

3D live-wire-based semi-automatic segmentation of medical images

Ghassan Hamarneh, Johnson Yang, Chris McIntosh, Morgan Langille

School of Computing Science, Simon Fraser University, Burnaby, BC, V5A 1S6, Canada

ABSTRACT

Segmenting anatomical structures from medical images is usually one of the most important initial steps in many applications, including visualization, computer-aided diagnosis, and morphometric analysis. Manual 2D segmentation suffers from operator variability and is tedious and time-consuming. These disadvantages are accentuated in 3D applications and, the additional requirement of producing intuitive displays to integrate 3D information for the user, makes manual segmentation even less approachable in 3D. Robust, automatic medical image segmentation in 2D to 3D remains an open problem caused particularly by sensitivity to low-level parameters of segmentation algorithms. Semi-automatic techniques present possible balanced solution where automation focuses on low-level computing-intensive tasks that can be hidden from the user, while manual intervention captures high-level expert knowledge nontrivial to capture algorithmically. In this paper we present a 3D extension to the 2D semi-automatic live-wire technique. Live-wire based contours generated semi-automatically on a selected set of slices are used as seed points on new unseen slices in different orientations. The seed points are calculated from intersections of user-based live-wire techniques with new slices. Our algorithm includes a step for ordering the live-wire seed points in the new slices, which is essential for subsequent multi-stage optimal path calculation. We present results of automatically detecting contours in new slices in 3D volumes from a variety of medical images.

Keywords: Medical imaging, medical image analysis, semi-automatic segmentation, live-wire, 3D

1. INTRODUCTION

Segmenting structures of interest from medical images is a necessary step towards quantification and visualization of anatomy and physiology. Numerous methods, ranging from manual to fully automated, have been developed. Although providing total control over the segmentation result, manual segmentation is tedious, time-consuming, suffers from inter/intra-operator variability, and hasn't been effective in allowing for an intuitive and effective integration of 3D image information. Automated methods, on the other hand, often suffer from sensitivity to parameter selection and do not necessarily produce the desirable delineations. Semi-automated techniques were designed to embrace user-interaction while still performing most of the delineation tasks automatically [1, 2]. However, guiding semi-automatic image segmentation algorithms through intuitive user interactions has mostly been limited to 2D, as in Snakes [3], Live-wire[4], and others [5, 6, 7, 8, 9, 10]). In this work we present a semi-automatic 3D segmentation method. We semi-automatically segment a number of 2D contours using 2D Live-wire in image slices with different orientations. Then generate additional contours on new slices automatically. The new contours are optimal paths connecting the points of intersection between the new slice planes and the original contours. Our method extends the work of Falcao *et al* [11] by using a novel approach for acquiring and ordering the data points. Point-ordering is essential for the subsequent optimal path calculation and is accomplished by traversing the lines connecting the points according to certain rules that allow us to deal with irregular 3D structures containing concavities. In the remainder of this paper we review the 2D Live-wire formulation in section 2.1 and present the an overview of the 3D extension in section 2.2. The details of our method including the point ordering algorithm are presented in section 2.3. We then present results on a variety of synthetic and real data and finally draw conclusions.

2. METHODS

Our 3D semi-automatic segmentation technique is based on extensions to the classical 2D live-wire method and an earlier extension by Falcao *et al*[11]. Although live-wire is an effective tool for 2D segmentation, its formal extension to 3D would require computationally demanding calculations of optimal paths from seed points to every voxel in an image volume and, furthermore, would necessitate nontrivial human-computer interface mechanisms to intuitively locate 3D coordinates of voxels at separating interfaces (e.g. those with large image gradient magnitude). Therefore, we generate 3D refinable segmentations by automatically calculating new live-wires in new image slices from user-defined 2D live-wires on selected slices.

