Automatic Zoom and Pseudo Haptics to Support Semiautomatic Segmentation Tasks

Wolfram Schoor
Fraunhofer Institute for Factory Operation and Automation (IFF)
Sandtorstrasse 22,
39106 Magdeburg, Germany
wolfram.schoor@iff.fraunhofer.de

Felix Bollenbeck
Leibniz Institute of Plant Genetics and Crop Plant Research (IPK)
Corrensstrasse 3,
06466 Gatersleben, Germany
bollen@ipk-gatersleben.de

Marc Hofmann
Fraunhofer Institute for Factory Operation and Automation (IFF)
Sandtorstrasse 22,
39106 Magdeburg, Germany
marc.hofmann@iff.fraunhofer.de

Rüdiger Mecke
Fraunhofer Institute for Factory Operation and Automation (IFF)
Sandtorstrasse 22,
39106 Magdeburg, Germany
ruediger.mecke@iff.fraunhofer.de

Udo Seiffert
Leibniz Institute of Plant Genetics and Crop Plant Research (IPK)
Corrensstrasse 3,
06466 Gatersleben, Germany
seiffert@ipk-gatersleben.de

Bernhard Preim
Otto von Guericke University,
Universitätsplatz 2,
39106 Magdeburg, Germany
preim@isg.cs.uni-magdeburg.de

ABSTRACT

This paper presents a new technique for navigating large amounts of biological image data during segmentation. Digitized microtome tissue slices of barley grains constitute the image data. Automatic zoom is dependent on mouse speed, thus enabling users to navigate an image more efficiently and with a tighter focus during segmentation. The user remains oriented and smoothly moves through the data. Furthermore, pseudo haptic feedback based on image cost map data compensates for user inaccuracies generated, for example, by shaky hands. A prototype was implemented and tested. An informal usability study revealed that users are inclined to prefer automatic zooming and pseudo haptics for semiautomatic segmentation tasks.

Keywords: Interaction techniques, zoom techniques, pseudo haptic, segmentation.

1 INTRODUCTION

Navigation techniques support the exploration of huge information spaces by mapping a subset of information onto the limited screen space with the aim of plausibly mapping transitions between successive subsets to meet user requirements. This can be achieved by panning (scrolling) and zooming in the two-dimensional continuous space. Zooming allows a user to view specific targets on different scales. Panning can be used to visit different locations on the same scale. Panning and zooming are the de facto standard for the navigation of large information spaces [vWN03].

Typical interfaces for manual and semiautomatic image data segmentation lack effective navigation. Users have to zoom in for detailed segmentation in difficult regions and then zoom out and pan for rapid segmentation in regions with clearly defined edges. Using GUI buttons to adjust the zoom to the required scale increases the time expended and can waste additional resources. A rate-based scrolling interface such as [ZSS97] can map input device speed to zooming speed. This is done with a mouse wheel for instance. While these interfaces require less time, they can also cause disorientation in the information space [JF98]. Another important aspect of navigation is the efficient mapping of a user's interaction onto the screen. Pseudo haptic feedback [LBE04], [DLB$^+$05] can enhance user interaction by supporting a user on the basis of the underlying image data. Undesired movements such as hand tremors are diminished. Speed and accuracy increase. Figure 1 is an image for such a segmentation task.

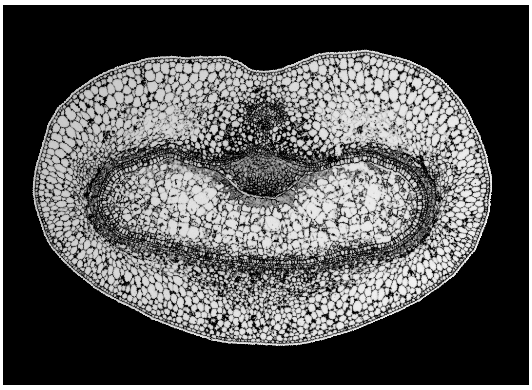

Figure 1: Slice of *Hordeum vulgare*

WSCG 2008 conference proceedings
WSCG'2008, February 4 – February 7, 2008
Plzen, Czech Republic.

2 RELATED WORK

2.1 Complex Image Data Segmentation

Different segmentation methods are employed in different fields of application. The live wire technique [SPP00], [GDGC02], also known as intelligent scissors [MB99], [MB98], was selected for this research.

