Hands-free Interactive Image Segmentation Using Eyegaze

M. Sadeghi, G. Tien, G. Hamarneh, and M. S. Atkins

Medical Image Analysis Lab, Simon Fraser University, Burnaby, BC, V5A 1S6, Canada

ABSTRACT

This paper explores a novel approach to interactive user-guided image segmentation, using eyegaze information as an input. The method includes three steps: 1) eyegaze tracking for providing user input, such as setting object and background seed pixel selection; 2) an optimization method for image labeling that is constrained or affected by user input; and 3) linking the two previous steps via a graphical user interface for displaying the images and other controls to the user and for providing real-time visual feedback of eyegaze and seed locations, thus enabling the interactive segmentation procedure. We developed a new graphical user interface supported by an eyegaze tracking monitor to capture the user's eyegaze movement and fixations (as opposed to traditional mouse moving and clicking). The user simply looks at different parts of the screen to select which image to segment, to perform foreground and background seed placement and to set optional segmentation parameters. There is an eyegaze-controlled "zoom" feature for difficult images containing objects with narrow parts, holes or weak boundaries. The image is then segmented using the random walker image segmentation method. We performed a pilot study with 7 subjects who segmented synthetic, natural and real medical images. Our results show that getting used the new interface takes about only 5 minutes. Compared with traditional mouse-based control, the new eyegaze approach provided a 18.6% speed improvement for more than 90% of images with high object-background contrast. However, for low contrast and more difficult images it took longer to place seeds using the eyegaze-based "zoom" to relax the required eyegaze accuracy of seed placement.

Keywords: computer-aided diagnosis (CAD), feature extraction, human computer interaction, eyegaze tracking, interactive image segmentation, random walker

1. INTRODUCTION

Segmentation is an important part of image processing, which often has a large impact on quantitative image analysis results. Fully automated operator independent segmentation procedures that successfully segment a large number of images are extremely difficult to design. Therefore, usually some kind of operator intervention is required, at least in pathological cases. Therefore, manual interaction, typically using a mouse or keyboard, is inevitable. Given the ever increasing number and resolution of digital images and the need to analyze these images for different applications (e.g. medical imaging and security), manual user interaction is becoming more and more repetitive and resulting in wrist-related injuries. In particular, increasing amounts of repetitive strain injuries (RSI)[1] have emerged from keyboard and mouse overuse. While the keyboard and mouse both contribute to computer-related RSI, most people suffering from RSI find that using the mouse causes more strain and pain than using the keyboard. This impending epidemic of computer-related repetitive strain injuries creates a need to develop a new form of interaction which would help speed up user interaction, minimize risk of wrist injury when processing large amounts of data, and make the interaction more intuitive. Motivated by these observations we designed a new interactive image segmentation approach using a novel form of input - eyegaze.

Eyegaze tracking as a form of input was primarily developed for users who are unable to make normal use of a keyboard or a pointing device. However, with the increasing accuracy and the decreasing cost of eyegaze tracking systems, it is quickly becoming practical for able-bodied users to use gaze as a form of input in addition

Further author information: (send correspondence to M.Sadeghi)

M. Sadeghi: E-mail: msa68@sfu.ca, Telephone: 1-778-782-5509

G. Tien: E-mail: gtien@cs.sfu.ca, Telephone: 1-778-782-5509

G. Hamarneh: E-mail: hamarneh@cs.sfu.ca, Telephone: 1-778-782-3007

M. S. Atkins: E-mail: stella@cs.sfu.ca, Telephone: 1-778-782-4288

to the keyboard and mouse, provided the resulting interaction is an improvement over current techniques.[2,3] This paper explores how gaze information can be effectively used as an augmented input, in addition to traditional input devices like a mouse and a keyboard for user guided image segmentation. We set out to explore the answers to questions such as: Can a user place seeds as labels for the object and the background parts of an image by only looking at the screen? Is it possible to provide a new user interaction environment for interactive image segmentation using the user's eyegaze information? This work is the first step to show the possibility of designing a new user-guided image segmentation method using eyegaze tracking. In particular, a novel joint method combining Random Walker (RW)[4] image segmentation and eyegaze tracking is proposed to demonstrate the potential of this approach.