2.1. Live-wire

The 2D live-wire implemented in this study uses the gradient magnitude $f_M(q)$ gradient direction $f_D(p, q)$, and canny edge detection $f_C(q)$ as cost terms for the local 2D live-wire cost function calculation from pixel p to pixel q $C(p, q)$.

$$C(p, q) = w_M f_M(q) + w_D f_D(p, q) + w_C f_C(q) \quad (1)$$

Each cost term is weighted by a weight constant w_M , w_D , and w_C to allow each cost term to contribute to the cost function at different rates. The gradient magnitude cost term for any pixel q is defined as

$$f_M(q) = 1 - \sqrt{\frac{dq^2}{dx^2} + \frac{dq^2}{dy^2}} / \max(G) \quad (2)$$

where $\max(G)$ represents the largest gradient magnitude in the 2D image. The live-wire algorithm finds the minimum cost path between any two pixels in the 2D image so the gradient magnitude must be inverted such that strong edges correspond to low costs. The gradient direction cost term pixel p going to pixel q is defined as

$$f_D(p, q) = \arccos\left(\frac{dp/dx}{G(p)}x \frac{dq/dx}{G(q)} + \frac{dp/dy}{G(p)}x \frac{dq/dy}{G(q)}\right) / \pi \quad (3)$$

where $G(p)$ represents the gradient magnitude for pixel p. The Canny [12] local cost term was implemented using the edge detection function from MATLAB's Image Processing Toolbox with default settings. The output of these edge detection functions is a binary image with edges represented as pixels containing ones and the rest of the background pixels containing zeros. Therefore, the binary images were inverted to give strong edges low costs. Furthermore, the function $C(p, q)$ is scaled by $\sqrt{2}$, if pixel q is a diagonal neighbor to pixel p, which corresponds to the Euclidian metric.

The 2D live-wire method requires an initial seed point from the user on the boundary of the desired object. The local cost calculation is then calculated for each pixel in the image starting from the initial seed point such that the path is minimal. The user chooses the next seed point based on the visible minimal path and the algorithm is repeated.

2.2. 3D extension

The 3D live-wire extension requires that the user first traces out a few initial contours using 2D live-wire in planes of their choice. It is recommended that the initial contours be distributed in the volume such that they capture the topological features of the target structure. Nevertheless, more contours can be added later as the user requires a more refined segmentation. The specific points along the initial contours will then be used as seed and target points for automatically generating additional orthogonal live-wire contours. The user has control over the number and orientation of the new slices. Alternatively, the user can provide a resolution that is desired for each dimension and the spacing and number of contours to match the resolution is calculated by the tool. Each new slice is tested for intersection with the user defined contours before an automatic contour is generated.