All image segmentation algorithms are intended to deconstruct an image into segments with specific *meaning* for a given application [HS85] or, expressed more technically, to identify regions that *uniformly* relate to a specific criterion such as texture or image intensity [HS85]. The literature, specifically [LM01], [Hah05] or [Kan05], presents a variety of classification approaches. With regard to interaction, this paper classifies image segmentation according to the following techniques.

Manual segmentation is the easiest image segmentation technique and is commonly used for high level segmentation tasks, especially when a priori knowledge of the image data is required. The user specifies a set of control points that will be connected afterward. The contour can additionally be smoothed by familiar interpolation techniques [Far88]. The main disadvantage of this technique is the time and concentration that precise segmentation requires since numerous control points have to be specified manually.

Automatic segmentation algorithms can be far more efficient than manual segmentation but can also fail to correctly segment the objects of interest, resulting in an inacceptable loss of accuracy. The advantage of these algorithms is that they do not require user input for segmentation. Calculation time depends on the complexity of the algorithm and ranges from real time to several hours [BBS$^+$06]. Lucheese et al. [LM01] and Hahn [Hah05] survey automatic methods, including histogram threshold, boundary-based, clustering and neural network algorithms.

Semiautomatic segmentation techniques represent a compromise between powerful low level segmentation algorithms and the knowledge required of the user. The user has to set a limited number of parameters specific to the dataset or support the segmentation process with preexisting knowledge. Semiautomatic methods function efficiently and accurately where automatic methods fail or cannot be applied. While semiautomatic algorithms are more time consuming than most automatic algorithms, the results of the segmentation process can nevertheless be used as training data for other automatic segmentation algorithms. Semiautomatic algorithms include region growing [AB94], [KWT88] and live wire [MB98], [SPP00]. Given the nature of the datasets considered here, difficult to predict shapes like those created by snakes are not adequate. Apparent gaps in the shapes also make region growing unfeasible. Since it is more easily controlled and more predictable, interactive segmentation that iteratively generates the correct curve segments promises better results. For these reasons, live wire was chosen for further investigations of semiautomatic segmentation algorithms.

2.2 Navigation and Interaction Techniques

The datasets consist of large and highly detailed images. The goal of a semiautomatic segmentation application is to enable users to access data with different levels of detail easily and efficiently as well as to maintain an overview of the data. The limited screen space necessitates different approaches, all of which should fulfill Shneiderman's maxim of visual seeking: "Overview first, zoom and filter, then details on-demand." [Shn96].

One such method is the space deformation approach. All information is visible simultaneously and important areas become magnified similar to the fisheye technique [Fur86]. The advantage of this approach is its simultaneous provision of focus and overall context. However, the magnification remains bound to the position of the cursor. Large scales produce substantial distortions around a magnified area. Nonetheless, this information may be important for the next path segments. The zoom's amplitude of magnification must be adjusted manually.

Another method is the information deformation technique [AS07], [WGCO07], which shrinks an image space content-aware to fit the screen space. One problem with these algorithms is that only limited information loss can be compensated without losing global context or experiencing undesired effects. Thus, global information can be mapped to the limited screen space only up to a low zoom level.

An overview window can generate focus and context within the representation of an image on a specific zoom level. Again, other zoom levels must be adjusted manually.

These methods can be supported by speed-dependent automatic zooming (SDAZ) as first set forth by Igarashi and Hinckley [IH00] in 2000, although already in use in such games as GTA (DMA Design 1997). Other examples of SDAZ are [vWN03], [vWN04] or [Kre05]. The main challenge is automatically adjusting the zoom level on the basis of a user's speed so that the user can navigate within the image data with a tighter focus and fully concentrate on the task at hand without being too distracted by navigation. SDAZ has improved navigation speed in numerous applications. Moreover, its fewer different interaction steps make it user friendly. Smooth transitions between successive subsets of information space are essential to render the relation to movement and the global information space comprehensible to the user.

Another well known interaction approach is pseudo haptic feedback as applied in [LBE04] or [DLB$^+$05].

The change in the control to display ratio (C/D ratio) conveys an impression of a surface's structure or ruggedness to users, similar to sensing bumps or hills. The well established principle of gravity active points [BS86] is closely related to pseudo haptics and was used to position objects precisely.

