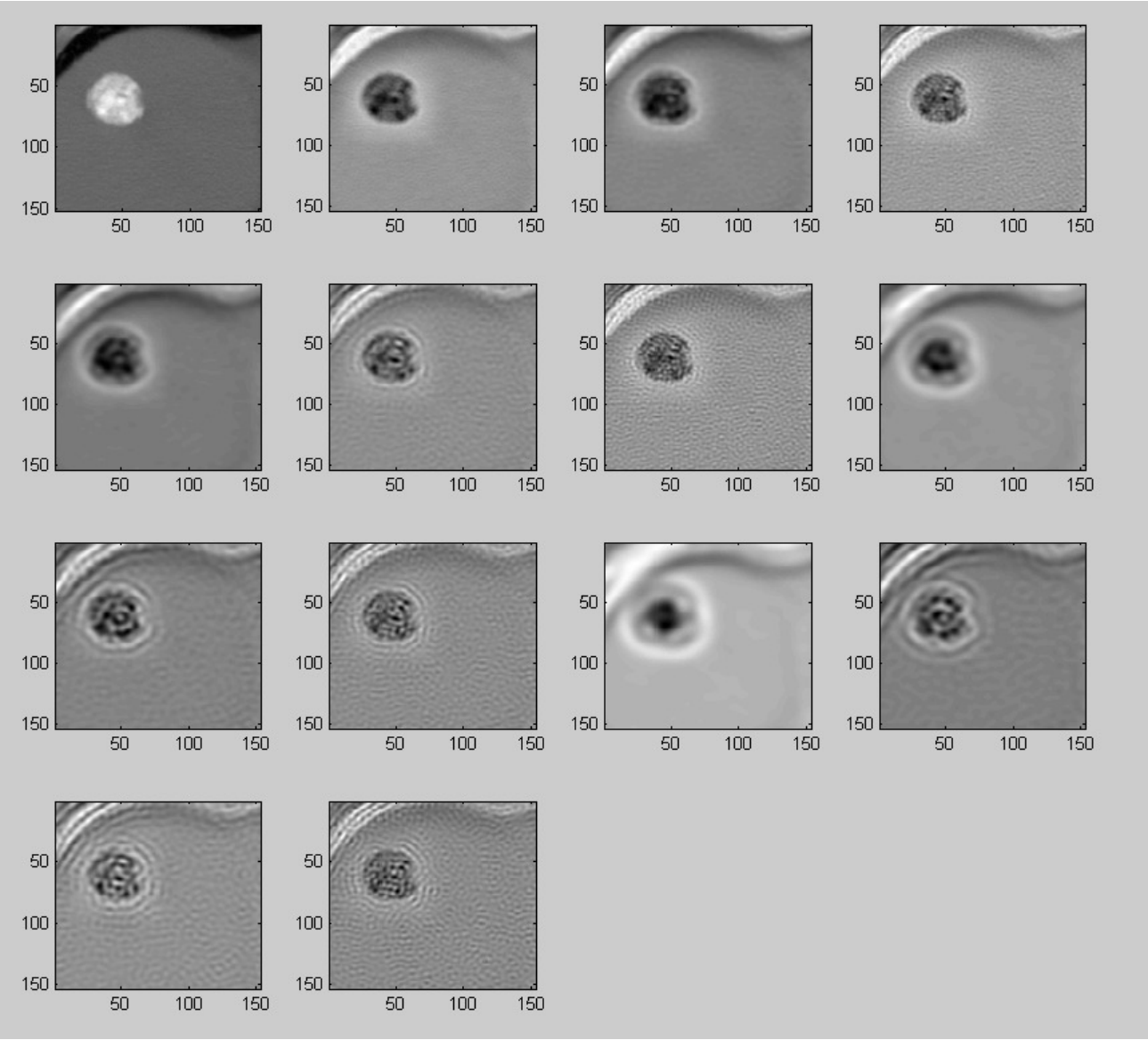

Figure 3 Image containing tumor (top left) and response to 13 Schmidt filters

We averaged the output of convolving the 13 filters with the image as seen in Figure 4.

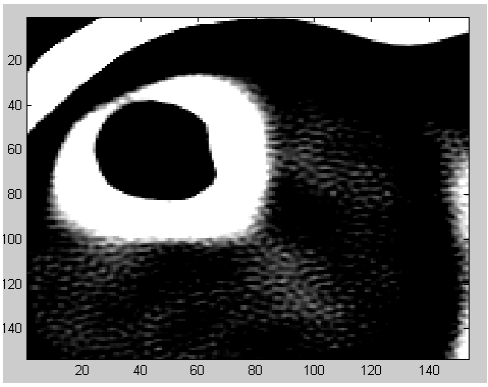

Figure 4 Average of applying the13 Schmidt rotationally invariant filters to the image

Finally, we performed Canny edge detection on the filtered output. As seen in figure 4, we can achieve a reasonable estimation of the edges using this method.