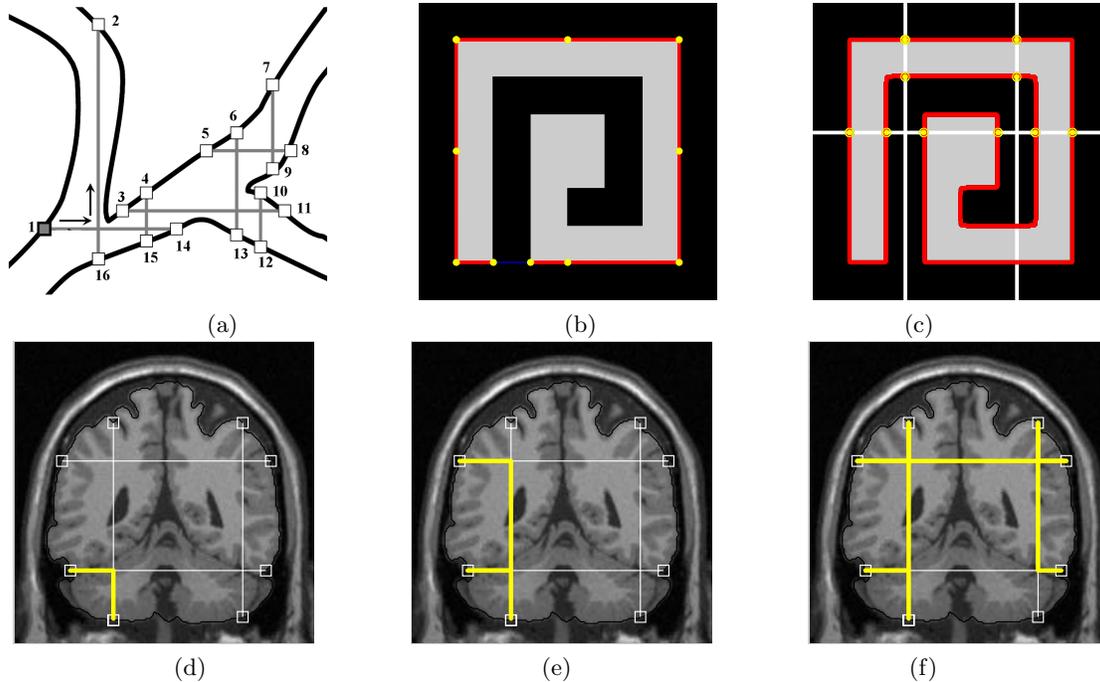


Figure 1. The *turtle* point ordering algorithm. (a) Ordering deals with complex, convex shapes (traversing the gray paths starting from point 1 and making left turns only). (b) Live-wire unable to detect target shape given 10 improperly placed points. (c) Proper choice of intersecting planes results in a new set of 10 points and a successful live-wire segmentation. (d-f) Ordering algorithm on a transversal MRI slice of the brain.

2.3. Point ordering

The intersection points from the manually delineated live-wire contours must be ordered before they can be used for automatic contour generation. The new algorithm for point ordering uses many observations about the points that are derived from orthogonal slices. A single user-defined contour intersects with each new slice at an even number of points: If we traverse the contour in a specific direction then for each time we cross from one side of the slice to the other, there corresponds a time when we will cross the slice in the other direction. We call these crossings *entering* and *exiting* the slice. These pairs of contour points that are entering and exiting a single slice are used to define the endpoints of intersecting line segments. These line segments in turn create a connected graph forming paths for an L-system “turtle” (similar to Logo programming language). Starting from an arbitrary point, the turtle traverses the graph by moving forward and making left turns only at the intersections (Fig. 1). The point ordering for the new live-wires is achieved by storing, as the path is traversed, the points where two consecutive left turns are performed.

The new point ordering algorithm can handle complex structures that may have more than a single closed contour in a single slice (Fig. 2(a,b)). If a 2D slice contains only a single contour than the point ordering algorithm orders the points as previously described (Fig. 2(c,e,g)). If the 2D slice contains more than a single contour, then there will be a disjoint set of intersecting line segments for each closed contour in the slice (Fig. 2(d,f)). Each disjoint set of intersecting line segments is then used to order the points independently from any other closed contours in the 2D slice (Fig. 2(h)).

3. RESULTS

The new implementation of the 3d live-wire tool was tested on many synthetic and medical volumes. The first example of a synthetic “cup” shows that the point ordering algorithm can handle non-convex shapes (Fig. 3(a)). The cortical surface of the brain from an MRI volume was manually segmented in several slices that were used to generate additional contours (Fig. 4). The combination of manual and automatic generated contours were used