1.1 Eyegaze tracking

The eyes are a rich source of information for gathering context in our everyday lives. In our daily communication, we use our eyes with minimal effort to convey who, what, or where. A user's gaze is postulated to be the best proxy for attention or intention.[5] Using eyegaze information as a form of input can enable a computer system to gain more contextual information about the user's objective, which in turn can be leveraged to design interfaces which are more intuitive and intelligent. Remote eye-tracking technology is a specialized application of computer vision. A camera is used to track the location of the center of the pupil (p) with reference to the corneal reflection (g) of one or more infrared sources. The reflection point is referred to as a "glint" (figure) 1.[6] Since the surface cornea is nearly spherical, the position of the glint remains more or less fixed as the eye moves to focus on different points-of-regard. Eye-tracking systems use the difference vector (p-g) between the pupil position and the corneal reflection to determine the gaze vector. As evident from figure 1, it is critical to have sufficient resolution on the subject's eye to be able to obtain a reasonable estimate of the gaze vector. This necessitates the need for high resolution imaging sensors in eye tracking systems. One commonly used technique for eye tracking and gaze estimation relies on the observation that when illuminated with infrared light, a person's pupil appears as a bright white dot to an infrared-sensitive camera with an infrared filter installed. The relative positions of the large pupil dot and the smaller glint are used to estimate a vector corresponding to the gaze direction.[7]
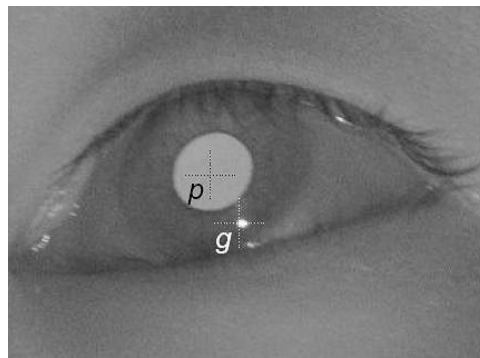

Figure 1. An image of the eye showing the center of the pupil (p) and the corneal reflection (g). The difference vector (p-g) is used to determine the gaze vector.[6]

Human eyegaze tracking is a rapidly growing research area with many useful applications. As the accuracy of gaze tracking and estimation systems improve, this natural way of interacting with the computer screen could provide a substitute means for controlling a computer for those physically or neurologically (e.g. unable to control their movement) impaired persons who find it difficult to, or are completely unable to, control a mouse. Also, it is especially useful for applications in human-computer interaction (HCI), such as video conferencing,[8] where the object on which the user focuses is enhanced or transmitted in higher quality, eye-typing,[9] where a human subject selects letters to be typed by looking at a screen display, and monitoring driver alertness,[10] where a computer system observes the appearance of the driver's eyes to ensure that the driver is not drowsy or otherwise impaired. Eyegaze information has also been used for medical applications in radiology since the 1960s.[11] Eye trackers have been used for medical image perception in radiology for designing radiologists' workstations,[2,12]

to enhance visual search, explore when errors are made and classify the errors according to the eyegaze fixations times. Eye trackers have also been used to understand the nature of expertise in radiology. All these results may be integrated to enable the future development of aids to image perception in radiology. Yang et al.[13] proposed using eye monitoring in many applications in visual search tasks, of which radiology is an important example. In an extension to this work, Atkins et al.[2] showed that eye monitoring is a useful tool to help design and assess radiology workstation interaction techniques: different interfaces affect response time performance through disruption of the visual search process.[2,12]

In addition, eye monitoring has been used in surgery, specifically in minimally invasive surgery (MIS) procedures.[14] Robots can be used for MIS, and simulators are commonly used for training MIS procedures in the laboratory, where eye monitoring systems are used to show the differences between experts and novices.[15] Eye monitoring results may also be used to provide better visualizations for training.[11] Eyegaze tracking in the operating room provides insights into the surgeon's cognitive processes during different surgical steps.[11] A number of exciting emerging applications are being developed to integrate eye monitoring information to provide gaze contingent control in surgery.
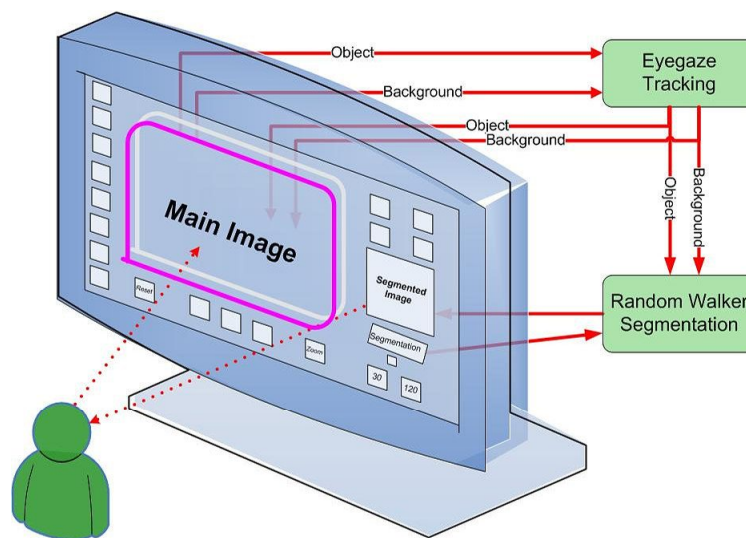

Figure 2. Overview of the proposed eyegaze-based interactive segmentation system.