3 DATA AQUISITION

Each of the datasets segmented consisted of approximately 2000 slices of a barley corn (*Hordeum vulgare*) $3\,\mu m$ high. Captured with a color CCD camera, the images originally measured 1600 x 1200 pixels with a pixel size of $1.83 \times 1.83\,\mu m$. Given the very high correlation of the color values (see the line approximation in the RGB space in Figure 2), the images were converted to a gray scale with a depth of 8 bits for better performance.

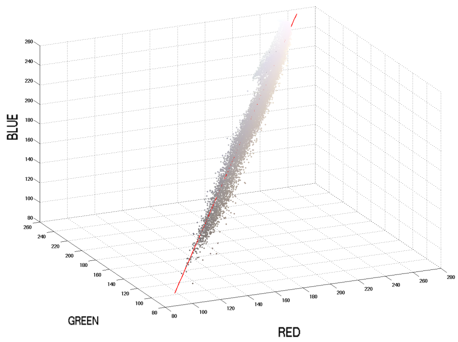

Figure 2: Distribution of the color values

In addition, automatic background elimination and alignment were applied to the datasets [GDB+07]. This reduced the size of each dataset to 5 GB.

An initial analysis of the data and related datasets revealed a high degree of complexity in the images. The image data contains similar, no or only partially continuous edges and the intensity changes in many images. Only vague information exists about the resulting structures being segmented (see Figure 1 and especially the areas encircled in red in Figure 3).

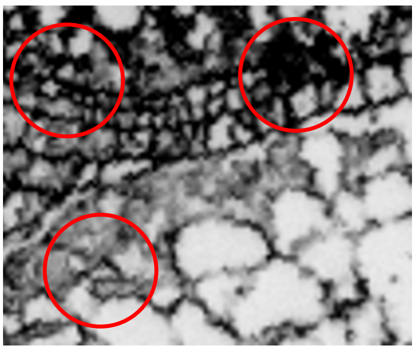

Figure 3: Detail of a highly complex slice of a microtome tissue

Figure 4 illustrates the complete segmentation of an image slice, manually segmented with the commercial visualization software package Amira [Kon07]. The total segmentation time for all tissues in Figure 4 was approximately 23 minutes.

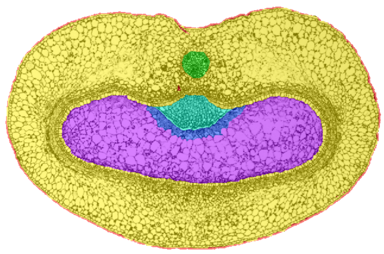

Figure 4: Complete segmentation of all basic tissues

4 AUTOMATIC ZOOMING AND PSEUDO HAPTIC FEEDBACK

Concepts were formulated and implemented to adapt and enhance the interaction and navigation techniques relevant for image segmentation.

4.1 Concept

Automatic zooming involves identifying constraints that describe the relation between zooming and cursors speed. To calculate the zoom level of the next time step (see Equation 1), certain terms must be taken into account. The preceding zoom level $Zoom_{OLD}$ is scaled with a $Speed_{AVG}$ term averaging cursor speed over a distinct time interval, a $Trend$ term calculated from recent time intervals and a $Prop$ term derived from the costs of the propagated path in the cost map.

$$\begin{aligned} Zoom_{NEW} = Min(0, (Max(1, \\ (Zoom_{OLD} \cdot Speed_{AVG} \cdot Trend \cdot Prop))) \end{aligned} \tag{1}$$

Another key concept is the integration of pseudo haptic feedback in segmentation for which the control to display ratio is an important term, which describes the relationship between the movement of the control device (e.g. mouse) and the actual movement of the cursor on the screen. The control to display ratio's nonlinear anisotropic behavior supports users when they are performing difficult segmentation tasks. As opposed to the calculation of segmentation in the direction of the gradient (orthogonal to the direction of the edge) with a larger C/D ratio, segmentation along an edge is calculated with a fixed C/D ratio. Consequently, users must amplify their movements to obtain the same results from cursor movement in the screen space. This eliminates small, undesired user hand motions such as shaking in the direction of segmentation. Since user speed is related to cursor movement, this technique indirectly influences the automatic zoom. Since the C/D ratio is initially a fixed value in a system, only changes of the C/D ratio must be calculated. This can be done based on cursor position (Equation 2).

$$CDR_{NEW} = \alpha \cdot \sqrt{\delta v^2 + Cost_{NORM}^2} \quad (2)$$

where δv is the magnitude of the direction change of the cursor in a time step, α a constant indicating the scale factor and $Cost_{NORM}$ the normalized costs from the cost map (see Equation 3). Figure 5 visualizes the change of the C/D ratio, the red arrows illustrating the movement of the cursor in the 2-D information space and the blue arrows indicating the respective movement of the input device necessary. The influence of the 3-D cost distribution on the change of the C/D ratio during segmentation counter the direction of the edge (i.e. uphill) is easily recognizable.