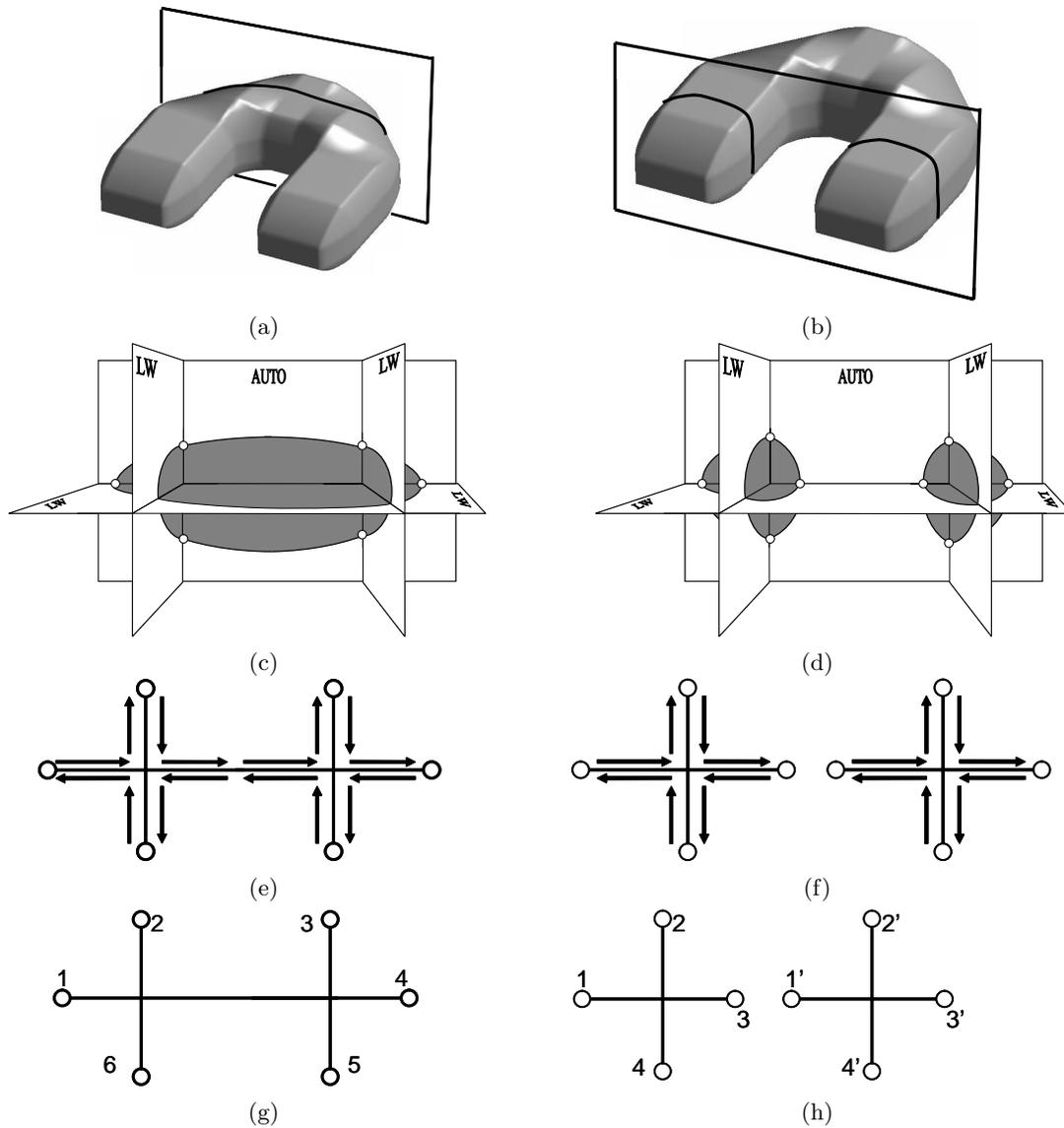


Figure 2. Illustration of how the 3D live-wire tool can manage to successfully order the points of more than a single closed contour in a single image slice. The 3D structure of a synthetic 'u' shaped tube is shown with an image slice containing a single contour (a) and a image slice containing two separate contours (b). (c,d)The automatic live wire slices and orthogonal manual live-wire slices are illustrated single and multiple contours. (e-h) The path of the Logo turtle and the resulting point ordering are shown for a single contour and for two disjoint contours.

to produce a solid 3D segmentation of the cortical surface of the brain (Fig. 3(b-c)). Only a small number of manually derived contours were needed to generate additional contours for both the synthetic cup and cortical brain examples.

Several other medical volumes were segmented including the hippocampus and ventricles from a human brain (Fig. 5, 6), and a kidney from a mouse (Fig. 7). All 3D objects from the medical volumes were successfully segmented using the 3d live-wire tool with only a few manually delineated contours in each dimension.

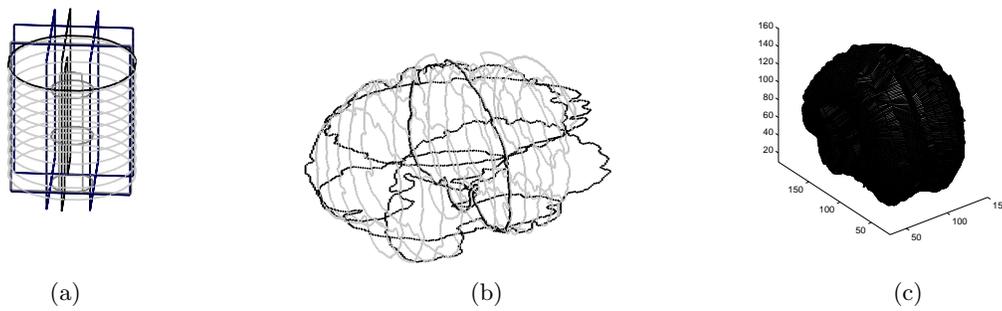


Figure 3. Segmentation results. Automatically detected contours (gray) for the (a) “cup” and (b) the cortical surface examples, from user-defined contours (black). (c) Extracted cortical surface.