1.2 Supervised segmentation

Segmentation of objects of interest is a challenging yet important step towards the analysis of digital images. The speed, automation, accuracy and robustness of the segmentation methods often have a large impact on the performance of the application at hand, e.g. computer aided diagnosis. Fully automatic methods sometimes fail, producing incorrect results and requiring the intervention of a human operator. This is often true in medical applications, where image segmentation is particularly difficult due to restrictions imposed by image acquisition, pathology and biological variation. So the level of automation in these methods can be relaxed by incorporating user interaction. The interactive part of a segmentation method is then responsible for mediating information between the user and the computational core of the segmentation algorithm. In such interactive segmentation techniques, the outcome produced by the computational part is translated into visual feedback to the user and the data input by the user are used as parameters for the algorithm. The actual communication between the computer and the user is done via the output and input devices controlled by the user interface. The user analyzes the visual information displayed on the screen and reacts accordingly, providing feedback for the computation. Supervised segmentation algorithms typically operate under one of the three paradigms for guidance (for surveys on interaction and editing in segmentation, the reader is referred to[16] and[17]):

1) Specification of parts of the boundary of the desired object or a nearby complete boundary that evolves to the desired boundary, as in active contours[18–22]

2) Specification of a small set of pixels belonging to the desired object and (possibly) a set of pixels belonging to the background, as in region growing,[23] graph cut,[24] and random walker[4]

3) Specification of starting and ending points along the target object's boundary and a minimal path approach that connect these points with a contour. The contour snaps to the desired boundary and wraps around the object, as in "live wire" and intelligent scissors.[25–28]

Other common types of user interaction include setting the parameter values (e.g. degree of regularization) or choosing from preset options in a menu.

The Random Walker[4] algorithm is a recently-developed, general-purpose interactive multi-label image segmentation method that allows the user to select background and objects' seeds. Given a small number of pixels with predefined labels, one can quickly determine the probability that a random walker starting at each unlabeled pixel will first reach one of the pre-labeled pixels. By assigning each pixel to the label for which the greatest probability is calculated, an image segmentation is obtained. Theoretical properties of this algorithm are developed in[4] along with the corresponding connections to discrete potential theory and electrical circuits. This algorithm is formulated on a graph using combinatorial analogues of standard continuous domain operators and principles. In this work, we adopted the RW method for interactive image segmentation given its optimality properties; ensuring global minima of the segmentation objective functional, and its ability to segment multiple objects as opposed to only binary segmentation. Figure 3 shows an example skin lesion image, the seeds used, the probability map and the resulting RW segmentation.

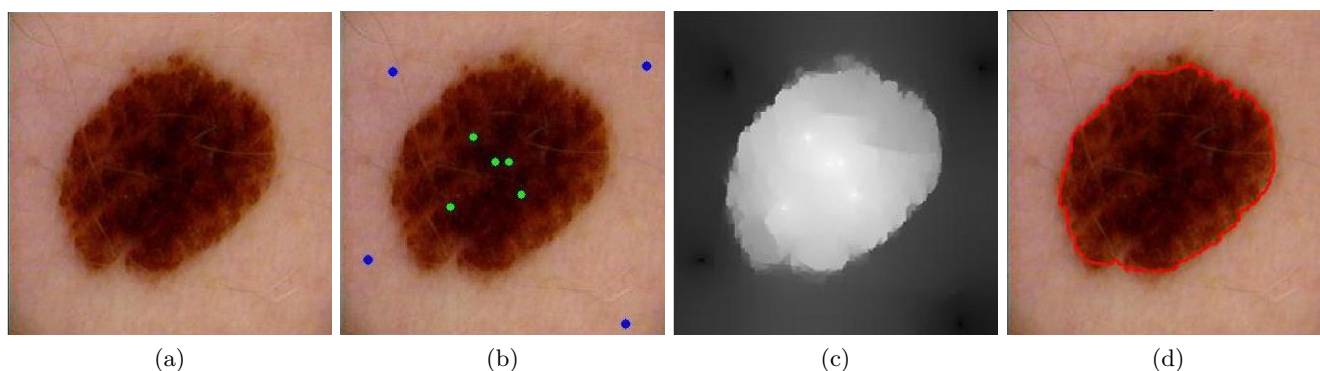| (a) | (b) | (c) | (d) |

Figure 3. An example of RW segmentation. (a) a skin lesion image; (b) the object and background seeds; (c) the probability map (of assigning a pixel to the lesion as opposed to the background) and (d) the segmentation result using RW.

The aim of this study is to compare two interactive methods for achieving user-guided image segmentation using the RW algorithm: (i) a mouse-based clicking method, and (ii) an eyegaze-based clicking method.