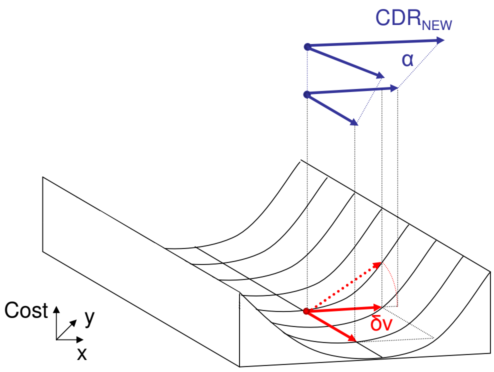

Figure 5: C/D ratio increases in relation to the image costs

4.2 Implementation

The algorithms were implemented in C++ and the graphic user interface was generated with WxWidgets 2.8[1].

The live wire algorithm was implemented largely based on [GDGC02]. Good results were achieved by using the parameters Laplace with a weight *wL* of 20 (5*x*5 and 9*x*9 filter), gradient magnitude with a weight *wG* of 30 and the point-line distance with a weight *wD* of 1 to generate the cost map as in Equation 3.

$$\begin{aligned} Cost = {} & wL \cdot LaPlaCost \\ & + wG \cdot GradientCost \\ & + wD \cdot (0.25 \cdot PointLineDist^2) \end{aligned} \quad (3)$$

Furthermore, an image pyramid was used to speed up algorithm performance to determine initial cost. Moreover, an exact cost calculation and cost update is preferably performed in the neighborhood of the cursor by overlapping tiles. Since OpenGL[2] can only handle images of a limited texture size, each image had to be subdivided into manageable texture sizes, whereas the image is loaded completely into the RAM for calculation and a tile class executes the calculation loops in the respective RAM limited by tile size.

Control points were introduced to attach correct presegmented parts of the shape. When the current path segment reaches a predefined length, control points with a specific path distance as well as the preceding path segment are set. These minor changes make real time interaction with the image data possible. This is a fundamental prerequisite to conducting usability studies based on the algorithms presented here.

The following is a basic technical description of the automatic zoom's kernel.

```
Pseudocode of the zoom function:

//initialize sum of the speeds with 0
sumSpeed = 0;
//initialize number of sampled speeds with 0
sumCount = 0;
//initialize change of zoom with 0
deltaZoom = 0;

//test if the mouse was moved
if dist(p, q) > 0 do
   //increase sumSpeed by speed
   //of last moved distance
   sumSpeed += dist(p,q) / passedTime;
   sumCount++;
end

//evaluate average speed every 500 msec
if 05-Sec-Timer do
   //calculate avg speed over last 500 msec
   avrSpeed = sumSpeed / sumCount;
   //reset sum of speed
   sumSpeed = 0;
   //reset count of sampled speeds
   sumCount = 0;
   //check if avg speeds exceeds maximum value
   if avrgSpeed > maxSpeed do
      //decrease deltaZoom
      deltaZoom -= 5;
   end

   //check if avg speeds is smaller
   //than minimum value
   if avrgSpeed < maxSpeed do
      //increase deltaZoom
      deltaZoom += 5;
   end
   //restart the 3 seconds timer
   03-Sec-Timer.restart();
end

if 01-Sec-Timer do
   //check if deltaZoom exceeds
   if  |deltaZoom| > maxDelta do
      //if deltaZoom is smaller than 0 zoom out
      if deltaZoom < 0 do
         decreaseZoom by 1;
      //if deltaZoom is bigger than 0 zoom in
```

[1] www.wxwidgets.com

[2] Hardware dependent.

```
        else
            increaseZoom by 1;
        end
    end
    //restart the 1 seconds timer
    01-Sec-Timer.restart();
end
```

