

# A Comparison of Feature Vectors in a Graph Cut-Based Liver Segmentation Algorithm

Laramie Paxton\*

*Math and Natural Sciences Department  
Marian University–Wisconsin, U.S.A*

Yufeng Cao

*Keck School of Medicine  
University of Southern California, U.S.A.*

Kevin Vixie, Yuan Wang

*Department of Mathematics & Statistics  
Washington State University, U.S.A.*

Chaan Ng

*M.D. Anderson Cancer Center  
University of Texas, U.S.A.*

Brian Hobbs

*Cleveland Clinic, U.S.A.*

---

---

## Abstract

Liver image segmentation presents a challenging set of conditions and is an active area of research. In this paper, we compare the effectiveness of five different feature vectors used in a preprocessing step for a graph cut-based semi-automatic liver segmentation algorithm. The feature vectors tested are formed using a median filter, averaging filter, Gaussian filter, neighborhood, and novel use of time series data. When compared to the expert-provided ground truth, the time series approach outperforms the others and yields results comparable to other recent models in the literature, giving a mean volume error (VOE) of 32.9 percent, mean Dice similarity coefficient (DSC) of 0.8, and mean runtime of 74 seconds. We also include a modified boundary term in the energy functional and normalize both terms in order to avoid further scaling of the boundary term. In place of a training process, we utilize sample Regions of Interest provided by expert radiologists to compute sample vector means for healthy and tumor tissues that are used in the regional term of the functional. Contribution: The time series feature vector method represents a novel approach that utilizes the time series data obtained from a sequence of 59 CT scans as a preprocessing step, along with using a simplified boundary term in the energy functional.

**Keywords:** liver tumor, time series, medical imaging, graph cut, image segmentation, feature vector

© 2012, IJCVSP, CNSER. All Rights Reserved

IJCVSP

ISSN: 2186-1390 (Online)  
<http://cennser.org/IJCVSP>

*Article History:*  
Received: 5 February 2021  
Revised: 6 July 2021  
Accepted: 16 August 2021  
Published Online: 20 August 2021

---

\*Corresponding author

*Email addresses:* [realtimemath@gmail.com](mailto:realtimemath@gmail.com) (Laramie Paxton), [yufengcaowsu@gmail.com](mailto:yufengcaowsu@gmail.com) (Yufeng Cao), [vixie@speakeasy.net](mailto:vixie@speakeasy.net) (Kevin Vixie), [yuan.wang.stat@gmail.com](mailto:yuan.wang.stat@gmail.com) (Yuan Wang), [cng@mdanderson.org](mailto:cng@mdanderson.org) (Chaan Ng), [hobbsb@ccf.org](mailto:hobbsb@ccf.org) (Brian Hobbs)

---

---

## 1. INTRODUCTION

With the number of cases of liver cancer on the rise, researchers are continuing to look for innovative, non-invasive, and accurate time-saving strategies to assist in the detection and treatment of the disease. Whether determining

tumor locations for surgical removal or measuring the response of tumors to a chemotherapy regiment, detecting the presence of tumors or tracking their growth is critical for effective treatment plans and increased survival rates. Even with the advent of advanced medical imaging techniques, such as CT scans or MRI's, radiologists have tended to rely on manual tracing of tumors. However, not only is this process very time-consuming, there is also a significant degree of variation between experts and even between segmentations performed by the same expert on different occasions. This leads to a lack of reproducibility in the process as a whole [1].

For this reason, researchers in the fields of computer vision and image analysis have worked to develop a wide variety of different techniques to assist in the process of liver tumor detection, from interactive methods that require clinicians to be highly involved in the process to semi-automatic and fully automatic segmentation techniques that rely, in the latter case, completely on segmentation algorithms. CT scans are currently the standard method used for liver tumor detection in the field due to their relatively low expense and imaging time and high resolution. However, there are still many challenges that make this problem a difficult one and prompt researchers to continue to develop improved methods to overcome them. For example, CT images may still exhibit a high degree of variation depending on the type and stage of tumor being scanned, the amount of contrast agent used in the patient, and even the delay involved with each scan. Moreover, the computer-assisted segmentation process for liver tumors is made more complicated due to 1) a lack of well-defined boundaries in many cases; 2) the lack of a "template" shape for the tumors; and 3) the low-contrast nature of the tumors and their surrounding tissues [2].