2. METHOD

2.1 System overview and the graphical user interface

A user imposes certain hard constraints on the segmentation by indicating certain pixels (seeds) that absolutely have to be part of the object and certain pixels that have to be part of the background. Intuitively, these hard constraints provide clues on what the user intends to segment. The user analyzes the visual information displayed on the screen and reacts accordingly, providing feedback to be used as input for the core RW segmentation algorithm. Figure 2 shows an overview of the developed system. The algorithm involves placing background "seeds" (colored red, as shown later on, in figure 4 and subsequent figures) on the image background, and object seeds (green) on the object to be segmented. Seed points appear as 8-pixel-wide circles and can be placed by clicking on the desired location. In eyegaze mode, the user "clicks" by staring at a spot for about a second. This fixation duration can be altered; typically as users become more experienced with using the system, this time can be decreased. To be completely hands-free in eyegaze mode, the large user interface buttons can also be "clicked" by staring at the button briefly. There are three modes: *Zoom*, *Add Seed*, and *Delete Seed*. In *Zoom* mode, clicking on the unzoomed image displays a portion of the image in 4× magnification, centered around
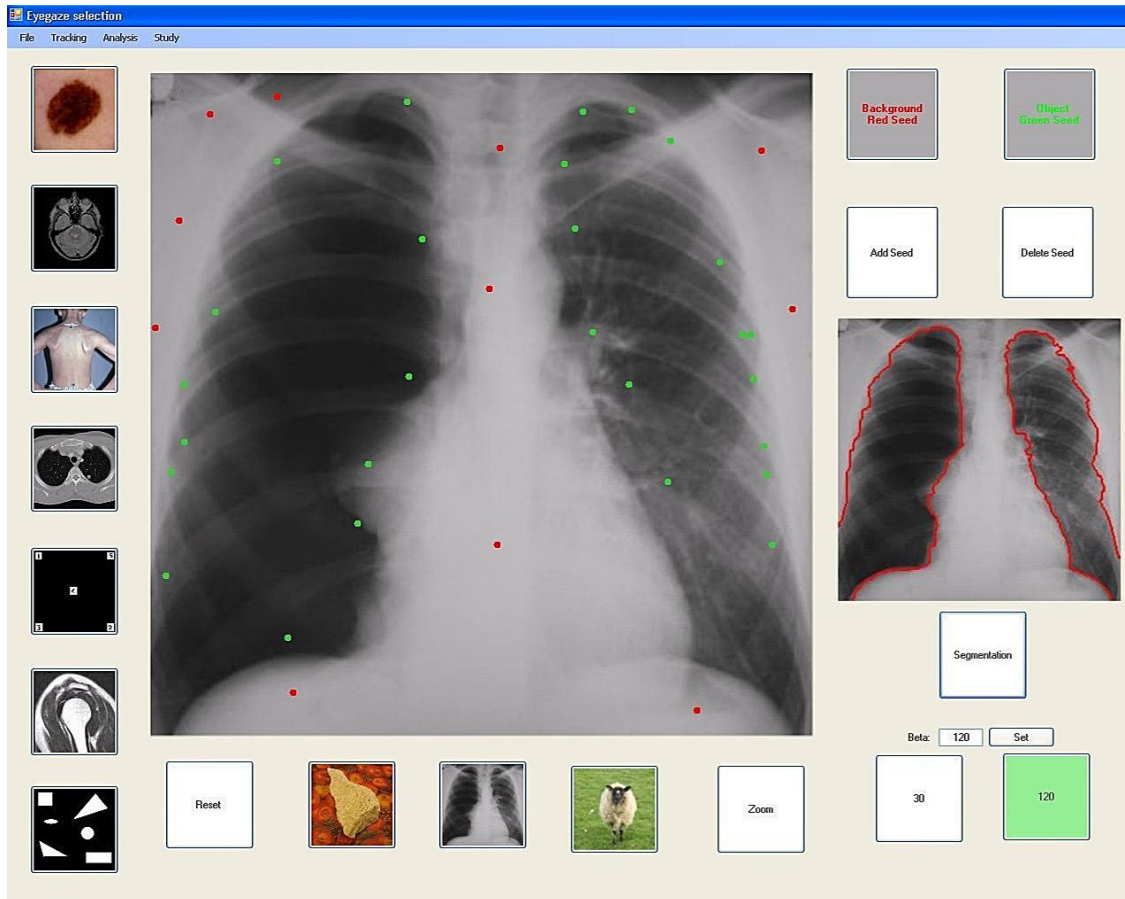
Figure 4. The graphical user interface developed for our experiments: red dots (dark color) are background seeds and green dots (light color) are object seeds. The segmented image appears at the right side of the interface.

the clicked point. Clicking on the zoomed image returns the view to the unzoomed image. In *Add Seed* mode, clicking on the image adds a seed to the clicked point. The type of seed (object or background) placed depends on whether the *Object* or *Background* button was clicked beforehand. In *Delete Seed* mode, clicking on the image removes any seed within a 25-pixel radius from the clicked point. The type of seed removed depends on whether the *Object* or *Background* button was clicked beforehand.

Using our interface, the type of seed must first be set by clicking on either the *Background* or the *Object* button. Once that is done, clicking *Add Seed* or *Delete Seed* and followed by clicking at the desired location(s) on the image will perform the corresponding action. All currently placed seeds can be removed by clicking *Reset*. Figure 4 shows the graphical user interface that we have used for our experiments. The user may click on the *Zoom* button and then click on the desired location to zoom in for more precision in placing seed points. The zoom automatically returns to the original view after one seed has been placed. When the user is satisfied with the placement of seed points, the user clicks on the *Segmentation* button. The segmentation result will be displayed after a few seconds in the segmented image box. If the user is not satisfied with the segmentation, he or she can either place more seeds and try again or adjust the segmentation parameter (beta) which is also accessible from the user interface.

2.2 Random walker interactive segmentation

At a high level, RW segmentation steps are :

- The user specifies seed points and the seeds are labeled as either object or background [*];

- The image is modeled as a graph where image pixels are represented by graph nodes;

- Graph edges connect neighboring pixels;

- The weight of an edge is set as a function of the intensity difference between a pixel and its neighbor. This function or mapping is controlled by a parameter *beta*;

- The probability that a random walker starting from a particular pixel reaching any of the labeled pixels (seeds) is computed for every pixel by solving a linear system of equations;

- The maximum probability label is assigned to each pixel, which constitutes the image segmentation.