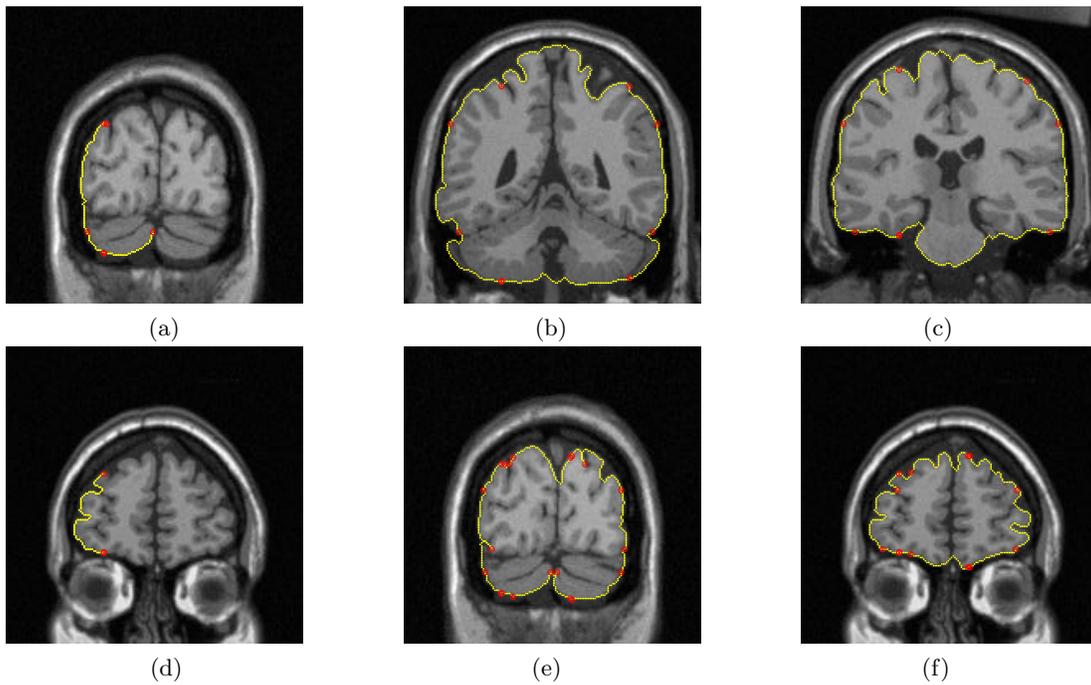


Figure 4. Segmenting the cortical surface in a brain MRI in new slices not visited by the user. (a-d) Five contours were detected using live-wire on orthogonal slices. (e-f) The results in (a) and (d) are refined by adding additional contours.

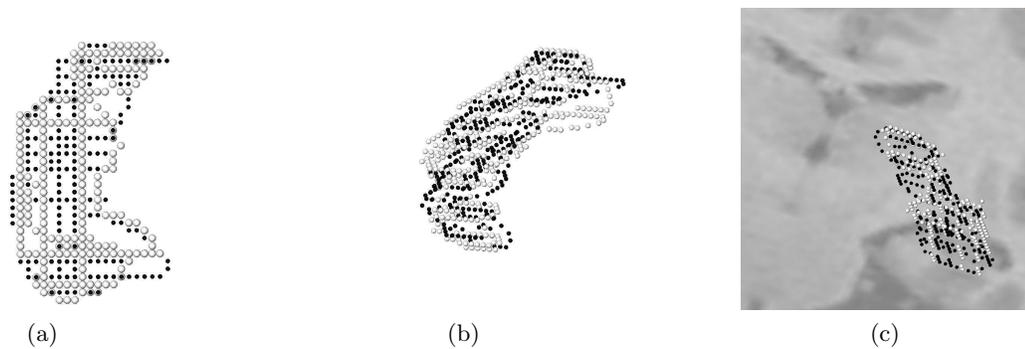


Figure 5. Segmented hippocampus using 3D live-wire. (a,b) show the manual and automatic generated contours in different rotations. (c) shows the segmented hippocampus in the original volume that it was derived from.