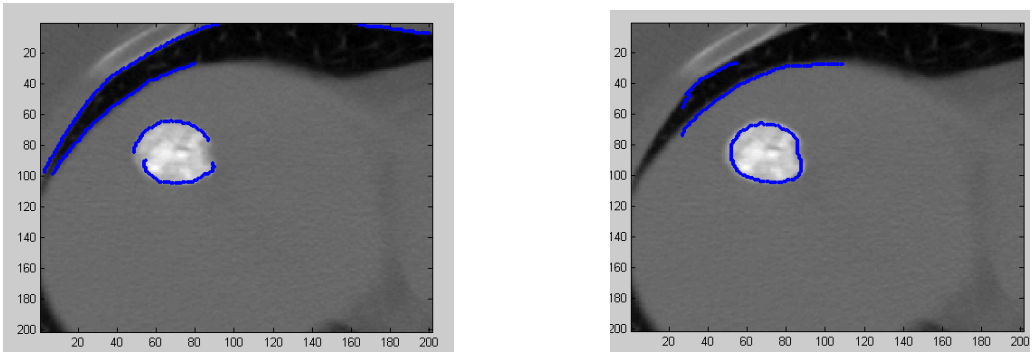

Figure 5 Edge estimation using intensity (left) and texture (right) using Schmidt filters

2.2 Kernel density estimation

As described above, edges can arise due to abrupt changes in texture or intensity or both. A weighted sum of the two (or more, if additional cues are used) edges can be used for the final edge density estimate. The ideal weighting between intensity and texture based edges can be learnt using training data.

In the general case, as described in [5], the KDE of the edge map can be constructed using:

$$p_{edge}(\mathbf{s}) = \frac{1}{N_{edge}} \sum_{i=1}^{N} E(\mathbf{s}_i) K_{\bar{\mathbf{I}}_i}(\mathbf{s} - \mathbf{s}_i)$$

Here, $E(s_i)$ is a weighted sum of the edges obtained from different cues including intensity, texture and prior knowledge including contours from adjacent slices.
In clearly distinguishable organs like lungs, we used a fixed-width isotropic Gaussian kernel. We used the fixed-point algorithm as described in [5] to achieve fast convergence. Specifically:

$$\mathbf{s} \leftarrow \frac{\sum_{i=1}^{N} E(\mathbf{s}_i) K_{\sigma^2\mathbf{I}}(\mathbf{s} - \mathbf{s}_i)\mathbf{s}_i}{\sum_{i=1}^{N} E(\mathbf{s}_i) K_{\sigma^2\mathbf{I}}(\mathbf{s} - \mathbf{s}_i)} = \mathbf{m}(\mathbf{s})$$

This is similar to the mean shift described in [6]. Spurious maxima can be avoided by choosing the kernel size appropriately. However, as would be expected for a mean-shift like algorithm, the snake may not progress into low edge density regions along the object boundary and also into boundary concavities depending on the kernel size and the distance of the initialization curve from the ridges. This drawback is addressed by generating additional sample points using the principal surface approach [7].

The sequence of steps for this approach is demonstrated below:

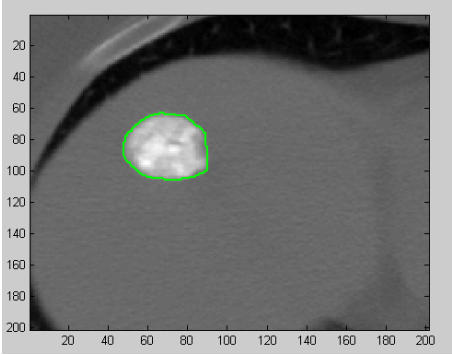

Figure 6 Liver tumor contoured by a radiation oncologist (in green), considered the gold standard

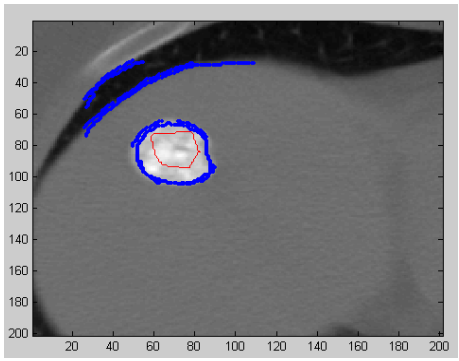

Figure 7 Combined intensity and texture edge (in blue) and initialization (in red)