More details can be found in.[4]

2.3 Details of eyegaze tracking and program logic

The eyegaze system, is a hands-free, remote human-computer interface that can be used to track a user's gaze point or allow an operator to interact with their environment using only their eyes. The eyegaze information may be recorded as a set of locations or regions of the image weighted by a fixation time, or the gaze information may be used as the basis to identify a region of the image for further analysis to determine its content. Alternatively, the gaze information may be used to define a region of interest (ROI) within the image for further analysis, such as identifying a polygonal region around the gaze region(s).
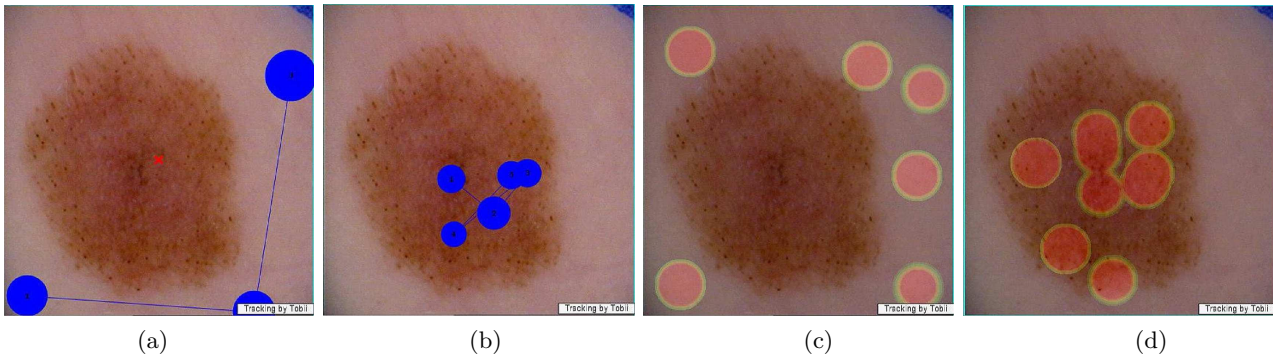

| (a) | (b) | (c) | (d) |

Figure 5. Eyegaze plot of (a) background seeds and (b) objects seeds placed by subject1. Eyegaze hot spot map of (c) background seeds and (d) object seeds placed by subject2.

3. RESULTS

3.1 Experimental setup

For this research, we have used Tobii 1750[†] eye tracker. We had two sudies; For first one, we used Tobii development kit and ClearView software[‡] and the second one is based on our implemented system. Figure 5 shows plots and hot spot maps of eyegaze information recorded for two different subjects as they label the object and background parts of a skin lesion image. These eyegaze plots and hotspots are a feature of the first study implemented with the Tobii and ClearView software analysis suite, and are not available in the custom interface of the second study (only raw eyegaze data are available). We have presented results of the second study in this paper. Seven subjects (computer science graduate students) participated in this study. Using the provided interface, they were asked to place both background and object seed points to segment several

---

[*]A recent version of RW that does not require seeds is proposed in[29]

[†](http://www.tobii.com)

[‡](http://www.tobii.com/market_research_usability/products_services/tobii_software_development_kit.aspx)

images. Before beginning, subjects watched a demonstration of how to use eyegaze to place the seed points and perform the image segmentation, then were given several practice trials to familiarize themselves with using the interface in both modes of interaction. Figure 6 shows the segmentation results of the training stage for one of the subjects. Each subject was randomly assigned to start with the mouse or with the eyegaze input and each subject segmented 5 images with each mode of interaction (see example images in left panel of GUI in Figure 4). The five images were divided in three categories according to the ease of segmentation. Two images were in the easy category, two images were in the medium difficulty category and one was a difficult image to segment. The level of difficulty reflects weak boundaries, narrow protrusions or indentations, holes in the object, etc. The subjects' eyegze movements and/or mouse clicks were recorded for our analysis.

Figure 6. The segmentation results of the three images in the training stage for one of the subjects (using eyegaze tracking).

3.2 Eyegaze calibration

To ensure better eyegaze tracking accuracy, participants were required to undergo a short (40 seconds) calibration procedure before using the eyegaze interface. The calibration procedure involves having the system display a number of points on the screen. This can be as low as only two calibration points, but will still yield high accuracy across the entire tracking range. The calibration also uses automatic filters to select good data points. These features enable very simple and quick calibrations, even on difficult subjects such as infants and low-vision subjects.

All eyegaze interactions are performed using simulated mouse clicks, so the entire interface is also accessible using a standard mouse. All the gaze data is recorded to a plain text file including left and right eye positions, gaze position (x_g, y_g), head position (x_h, y_h, z_h), and pupil diameter (Φ). This data can be used to replay a recorded session. The gaze location is shown as a green crosshair on the screen, averaged over the 6 previous gaze locations.

In addition to setting seed points on an image, eyegaze can be used to click on buttons. The button changes colour when the user gazes at it, and the button is clicked when the button has accumulated 40 gaze samples. To interact with the main image, a simulated mouse click is made at each eyegaze fixation point. Our real-time fixation detection algorithm triggers if M of the previous N gaze samples lie within a D pixel radius of the average position of the previous N samples. M and D are user configurable and were set to 20 samples and 50 pixels, respectively. N is set to 1.5 times M. The action performed by the mouse click depends on the click interaction mode.