## 2. RELATED WORKS

Graph cuts [3], [4] are a popular and versatile image segmentation technique. Strengths of this technique include the fact that they are independent of initializations, unlike the active contour method, and instead of being iterative, they most often calculate a global energy minimization. The theoretical basis upon which graph cuts depend is that of combinatorial optimization. Therefore the main goal is to find the minimum of a given energy functional by executing the lowest-cost graph cut that will divide an undirected, connected graph into two separate pieces that are disconnected. We most often represent our graph using pixels for vertices and derive the edges of the graph using 4- or 8-neighborhood connectivity. We then add a node for the source and another node for the sink along with edges that connect each pixel to each of these nodes.

Incorporated into our energy functional are both a data fidelity term, which is regional, and a perimeter regularization term, which represents the boundary. The data

fidelity term has a cost associated with it involving the classification of pixels over particular regions, such as the foreground or the background. On the other hand, the perimeter regularization term has a cost that involves the differences in pixels on various boundaries, and the key point to make in this case is that the penalty is applied only to those edges where we make cuts. In this way, we obtain a minimization using the *max-flow min-cut theorem* [5], which states that the weight of the edges in the cut of minimum capacity in a flow network equals the maximal flow that can travel along the network. Here, the cut of minimum capacity refers to the minimum weight associated with all the edges that would need to be removed in order to have our source disconnected from our sink in the graph.

In the proposed model, we employ the Boykov-Kolmogorov (BK) max-flow algorithm [6], which is a very common algorithm for running graph cuts. The notable issue that arises from using un-modified graph cuts involves the challenges that weakly defined boundaries and noise in images pose; therefore, many researchers have included additional methods to address these issues, such as the random walkers algorithm [7]. Another noteworthy example can be found in [2], which utilizes a four-step process including a kernelized fuzzy c-means (FCM), confidence connected region growing algorithm, and graph cut. In [8], kernel density estimation is used to develop a nonlinear statistical shape prior in such a way that the energy functional can be minimized through iterative graph cuts.

## 3. PROPOSED METHOD

The focus of this paper is to compare the effectiveness of several different feature vectors used at the preprocessing step in conjunction with the BK algorithm for performing a semi-automatic liver segmentation. While the BK algorithm is relatively simple to implement computationally, these feature vectors are desirable for their ability to improve the graph cut's ability to handle noise and ambiguous boundaries in the tumor. Our contribution here is that one of these feature vectors represents a novel approach that utilizes the time series data obtained from a sequence of 59 CT scans taken 0.5 seconds apart. It outperforms the other methods tested and yields results that are comparable with other models in the literature in terms of effectiveness, as shown in Table 1 and Table 2, respectively. Another contribution is that we also incorporate a simplified boundary term and a normalization step in the energy functional used in the graph cut below, the latter of which allows us to fix  $\lambda$  to one.

We evaluated the feature vectors and the associated segmentation method using a dataset of six liver tumors contained in 2D CT images taken at the M.D. Anderson Cancer Center at the University of Texas. We used ground truth segmentations provided from expert radiologists for comparison, and the training data used consisted of Regions of Interest (ROI's) from which we collected sample

means for healthy and tumor tissues. These were then used in the energy functional defined below. For each patient, a total of 59 images (with  $512 \times 512$  resolution) were taken 0.5 sec apart with a pixel spacing of 0.70 mm or 0.86 mm and slice thickness of 5 mm. We used the 59th image in each sequence for the segmentation. All computations were performed using Matlab 2018a on a personal computer with 4 Gb of RAM and a 2.5 GHz Intel Core i5 CPU.