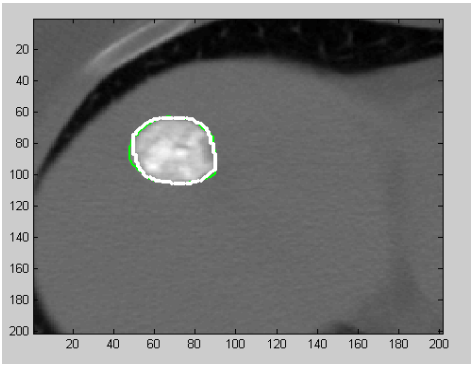

Figure 8 Final result using non-parametric snakes algorithm (in white) compared to hand-drawn contour (in green).

In most cases, the edge detection found the inside edge that was 1-4 pixels inside the manually drawn contour. We used a morphological operator to grow the contours by n pixels, where n was determined based on the training data to maximize overlap.

2.3 Principal curves

Principal curves [7, 8] were used to add points in areas where the output of the non-parametric snakes was sparse. Depending on the initialization and shape of the edge maps, the output of the the non-parametric snakes occasionally resulted in points clustering at edges of low density regions as seen in Figure 9. Using the updated edge map consisting of the weighted sum of the output of the edge detector from the intensity and texture edges and the user input (if applicable), and applying the principal curves algorithm, we can extrapolate additional points along the ridge of the KDE.

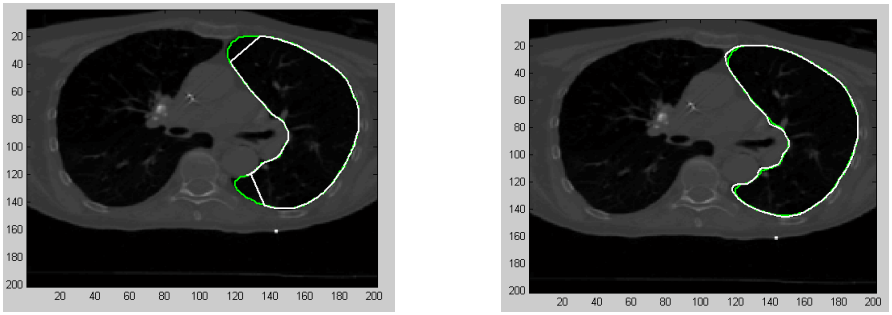

Figure 9 Final result using non-parametric snakes algorithm (in white) compared to hand-drawn contour (in green) (left) and extrapolating using principal curves (right)

For each nearest neighbor pair of points in the final output of the nonparametric snake, the midpoint $\mathbf{s}_0$ of the line segment connecting each pair is taken as an initial point. Let $\mathbf{v}_0$ be the major eigenvector of the local covariance of $p_{edge}(\mathbf{s})$ at $\mathbf{s}_0$ and $\mathbf{g}_0$ be the gradient at this point. Let $\mathbf{m}(\mathbf{s})$ be the mean-shift update direction as defined earlier. Starting from $\mathbf{s}_0$, we iterate the following:

$$\mathbf{s} \leftarrow sign(\mathbf{v}_0^T \mathbf{g}_0)\mathbf{v}_0\mathbf{v}_0^T\mathbf{m}(\mathbf{s})$$

This iteration has been shown to converge to the ridge of the edge distribution, therefore is an interpolating sample from the desired boundary. The process is repeated until a satisfactory number of samples are obtained from the boundary, satisfying the requirement of minimum nearest neighbor distances.. This is determined considering the distribution of nearest neighbor distances between the samples in the nonparametric snake

2.4 Learning weights

When a selection of manually segmented slices is available for training, the data can be used to create thresholds for edge detection as well as for learning the weights of edges obtained using texture and intensity. We created a simple loss function based on the overlap between the manually contoured and algorithmically derived contours using the Jaccard overlap measure.

Jaccard similarity coefficient defined as:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

where A=pixels in manually segmented contour and B=pixels in contour using non-parametric snakes.
The loss function is defined as

$$L = 1 - J(A,B)$$

The thresholds for each organ and tumor for the Canny edge were the values at which L was minimized for the training data. We similarly learned the relative weights of the texture and intensity edges to achieve maximum overlap with the

supervised samples. This was used for segmentation of other slices in that volume in a 3D-CT scan set. As described in [8], we can also incorporate user input in the segmentation.

3. Data

After attaining the necessary approval, we obtained routinely collected 3D CT data used for treatment planning from 4 patients, two with lung tumors and two with liver tumors. We had 100 slices manually annotated by a radiation oncologist. We compared the performance of our segmentation algorithms with the manually annotated contours using the Jaccard overlap measures.