3.3 Qualitative and quantitative results

Figures 7 and 8 show sample images classified into *Easy*, *Medium* and *Difficult*, with overlaid sample segmentation results (red contours) obtained via eyegaze or mouse-based interaction. Note the high similarity of segmentation results. Figure 9 (a) shows that for the *Easy* category images, eyegaze-based segmentation is faster than using the mouse; for *Medium* images eyegaze and mouse were comparable; and for *Difficult* images using the mouse achieves faster segmentation compared to eyegaze. Since the eyegaze input requires no time for acquiring the cursor position by making manual motions (such as a hand motion to get the mouse), but involves a short dwell

time, we have observed that the performance of the two input approaches is similar. If we isolate the actions the user must perform to point and click on a target with the mouse, the total time would be:

$$T_{mouse-click} = T_{locate-target} + T_{grab-mouse} + T_{locate-cursor} + T_{move-mouse} + T_{click-mouse} \qquad (1)$$

By contrast, the total time for selection using eyeclick would be

$$T_{eye-click} = T_{locate-target} + T_{dwell-time} \qquad (2)$$

where $T_{locate-target}$ is the amount of time it takes the user to conduct a visual search for the target. It can be reasonably expected that the time to locate the target is the same in both approaches. Equations (1) and (2) demonstrate why segmentation time for images of the *Easy* and *Medium* category in the eyegaze mode is less than or close to the segmentation time of images in mouse-click mode. For images of the *Difficult* category, since locating the targets is not easy and the user has to be very accurate to place a seed in narrow or small parts of the image by eyegaze, $T_{eye-click}$ is larger than $T_{mouse-click}$. Further, our experiments proved that due to the low accuracy of placing seeds for images of the *Difficult* category, users needed a larger number of corrections for misplaced seeds and, hence, they had to repeat the segmentation until the desired result is obtained.

Figure 9(b) presents the average number of clicks to zoom, delete seeds, and place background and object seeds. For images of the *Difficult* category, due to the high error rate and necessity of editing misplaced seeds to locate accurate one, there is an obvious large difference between the number of different clicks in eye-click and mouse-click modes, which is the most important reason for low speed of eye-click mode rather than mouse-click mode for the *Difficult* category. Also, there is some inter-subject variability of the eye tracking accuracy due to calibration drift and changes in subject posture over the course of the experiments.

(Easy)   (Easy)

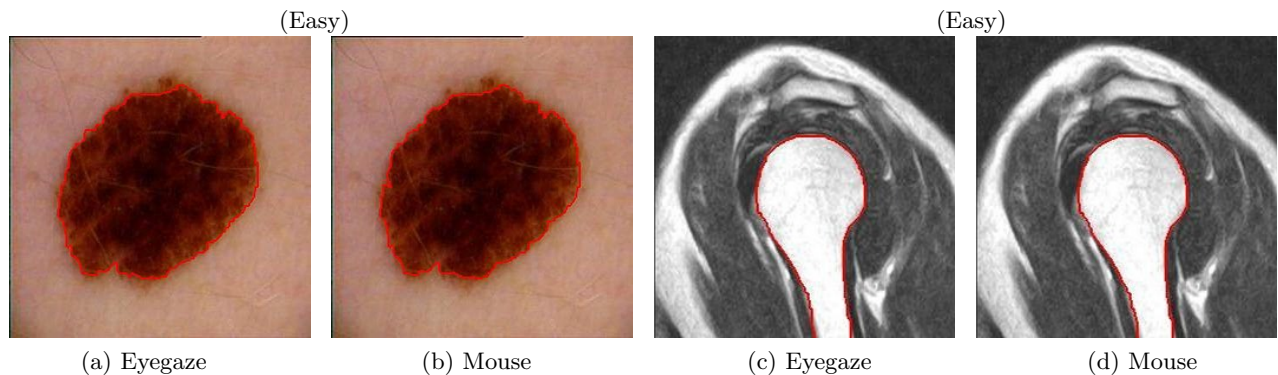(a) Eyegaze   (b) Mouse   (c) Eyegaze   (d) Mouse

Figure 7. Eyegaze segmentation vs. mouse segmentation for the images categorized in *Easy* category. (a) and (b) Skin lesion segmentation, (c) and (d) Shoulder bone image segmentation.