In the brief description of pseudo haptic feedback below, the new C/D ratio ensues from the related cost map and the current mouse movement.

```
Pseudocode for pseudo haptic feedback:


// 2-D vector from last mouse position
// to current mouse position
deltaMouse = currMousePos - lastMousePos;

// calculate normalized cost from last
// mouse position to current mouse position
currCost = CalcCost(lastMousePos, currMousePos);

// calculate length of mouse change
deltaV = Length(deltaMouse);

// calculate new C/D ratio
ratio = CDRmax * sqrt(deltaV*deltaV +
        currCost*currCost);

// modify new mouse position via ratio
currMousePos = lastMousePos + deltaMouse * ratio;
```

5 USABILITY STUDY

A preliminary usability study investigated the strengths and limitations of automatic zooming and pseudo haptic feedback for semiautomatic segmentation.

5.1 Setup and Object of Investigation

The participants of the study were divided into four groups based on their abilities and ages[3] (see Table 1). Three different segmentation methods (manual segmentation, conventional live wire and improved live wire with automatic zoom and pseudo haptics) were tested. The order of segmentation methods was altered for each participant so that the learning rate influenced each segmentation time.

Number	Age	Skills
1	Young	Good
2	Young	Moderate
3	Medium	Good
4	Medium	Moderate

Table 1: Study group

The first aspect demonstrated was the completion time of a predefined segmentation task. The pool of participants consisted of sixteen untrained individuals. Hence, the training scenario consisted of a specially generated dataset. The original data was enriched with a superimposed ideal segmentation shape of a pericarp, the tissue surrounding a seed that develops from the ovary wall of a flower. An initial manual segmentation performed by four experts defined the gold standard segmentation.

Augmentation without zooming is represented as a thin dashed line. The thickness of the line is proportional to the magnitude of the edge. When the zoom was used in critical segmentation regions, several lines were inserted to compel the user to apply the global context. Figure 6 shows two different zoom levels of the dataset.

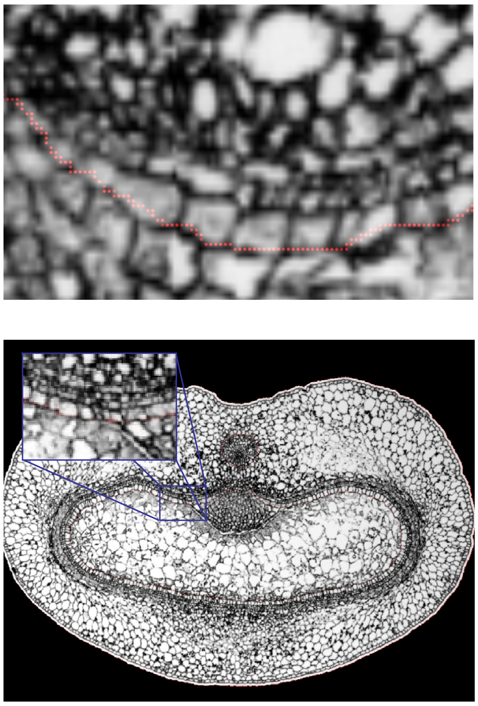

Figure 6: Top: Detail of dataset with zoom
Bottom: Detail without zoom

The second aspect is the accuracy of the segmentation results.

Errors were quantified as the ratio of misclassified pixels to the total number of pixels (error rate). Standard deviation or the mean from the gold standard are a more informative comparative characteristic of accuracy. The point with the closest Hausdorf distance was selected to calculate this.

5.2 Observations and Comments

In their responses to an informal questionnaire, most participants indicated that the conventional pan and zoom mechanism buttons/scrollbars/mouse wheel are easy to use but require too many discrete manipulation steps to navigate between two separate points with different zoom levels. Thus, users sometimes lose their focus on the task at hand. In other words, navigation

[3] Under twenty-nine years of age = young.

requires too much attention from users. Furthermore, users complained, that the discrete steps in the visualization of different zoom levels were laborious. Hence, many users avoided manual zoom changes. All participants experienced the automatic zoom as unfamiliar in the beginning but were able to adapt quite quickly. Users were able to navigate with a tighter focus and concentrate on the segmentation task. Pure manual segmentation of the dataset was a Sisyphean task in comparison with the two semiautomatic segmentation methods with speed-dependent automatic zooming and pseudo haptic feedback or conventional interface control. Virtually all the participants rated the pseudo haptic feedback as an excellent solution enabling direct manipulation with small control amplitudes and the elimination of inaccuracies.