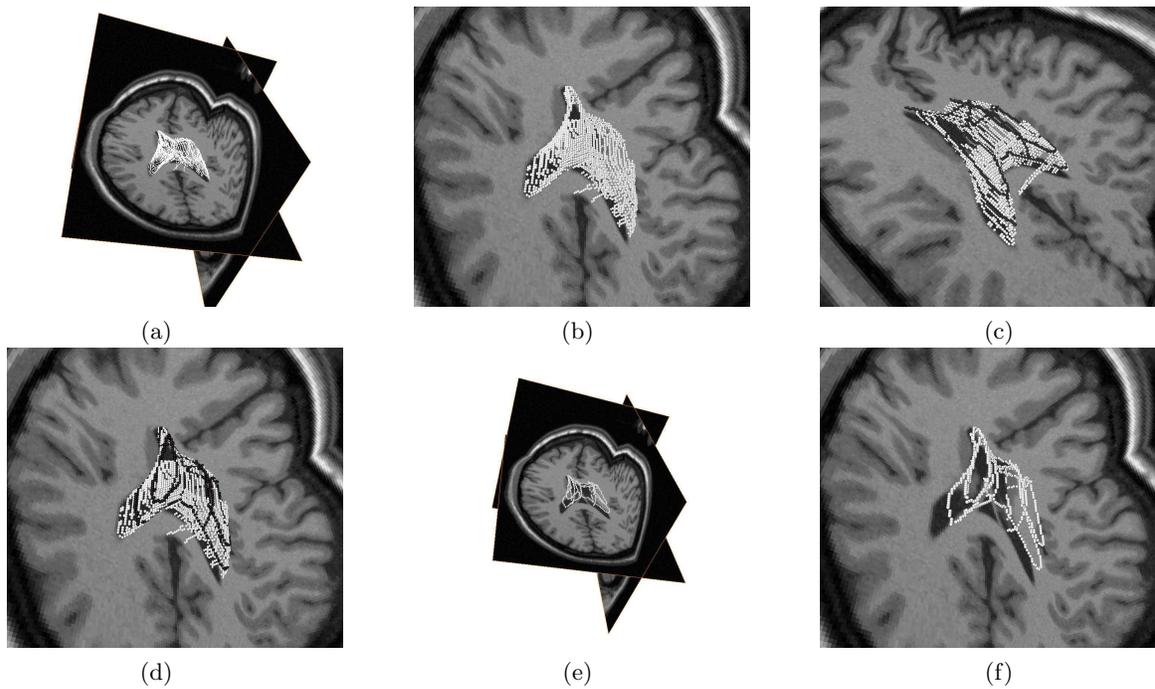


Figure 6. Segmentation of a brain ventricle using 3D live-wire. The 3D segmentation is shown with respect to the original 3D volume. Manual contours are shown in white and additional automatic contours are shown in black.