### 3.1. Preprocessing: Feature Vectors

We “vectorize” each pixel  $p$  in the image in the following ways.

#### 3.1.1. Median Filter

We create a multiscale descriptor by forming a feature vector, with ten entries, using a median filter over a  $k \times k$  neighborhood of  $p$ , where  $k = 1, 2, \dots, 10$ .

#### 3.1.2. Averaging Filter

We create a second multiscale descriptor by taking the anisotropic average around each pixel at different length scales. That is, each entry in our 10-vector for pixel  $p$  is the result of applying an averaging filter over a  $k \times k$  neighborhood of  $p$ , where  $k = 1, 2, \dots, 10$ .

#### 3.1.3. Gaussian Filter

Similarly, we form a feature vector by using a lowpass Gaussian filter over a  $k \times k$  neighborhood of  $p$ , where  $k = 1, 2, \dots, 10$ .

#### 3.1.4. Neighborhood

This feature vector consists of the eight entries not equal to  $p$  of a  $3 \times 3$  neighborhood of  $p$ .

#### 3.1.5. Time Series

We form this feature vector by letting each pixel in our image be represented with a 59-vector whose entries are the pixel’s intensities (in Hounsfield units) at each step of the sequence of 59 images. Thus for a pixel  $p$ , we have

$$p \longrightarrow (p_1, p_2, p_3, \dots, p_{59}). \quad (1)$$

We use the time series data to develop a feature vector because it yields the advantage of using the available data in a way that incorporates the temporal information regarding the healthy and tumor tissue intensity differences along with the spatial information in the initial image. These differences over time arise primarily due to the different way in which the contrast agent is processed in healthy versus tumor tissues. We do this without smoothing the images so that we preserve the time series intensity values for each pixel. We did not perform registration or motion correction on the series.

### 3.2. Segmentation

Graph cuts work by finding the minimizer of an energy functional in which the “cheapest” cut is sought that forms a disconnected graph from an initial connected, undirected graph formed in the following way. Pixels in the image become nodes in the graph and the edges are formed using 4- or 8- connectivity neighborhoods. By adding a source and sink to the graph, we may form edges between each node and these two points. The functional to minimize contains a term that is calculated depending on whether pixels are classified in the foreground or background, and hence is known as a *regional* term. The second term is computed by adding up the “cost” of any edges where cuts are made and thus is known as a *boundary* term.

The energy functional we want to minimize is given by

$$F(L) = \sum_i \|I_i - \mu_{L_i}\|_2 + \lambda \left( \sum_{\substack{\{i,j \mid L_i \neq L_j, \\ i,j \text{ are neighbors}\}} \min\{\|I_i - I_j\|_2^{-1}, 1\} \right), \quad (2)$$

where  $I_i$  denotes the  $i$ th vectorized pixel with label  $L_i = 1, 2$  with sample tissue mean  $\mu_1$  or  $\mu_2$ , respectively.  $L$  is a particular segmentation, and “neighbors” are defined via 4-connectivity. We set  $\lambda$  equal to one and incorporate a normalizing feature for both terms, as described below.

Loosely speaking, the first term represents the 2-norm of the difference between each vectorized pixel  $p$  and the sample tissue mean for the region it is classified in. These sample means are derived from the ROI’s provided in the training data from expert radiologists.

In the second term, we use a simplified variant of a standard way [9] of determining the “cost” of each cut:

$$\min\{\|I_i - I_j\|_2^{-1}, 1\}.$$

This allows for two neighboring (vectorized) pixels whose 2-norm of their difference is large to have a smaller cost for cutting the edge between them. This encourages cuts along ground truth boundaries and discourages them elsewhere. However, we want to avoid making the boundary term significantly smaller than the regional term and allowing for too many cuts that are not along ground truth boundaries. Therefore, we normalize each term in  $F(L)$  above so that each term in each summation is between zero and one. This allows us to fix  $\lambda$  equal to one without the need for further scaling.

### 3.3. Segmentation Algorithm

**input:** Image, matrix  $M$  of vectorized pixels formed using one of the feature vectors, and ROI’s for healthy and tumor sample means.