4. CONCLUSION

The study statement and hypothesis of this work is: "Gaze information can be used as a form of input for interactive image segmentation". It is the first work in the interactive image segmentation field which explores the design space of interaction techniques that use gaze information for placing object and background seeds. We present the results of user experiments which compare a novel gaze-augmented interaction technique with a traditional mouse based interaction. To the best of our knowledge, this is the first study that examines the use of eyegaze tracking and fixation-based clicking as user input in a graph-theoretic interactive image segmentation technique. We show that the resulting interaction is either comparable to or an improvement over existing input methods and that it is indeed possible to devise novel interaction techniques that use gaze as a form of input for interactive image segmentation, even given the unfamiliarity of the subjects with the eyegaze interface compared to the mouse. Clearly, an important advantage is minimizing wrist strain for repetitive tasks. With the random walker's ability to segment multiple objects, we can now explore multi-label segmentation scenarios

(Medium)   (Difficult)

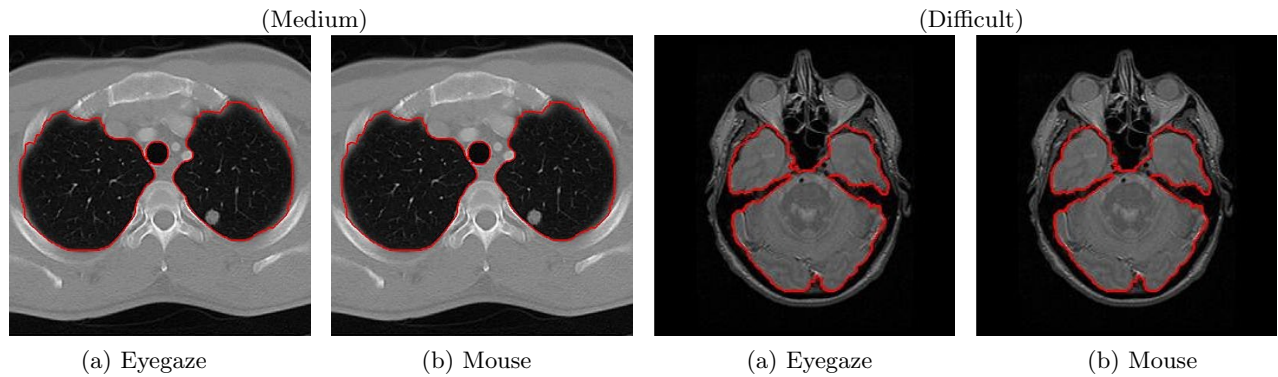(a) Eyegaze   (b) Mouse   (a) Eyegaze   (b) Mouse

Figure 8. Eyegaze segmentation vs. mouse segmentation, (a) and (b) Lung CT image segmentation categorized in *Medium* category, (c) and (d) brain image segmentation categorized in *Difficult* category.

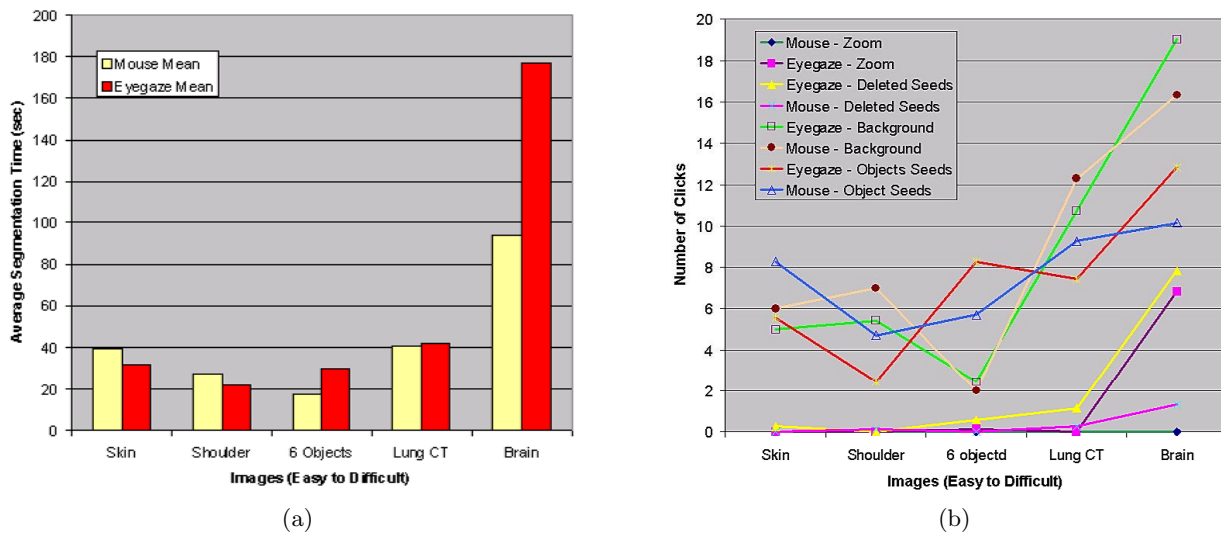(a)   (b)

Figure 9. (a) Average segmentation time for the images used in our study, (b) average number of clicks for background and object seeds placement, zoom and delete (eyegaze vs. mouse)

using eyegaze. We are also working on improving the zoom feature as well as investigating other interactive segmentation methods, and improving the user interface further. In future work, we will explore if eyegaze may produce other forms of strain or negative side effects.

REFERENCES

[1] Damany, S. and Bellis, J., [*It's Not Carpal Tunnel Syndrome! RSI Theory and Therapy for Computer Professionals*], Simax, Philadelphia (2000).

[2] Moise, A., Atkins, M. S., and Rohling, R., "Evaluating different radiology workstation interaction techniques with radiologists and laypersons," *Journal of Digital Imaging* **18**(2), 116–130 (2005).