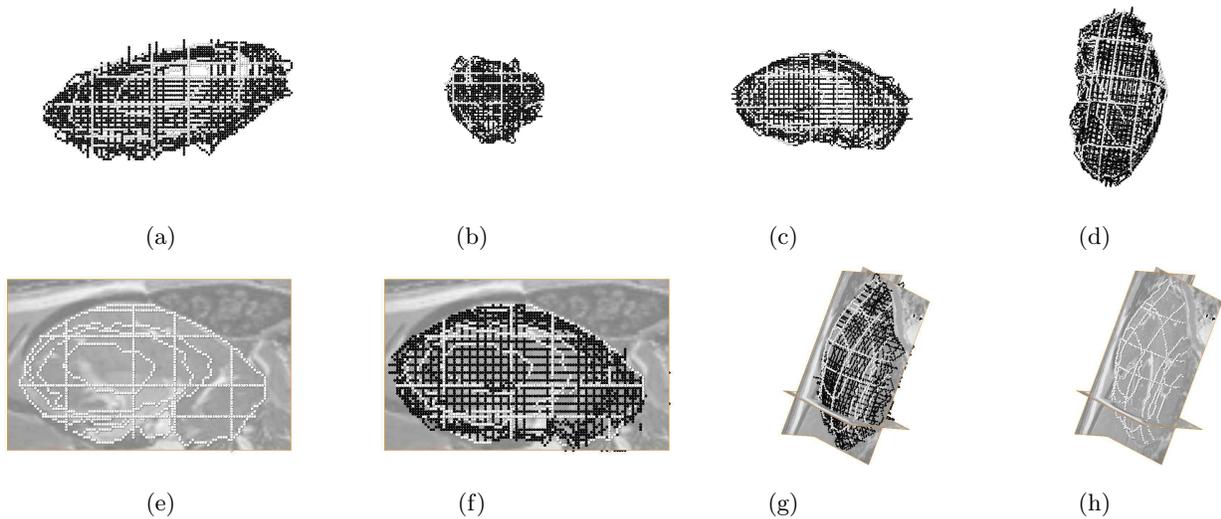


Figure 7. Segmentation of a mouse kidney using 3D live-wire. (a-d) represent different rotations of the segmented kidney. (e-h) show the segmented kidney with respect to the original 3d volume that contained it. Manual contours are shown in white and automatic generated contours are shown in black.

4. CONCLUSIONS

In an effort to create a highly-automated method for segmenting structures from medical images, we adopted live-wire, a robust technique for semi-automatic detection of 2D boundaries, and extended it such that we are able to generate additional segmented contours in new slices automatically. Compared to earlier techniques[11], the presented method offers more freedom in choosing the initial live-wire contours, the user need not scan through many slices to explicitly partition the object beforehand, and finally the data point ordering is no longer restricted to their angle but rather based on a connected graph approach that deals with a wide variety of shapes.

We presented a method for highly automated segmentation of 3D structures based on an extension of the 2D live-wire technique. Our algorithm for ordering intersection points (between live-wires and new slices) is necessary for generating new accurate live-wire contours automatically and is of particular importance for non-start shaped objects with concavities. We demonstrated results on various synthetic as well as real 3D medical data.

5. ACKNOWLEDGEMENTS

GH was partly funded by NSERC (Natural Sciences and Engineering Research Council of Canada) discovery grant and Simon Fraser University Presidents Research Grant Fund. CM and JY were funded by NSERC USRA (Undergraduate Student Research Award).

REFERENCES

1. Y. Kang, K. Engelke, and W. A. Kalender, "Interactive 3d editing tools for image segmentation," *Medical Image Analysis* **8**(1), pp. 35–46, 2004.
2. S. D. Olabarriaga and A. W. M. Smeulders, "Interaction in the segmentation of medical images: A survey," *Medical Image Analysis* **5**(2), pp. 127–142, 2001.
3. M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision* **1**(4), pp. 321–331, 1987.
4. W. A. Barrett and E. N. Mortensen, "Interactive live-wire boundary extraction," *Medical Image Analysis* **1**(4), pp. 331–341, 1996.
5. T. McInerney and H. Dehmshki, "User-defined b-spline template-snakes.," in *MICCAI(2)*, pp. 746–753, 2003.
6. E. N. Mortensen and W. A. Barrett, "Intelligent scissors for image composition," in *Computer Graphics Proceeding, Annual Conference Series*, pp. 191–198, ACM, (Los Angeles, CA), Aug. 1995. SIGGRAPH'95.
7. E. N. Mortensen and W. A. Barrett, "Interactive segmentation with intelligent scissors," *Graphical Models and Image Processing* **60**, pp. 349–384, September 1998.
8. T. Mitsunaga, T. Yokoyama, and T. Totsuka, "Autokey: Human assisted key extraction," in *Computer Graphics Proceedings, SIGGRAPH'95, Annual Conference Series*, pp. 265–272, ACM, (L.A., CA), August 1995.
9. M. Gleicher, "Image snapping," in *Computer Graphics Proceedings, SIGGRAPH'95, Annual Conference Series*, pp. 183–190, ACM, (L.A., CA), August 1995.
10. A. X. Falco, J. K. Udupa, S. Samarasekera, S. Sharma, B. E. Hirsch, and R. de A. Lotufo, "User-steered image segmentation paradigms: Live wire and live lane," *Graphical Models and Image Processing* **60**, pp. 233–260, July 1998.
11. A. X. Falcao and J. K. Udupa, "Segmentation of 3d objects using live wire," in *Medical Imaging: Image Processing, Proc. SPIE* **3034**, pp. 228–235, 1997.
12. J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-8**(6), pp. 679–698, 1986.