**output:** Segmented image

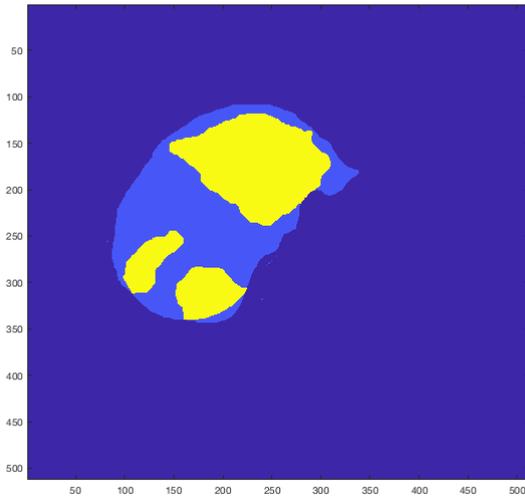


Figure 1: Segmentation for Patient 1 using a time series feature vector.

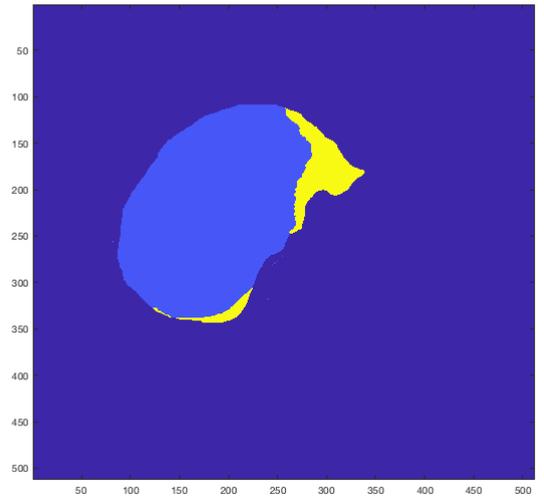


Figure 3: Segmentation for Patient 1 using an averaging filter feature vector.

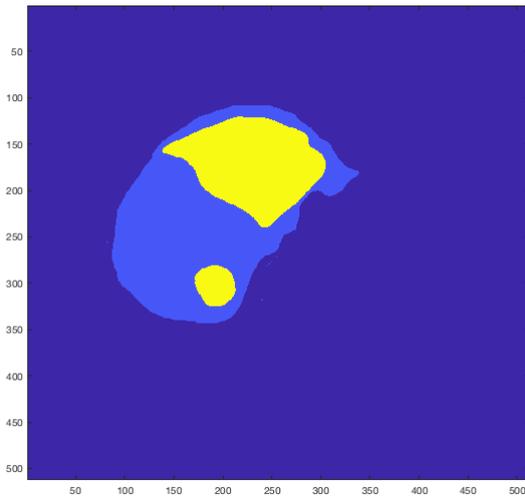


Figure 2: Ground truth segmentation for Patient 1 corresponding to Fig.'s 1, 3, and 4.

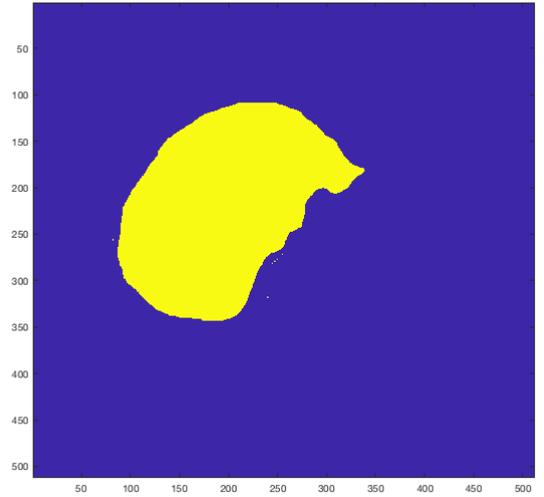


Figure 4: Segmentation for Patient 1 using a Gaussian filter, median filter, or a neighborhood feature vector. That is, all three filters yield the same segmentation results.