[3] Tien, G. and Atkins, M. S., "Improving hands-free menu selection using eyegaze glances and fixations," in [*ETRA '08: Proceedings of the 2008 symposium on Eye tracking research & applications*], 47–50, ACM, New York, NY, USA (2008).

[4] Grady, L., "Random walks for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **28**(11), 1768–1783 (2006).

[5] Zhai, S., "What's in the eyes for attentive input," *Commun. ACM* **46**(3), 34–39 (2003).

[6] Kumar, M., *Gaze-enhanced user interface design*, PhD thesis, Stanford University, Department of Computer Science (2007).

[7] Ohno, T. and Mukawa, N., "A free-head, simple calibration, gaze tracking system that enables gaze-based interaction," in [*ETRA '04: Proceedings of the 2004 symposium on Eye tracking research & applications*], 115–122, ACM, New York, NY, USA (2004).

[8] Machin, D., Xu, L.-Q., and Sheppard, P., "A novel approach to real-time non-intrusive gaze finding," 428–437, British Machine Vision Conference (1998).

[9] Johansen, A., Hansen, D., Hansen, J., and Nielsen, M., "Eye typing using markov and active appearance models," 132–136, Sixth IEEE Workshop on Applications of Computer Vision (2002).

[10] Ji, Q. and Yang, X., "Real-time eye, gaze, and face pose tracking for monitoring driver vigilance," *Real-Time Imaging* **8**(5), 357–377 (2002).

[11] Atkins, M. S., Nicolaou, M., and Yang, G.-Z., [*Passive Eye Monitoring*], ch. Eye Monitoring Applications in Medicine, 323–343, Signals and Communication Technology, Springer Berlin Heidelberg (2008).

[12] Atkins, M. S., Moise, A., and Rohling, R., "An application of eyegaze tracking for designing radiologists' workstations: Insights for comparative visual search tasks," *ACM Trans. Appl. Percept* **3**(2), 136–151 (2006).

[13] Yang, G.-Z., Dempere-Marco, L., Hu, X.-P., and Rowe, A., "Visual search: psychophysical models and practical applications," *Image and Vision Computing*, 273–287 (2002).

[14] Nicolaou, M., James, A., Darzi, A., and Yang, G.-Z., "A study of saccade transition for attention segregation and task strategy in laparoscopic surgery," in [*MICCAI (2)*], 97–104 (2004).

[15] Law, B., Atkins, M. S., Kirkpatrick, A. E., and Lomax, A. J., "Eye gaze patterns differentiate novice and experts in a virtual laparoscopic surgery training environment," in [*ETRA '04: Proceedings of the 2004 symposium on Eye tracking research & applications*], 41–48, ACM, New York, NY, USA (2004).

[16] Olabarriaga, S. D. and Smeulders, A. W. M., "Interaction in the segmentation of medical images: A survey," *Medical Image Analysis* **5**(2), 127–142 (2001).

[17] Kang, Y., Engelke, K., and Kalender, W. A., "Interactive 3d editing tools for image segmentation," *Medical Image Analysis* **8**(1), 35–46 (2004).

[18] Kass, M., Witkin, A., and Terzopoulos, D., "Snakes: Active contour models," *International Journal of Computer Vision* **1**(4), 321–331 (1987).

[19] Chan, T. F. and Vese, L. A., "Active contours without edges," *Image Processing, IEEE Transactions on* **10**(2), 266–277 (2001).

[20] Cootes, T. F., Taylor, C. J., Cooper, D. H., and Graham, J., "Active shape models—their training and application," *Comput. Vis. Image Underst.* **61**(1), 38–59 (1995).

[21] Pluempitiwiriyawej, C., Moura, J. M. F., Wu, Y.-J. L., and Ho, C., "Stacs: new active contour scheme for cardiac mr image segmentation," *Medical Imaging, IEEE Transactions on* **24**(5), 593–603 (2005).

[22] Goldenberg, R., Kimmel, R., Rivlin, E., and Rudzsky, M., "Fast geodesic active contours," *Image Processing, IEEE Transactions on* **10**(10), 1467–1475 (2001).

[23] Adams, R. and Bischof, L., "Seeded region growing," *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **16**(6), 641–647 (1994).

[24] Boykov, Y. and Jolly, M. P., "Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images," *Proc. ICCV*, 105–112 (2001).

[25] Mortensen, E. and Barrett, W., "Interactive segmentation with intelligent scissors," *Graphical Models in Image Processing* **60**(5), 349–384 (1998).

[26] Sethian, J., "A fast marching level set method for monotonically advancing fronts," in [*Proc. Nat. Acad. Sci.*], **93**(4), 1591–1595 (1996).

[27] Cohen, L. and Kimmel, R., "Global minimum for active contour models: A minimal path approach," *International Journal of Computer Vision* **24**, 57–78 (1997).

[28] Barrett, W. A. and Mortensen, E. N., "Interactive live-wire boundary extraction," *Medical Image Analysis* **1**, 331–341 (1997).

[29] Grady, L., "Multilabel random walker image segmentation using prior models," in [*Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*], **1**, 763–770, IEEE Computer Society, Washington, DC, USA (2005).