4. Results and discussions

Overlap with manual segmentation: We achieved high overlap with the gold standard in the case of lung tumors with either intensity or texture or both. For liver tumors, we obtained better segmentation using both the intensity and the texture. In all cases, we obtained more than 85% overlap. We achieved even higher overlap if we used contours from an adjacent slice for the initialization. Preliminary data suggest that there are similar differences in overlap measures between two clinicians who contour the same dataset. Moreover, this approach allows for the easy incorporation of user input, either in a semi-supervised fashion or by propagating manually drawn contours from one slice to adjacent slices.

Comparison to other techniques: we also performed some preliminary performance comparisons with level sets and active contours. Non-parametric snakes, especially when we used a fixed-width isotropic Gaussian kernel, performed extremely well from a speed perspective, as they achieved good overlap after just two to three iterations with typically a 5-10x gain in speed.

5. Conclusions and future work

Non-parametric snakes provide fast segmentation that is robust to noise and allow the incorporation of multiple cues including intensity, texture and user input. They do not require the optimization of many parameters, unlike many other segmentation algorithms. Although we used a fixed, predetermined kernel width in this implementation, the kernel width can be optimized automatically using leave-one outcross-validation if desired. Alternatively, variable bandwidth kernels can used to improve robustness to noise.[5] Using non-parametric snakes based on a kernel density estimate combining intensity and texture based edges provides a robust, near real-time segmentation algorithm. This framework allows external information in the form of shape priors, user input and information from adjacent slices to be incorporated easily. One of the limitations is that there is no constraint on the final shape of the contour. Thus, the algorithm can have issues in regions of sparse density. Multiple competing edges can also cause problems for the algorithms, or as there

We are working on improving the robustness of the algorithm incorporating better edge features and experimentation with data from multiple sources. As this is a near real-time, algorithm, we are currently creating an interactive segmentation procedure for radiation oncologists that incorporates machine learning with image processing.

Acknowledgments

We would like to acknowledge Dr. Apiradee Srisuthep, Research Fellow in the Department of Radiation Medicine, OHSU, Portland, OR for her help in acquiring the manually labeled data. This work was supported in part by the National Library of Medicine (NLM) Training grant 2T15LM007088 (JKC) and NSF grants ECS-0524835 and ECS-0622239 (UO, DE)

REFERENCES

[1] “Radiation Therapy for Cancer: Q&A - National Cancer Institute”; http://www.cancer.gov/cancertopics/factsheet/therapy/radiation

[2] “Intensity-Modulated Radiation Therapy”; http://www.radiologyinfo.org/en/info.cfm?pg=imrt&bhcp=1

[3] M. Kass, A. Witkin, D. Terzopoulos, “Snakes: Active contour models,” International Journal of Computer Vision, vol. 1, pp. 321-331, 1988.

[4] C. Xu, J. Prince, “Snakes, shapes, and gradient vector flow,” IEEE Transactions on Image Processing, vol. 7, pp. 359-369, 1998.

[5] U. Ozertem, D. Erdogmus, “Nonparametric Snakes,” IEEE Transactions on Image Processing, vol. 16, pp. 2361-2368, 2007.

[6] U. Ozertem, D. Erdogmus, Tian Lan, “Mean Shift Spectral Clustering for Perceptual Image Segmentation,” Proceedings of ICASSP 2006, vol. 2, 2006.

[7] D. Erdogmus, U. Ozertem, “Self-Consistent Locally Defined Principal Surfaces,” Proceedings of ICASSP 2007, vol. 2, pp. 549-552, 2007.

[8] J. Kalpathy-Cramer, U. Ozertem, W. Hersh, M. Fuss, D. Erdogmus, “Semi-supervised Segmentation using Non-parametric Snakes for 3D-CT Applications in Radiation Oncology”, Machine Learning for Signal Processing, 2008. MLSP 2008. IEEE Workshop on , vol., no., pp.109-114, 16-19 Oct. 2008

[9] J.F. Canny, A computation approach to edge detection, IEEE Transaction on Pattern Analysis and Machine Intelligence, vol 8, p 679-698, Nov1986

[10] T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. International Journal of Computer Vision, 43(1):29-44, June 2001.

[11] C. Schmid. Constructing models for content-based image retrieval. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, volume 2, pages 39-45, 2001.

[12] J.M. Geusebroek, A.W.M. Smeulders and J. van de Weijer. Fast Anisotropic Gauss Filtering. IEEE Transactions on Image Processing, 12(8):938-943, 2003.

[13] http://www.robots.ox.ac.uk/~vgg/research/texclass/filters.html