1. Load ROI's and compute tissue mean vectors using  $M$ .
2. Generate 4-connectivity matrix  $B$  from  $M$ .
3. Generate edge weight matrix from  $B$ .
4. Generate terminal weight matrix from  $M$ .
5. Run BK graph cut algorithm.
6. Apply pre-made liver mask to image.
7. Apply color map and reshape.

### 3.4. Statistical Evaluation

In order to evaluate and compare the effectiveness of the segmenter performance using each of the feature

vectors above, we computed the following statistics based on [10].

We now give the definitions of each performance metric used below to evaluate the model.

#### *Volumetric Overlap Error*

In order to compute the volumetric overlap error (VOE), we take the total quantity of the pixels contained in the intersection of our tumor ( $S$ ) after segmentation and our ground truth ( $T$ ) and divide by the overall quantity contained in the union of these. If the tumor is segmented perfectly, then the VOE will be 0.

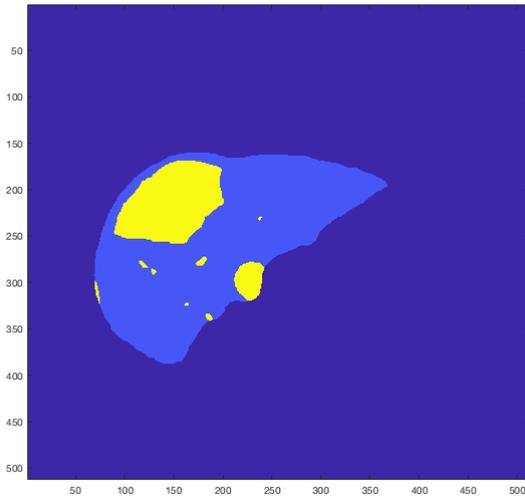


Figure 5: Segmentation for Patient 2 using a time series feature vector.

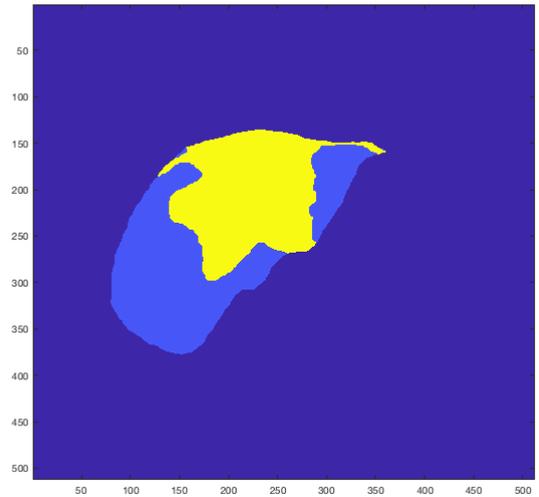


Figure 7: Segmentation for Patient 3 using a time series feature vector.

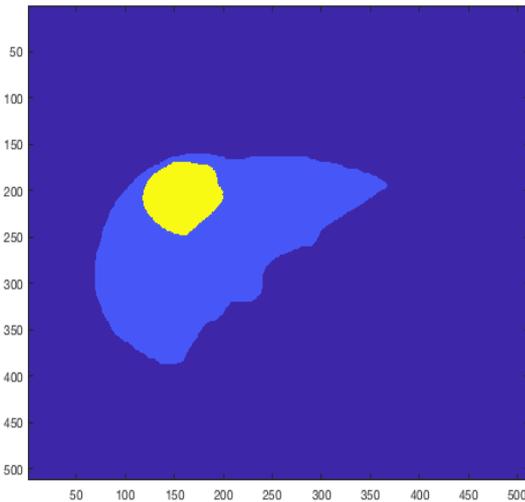


Figure 6: Ground truth segmentation for Patient 2 corresponding to Fig. 5.