5.3 Results

The formal user study delivered virtually all the preliminary results. Both methods using semiautomatic segmentation facilitate significantly faster completion times than manual selection. Speed-dependent automatic zoom combined with pseudo haptic feedback is generally not faster than the conventional method. This slightly contradicts the subjective impressions of the majority of the participants (see Section 5.2) and may be attributable to the dynamics of automatic zooming and panning. Figure 7 presents the results of the completion time test for each segmentation method. The standard deviation of the completion time is visualized with continuous T-lines for each group.

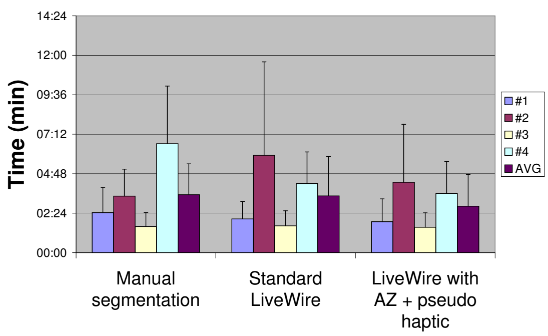

Figure 7: Completion times of manual segmentation, standard live wire and live wire with automatic zoom (AZ) and pseudo haptics

Groups 2 and 4 experienced consistently longer segmentation times with strong deviations (nearly equaling the average segmentation times). Groups 1 and 3 had far better segmentation times than groups 2 and 4, which consisted of test persons with average computer skills. A significant and unexpected finding was that experienced users' segmentation times hardly varied (only 6 %) regardless of whether they were using the improved live wire with automatic zoom (AZ) and pseudo haptics or the conventional live wire. On the other hand, members of groups 2 and 4 were on average 28 % and 14 % faster with the automatic zoom and pseudo haptics than with the standard live wire method. This may have been because the automatic navigation provides inexperienced users better support, whereas experienced users are able to navigate and simultaneously concentrate on their current tasks (just as in game playing for example). Disregarding extreme individual deviations, the ages of users with the same level of knowledge appear to play no role in the average segmentation time. The average segmentation times of all users are presented in Figure 7 (violet bars).

Figure 8 (bottom) lists the error rates of all participants, i.e. the ratio between misclassified pixels to the total number of pixels. Figure 8 (top) presents the average pixel deviation (mean) and the standard deviation from the mean in relation to the gold standard segmentation. The value of the standard deviation from the mean indicates local misclassifications. Since accuracy did not deviate strongly among the user groups, only the averaged results per segmentation method are presented here.

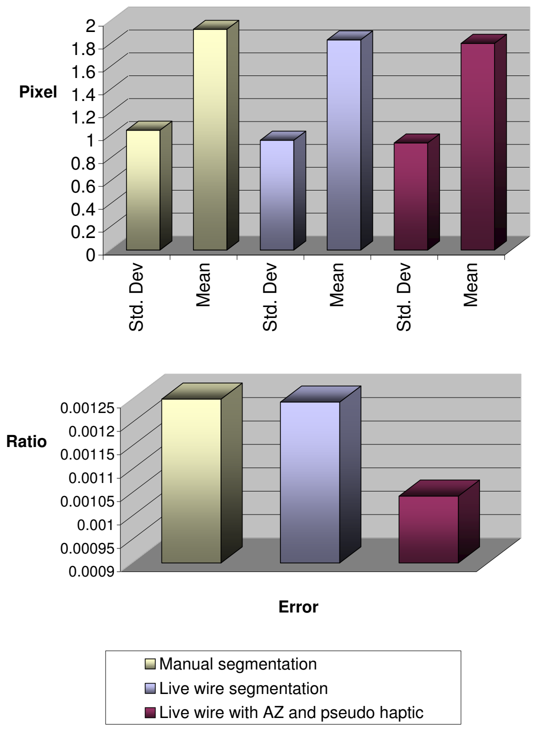

Figure 8: Test subject' accuracy with the segmentation methods tested (top) mean and standard deviation (bottom) error rate

Figure 9 visualizes a typical segmentation process. The transitions in the automatic zoom curve are smoother than in the manual zoom curve. Furthermore, the completion time is significantly shorter.