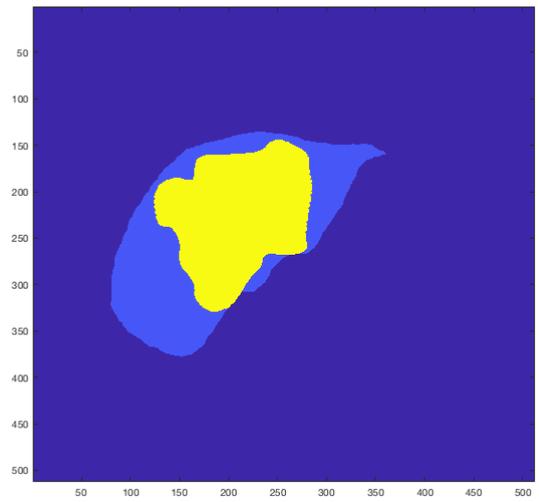


Figure 8: Ground truth segmentation for Patient 3.

$$VOE (\%) = \left(1 - \frac{|S \cap T|}{|S \cup T|}\right) \times 100 \quad (3)$$

#### *Dice Similarity Coefficient*

In order to gauge how well the segmentation has performed overall, we use the Dice similarity coefficient (DSC). In the case that we have a tumor segmented perfectly, the DSC will be 1.

$$DSC = \frac{2|S \cap T|}{|S| + |T|} \quad (4)$$

## 4. Results

We first present the mean VOE and DSC scores along with the mean run-time in Table 1 below for each feature vector segmentation on the dataset of six tumors described above along with a comparison of several other recent models in the literature as shown in Table 2. Next, we include segmented images for different feature vector segmentations using the method outlined above along with the ground truth segmentation. Yellow represents tumor tissue. The first four figures all correspond to Patient 1. Fig. 1 results from the time series feature vector method and Fig. 2 shows the ground truth for Patient 1. Fig. 3

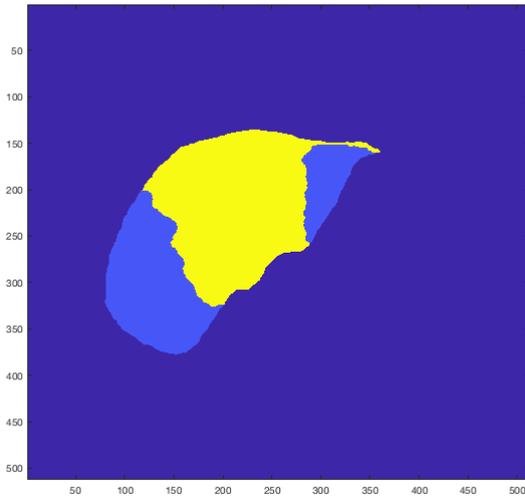


Figure 9: Segmentation for Patient 3 using an averaging filter feature vector.

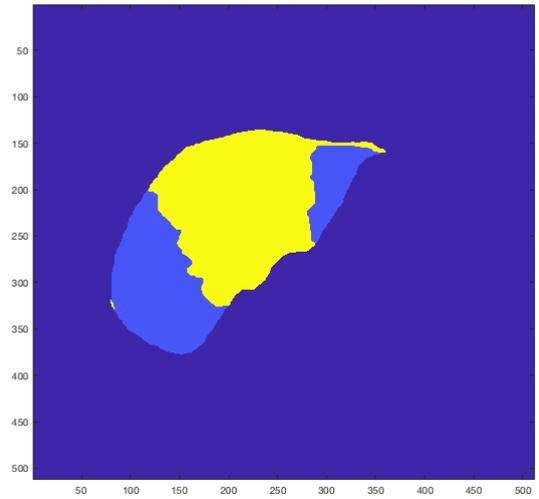


Figure 11: Segmentation for Patient 3 using a neighborhood feature vector.

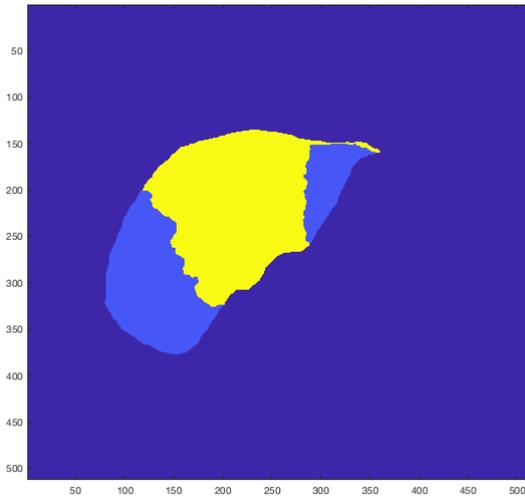


Figure 10: Segmentation for Patient 3 using a Gaussian filter feature vector.

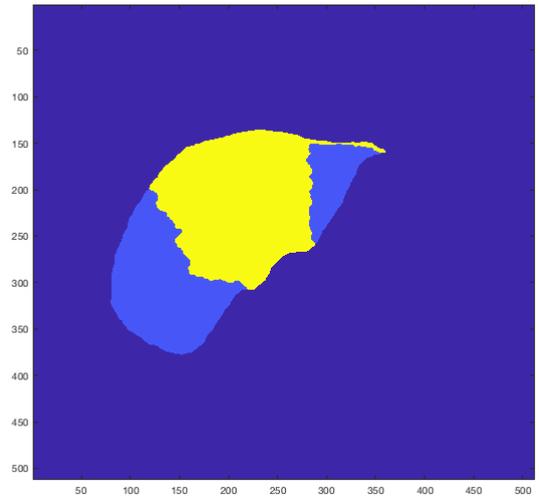


Figure 12: Segmentation for Patient 3 using a median filter feature vector.

corresponds to the averaging filter feature vector. Fig. 4 corresponds to the Gaussian filter feature vector. Note that the segmentations using the Gaussian filter, median filter, and neighborhood-based feature vectors are nearly identical for this patient.

The next images highlight the time series feature vector segmentation approach for two additional patients. Fig. 5 corresponds to the time series feature vector for Patient 2, and Fig. 6 shows the ground truth for this patient. Then we have the time series feature vector for Patient 3 in Fig. 7, with the ground truth for Patient 3 shown in Fig. 8, the averaging filter feature vector in Fig. 9, the Gaussian filter

feature vector in Fig. 10, the neighborhood feature vector in Fig. 11, and the median filter feature vector in Fig. 12.

From the first table, we see that the method using the time series feature vector outperforms the others in this case and obtains a mean VOE of 32.9%, mean DSC of 0.8, and mean runtime of 74 seconds. These results using the time series graph cut method are comparable to other recent results in the literature, as shown in Table 2. We observe that the averaging filter feature vector is the worst performing, with a mean VOE of 46.6%, DSC of 0.65, and mean runtime of 83 seconds. On the other hand, the Gaussian filter, median filter, and neighborhood-based feature

Table 1: Feature Vector Comparison.

Method	Statistical Measures Reported as Averages		
	VOE (%)	DSC	Run-time (sec.)
Median	40.8	.73	60
Averaging	46.6	.65	83
Gaussian	40.0	.74	91
Neighborhood	40.1	.74	45
Time Series	32.9	.80	74

Table 2: Model Comparison.

Model	Statistical Measures Reported as Averages		
	VOE (%)	DSC	Run-time (sec.)
Ronneberger et al. [11], 2015	39.0	.73	n/a
Moghbel et al. [7], 2016	22.8	.75	30
Wu et al. [2], 2017	29.0	.83	45
Zeng et al. [12], 2018	33.9	.73	n/a
Gruber et al. [13], 2019	21.8	n/a	n/a
Proposed Method	32.9	.80	74

vectors are all remarkably similar with a mean VOE of approximately 40% and DSC of approximately 0.74. Their runtimes are 91, 60, and 45 seconds, respectively. This similarity is likely due to the fact that all of these methods utilize information contained in the neighborhood of each pixel  $p$  in the image to form the feature vector for the pre-processing step, whereas the time series approach utilizes changes in each pixel's intensity over time.