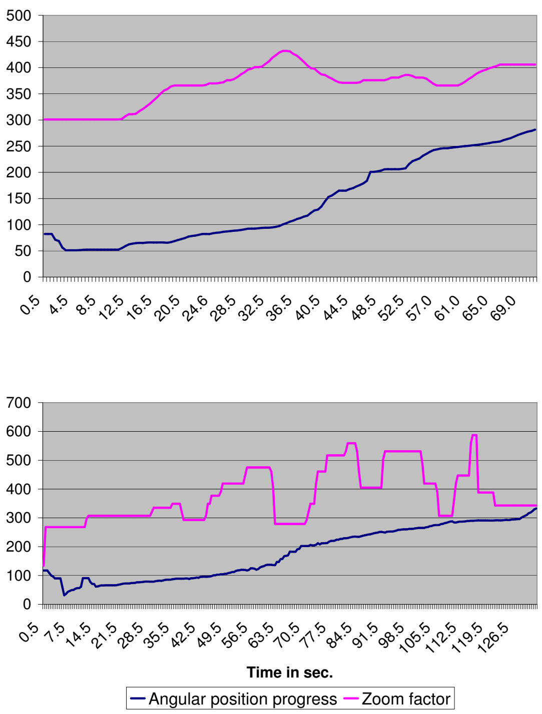

Figure 9: Comparison of one representative participant's unwound segmented shape with automatic zoom (top) and with manual zoom adjustment (bottom)

6 FUTURE DIRECTIONS

One challenge will be to design plausible transitions between different zoom levels that allow users to change speeds quickly without causing any confusion. At fast speeds, different users sometimes expect different system feedback from the same user action. Further research is therefore required, especially on adjusting the zooming parameters to individual users. Graphically manipulating the zoom function is planned, thus making it possible to save results in a specific user file.

Development of a board-like input device for two hands is also planned: Such a device would increase the speed and accuracy of segmentation as Kron and Schmidt [KS05] have demonstrated. Another approach deserving further investigation is inverse panning: The user remains stationary and the image data moves through the screen space, similar to cutting wood with a jigsaw.

Flicking movements represent another technique [MC03], [DAZ99] and can be employed to quickly and precisely access a single control from a small group of plausible candidates. Whether the integration of additional navigation controls such as a default zoom level or a last contour segment reset can be suitably mapped to distinct mouse movements deserves more investigation.

While the informal user study demonstrated the capabilities expected for a biological test dataset, the results have to be verified for different segmentation scenarios especially in the context of practical use. Furthermore, the refined prototype algorithms must be studied additionally to determine whether the haptic or the automatic zoom or a combination of the two improve the standard live wire technique most.

7 CONCLUSION

This paper introduces a new technique that improves semiautomatic segmentation. The automatic zoom is based on a user's interaction speed and the underlying image data. Slow user input produces an in-depth view of an image, thus making it easier to segment unclear or overly detailed regions. Fast user input indicates strong edges that can typically be segmented easily with live wire. This can be done on a lower zoom level on which the user can still recognize the edges and maintain an overview. Pseudo haptic feedback additionally supports accurate segmentation in complex areas. Movement toward the edge direction is penalized with a larger C/D ratio, thus increasing control device amplitude.

In an informal usability study, most users preferred this technique over segmentation with manual zoom. Moreover, they described their impression of pseudo haptics as a "magical force" that binds the cursor to the correct position in an image or leads it there.

In terms of objective results, the approach presented here increased accuracy. Misclassifications (error rate) were reduced by 16 % and the mean and standard deviation of the Hausdorf distances from the gold standard of manual segmentation was 10 % and 10 % lower. The new approach additionally cut all users' segmentation times by nearly 20 %. It increased accuracy with 16 % fewer misclassifications than the standard live wire method. However, the mean and standard deviation from the target contour were only reduced 1 % and 2 %, respectively. Average segmentation speed was reduced approximately 18 %.

8 ACKNOWLEDGEMENT

This research work is being supported by the BMBF grants 0313821B and 0313821A.

9 REFERENCES

[AB94] R. Adams and L. Bischof. Seeded region growing. *IEEE Trans. Pattern Anal. Mach. Intell.*, 16(6):641–647, 1994.

[AS07] S. Avidan and A. Shamir. Seam carving for content-aware image resizing. In *SIGGRAPH '07: ACM SIGGRAPH 2007 papers*, page 10, New York, NY, USA, 2007. ACM Press.