In summary, the contributions of this paper are 1) We compare various feature vectors for use with the BK graph cut algorithm. 2) We find that the proposed time series method outperforms the others (Table 1) and achieves results comparable to other models in the literature (Table 2). 3) We incorporate a modified boundary term in the energy functional together with a normalization that scales both terms and removes the need for optimizing  $\lambda$ .

## 5. CONCLUSIONS

In this paper, we evaluate the effectiveness of five different feature vectors used as a preprocessing step in a semi-automatic graph-cut based liver segmentation method. A novel use of time series data to develop a feature vector provides the most effective method of those tested and achieves results comparable with other recent models in the literature. We also make use of a modified boundary cost term and normalize the regional and boundary terms in the energy functional. Sample ROI's are provided by expert radiologists, and we use these regions to compute sample means for the healthy and tumor tissues that are used to compute the regional term in the functional. The method using the time series data provides a relatively high degree of accuracy for a short runtime and an algorithm

that is simple to implement computationally with no additional training process. Further testing and refinement using the time series method would be beneficial given the promising preliminary results from this study.

## 6. Acknowledgment

Note: The NIH grant P30-CA016672 helped to support a portion of this research.

## References

- [1] S. Kadoury, E. Vorontsov, A. Tang, Metastatic liver tumour segmentation from discriminant grassmannian manifolds, *Physics in Medicine & Biology* 60 (16) (2015) 6459.
- [2] W. Wu, S. Wu, Z. Zhou, R. Zhang, Y. Zhang, 3d liver tumor segmentation in CT images using improved fuzzy c-means and graph cuts, *BioMed Research International* 2017.
- [3] Y. Boykov, G. Funka-Lea, Graph cuts and efficient nd image segmentation, *International Journal of Computer Vision* 70 (2) (2006) 109–131.
- [4] M. G. Linguraru, W. J. Riechbourg, J. Liu, J. M. Watt, V. Pamulapati, S. Wang, R. M. Summers, Tumor burden analysis on computed tomography by automated liver and tumor segmentation, *IEEE Transactions on Medical Imaging* 31 (10) (2012) 1965–1976.
- [5] G. Dantzig, D. R. Fulkerson, On the max flow min cut theorem of networks, *Linear Inequalities and Related Systems* 38 (2003) 225–231.
- [6] Y. Boykov, V. Kolmogorov, An experimental comparison of min-cut max-flow algorithms for energy minimization in vision, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26 (9) (2004) 1124–1137.
- [7] M. Moghbel, S. Mashohor, R. Mahmud, M. I. B. Saripan, Automatic liver tumor segmentation on computed tomography for patient treatment planning and monitoring, *EXCLI Journal* 15 (2016) 406.

- [8] J. C. Chang, T. Chou, Iterative graph cuts for image segmentation with a nonlinear statistical shape prior, *Journal of Mathematical Imaging and Vision* 49 (1) (2014) 87–97.
- [9] Y. Boykov, M. Jolly, Interactive graph cuts for optimal boundary and region segmentation of objects in nd images computer vision, *Proc. 8th IEEE Int. Conf. on Computer Vision ICCV 2001*.
- [10] A. A. Taha, A. Hanbury, Metrics for evaluating 3d medical image segmentation: analysis, selection, and tool, *BMC Medical Imaging* 15 (1) (2015) 29.
- [11] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2015, pp. 234–241.
- [12] Y.-z. Zeng, S.-h. Liao, P. Tang, Y.-q. Zhao, M. Liao, Y. Chen, Y.-x. Liang, Automatic liver vessel segmentation using 3d region growing and hybrid active contour model, *Computers in biology and medicine* 97 (2018) 63–73.
- [13] N. Gruber, S. Antholzer, W. Jaschke, C. Kremser, M. Haltmeier, A joint deep learning approach for automated liver and tumor segmentation, *arXiv preprint arXiv:1902.07971*.