[BBS+06] C. Brüß, F. Bollenbeck, F.-M. Schleif, W. Weschke, T. Villmann, and U. Seiffert. Fuzzy image segmentation with fuzzy labelled neural gas. In *Proc. of the 14th European Symposium on Artificial Neural Networks ESANN*, 2006.

[BS86] E. A. Bier and M. C. Stone. Snap-dragging. In *SIGGRAPH '86: Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, pages 233–240, New York, NY, USA, 1986. ACM Press.

[DAZ99] M. S. Dulberg, R. S. Amant, and L. Zettlemoyer. An imprecise mouse gesture for the fast activation of controls. In *In Proceedings of INTERACT' 99*, pages 375–382, 1999.

[DLB+05] L. Dominjon, A. Lecuyer, J. Burkhard, P. Richard, and S. Richir. Influence of control/display ratio on the perception of mass of manipulated objects in virtual environments. In *Proceedings of IEEE Virtual Reality*, pages 19–25. IEEE Computer Society Press, 2005.

[Far88] G. Farin. *Curves and surfaces for computer aided geometric design: a practical guide*. Academic Press Professional, Inc., San Diego, CA, USA, 1988.

[Fur86] G. W. Furnas. Generalized fisheye views. In *CHI '86: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 16–23, New York, NY, USA, 1986. ACM Press.

[GDB+07] S. Gubatz, V.J. Dercksen, C. Brüss, W. Weschke, and U. Wobus. Analysis of barley (Hordeum vulgare) grain development using three-dimensional digital models. *The Plant Journal*, 6:preprint, 2007.

[GDGC02] O. Gérard, T. Deschamps, M. Greff, and L. D. Cohen. Real-time interactive path extraction with on-the-fly adaptation of the external forces. In *ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part III*, pages 807–821, London, UK, 2002. Springer-Verlag.

[Hah05] H. K. Hahn. *Morphological Volumetry*. PhD thesis, Universität Bremen, 2005.

[HS85] R. M. Haralick and L. G. Shapiro. Image segmentation techniques. *Computer Vision, Graphics, and Image Processing*, 29(1):100–132, 1985.

[IH00] T. Igarashi and K. Hinckley. Speed-dependent automatic zooming for browsing large documents. In *UIST '00: Proceedings of the 13th annual ACM symposium on User interface software and technology*, pages 139–148, New York, NY, USA, 2000. ACM Press.

[JF98] S. Jul and G. W. Furnas. Critical zones in desert fog: aids to multiscale navigation. In *UIST '98: Proceedings of the 11th annual ACM symposium on User interface software and technology*, pages 97–106, New York, NY, USA, 1998. ACM Press.

[Kan05] H. W. Kang. G-wire: A livewire segmentation algorithm based on a generalized graph formulation. *Pattern Recognition Letters*, 26(13):2042–2051, 2005.

[Kon07] Konrad-Zuse-Zentrum für Informationstechnik Berlin (ZIB). Amira - Advanced 3D Visualization and Volume Modeling, 2007. www.amira.zib.de last visit: Sep. 6th 2007.

[Kre05] I. P. Krepps. Design and evaluation of a multi-floor speed dependent automatic zoom interface. Master's thesis, Royal Military College of Canada, April 2005.

[KS05] A. Kron and G. Schmidt. Haptic telepresent control technology applied to disposal of explosive ordnances: Principles and experimental results. In *Proc. of ISIE'05, IEEE International Symposium on Industrial Electronics*, volume 4, pages 1505– 1510, 2005.

[KWT88] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, V1(4):321–331, January 1988.

[LBE04] A. Lécuyer, J.-M. Burkhardt, and L. Etienne. Feeling bumps and holes without a haptic interface: the perception of pseudo-haptic textures. In *CHI '04: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 239–246, New York, NY, USA, 2004. ACM Press.

[LM01] L. Lucchese and S. K. Mitra. Color image segmentation: a state-of-the-art survey. In *Proceedings of the Indian National Science Academy*, pages 207–221, 2001.

[MB98] E. N. Mortensen and W. A. Barrett. Interactive segmentation with intelligent scissors. *Graphical Models and Image Processing*, 60(5):349–384, September 1998.

[MB99] E. N. Mortensen and W. A. Barrett. Toboggan-based intelligent scissors with a four-parameter edge model. *cvpr*, 02:2452, 1999.

[MC03] M. Moyle and A. Cockburn. The design and evaluation of a flick gesture for 'back' and 'forward' in web browsers. In *AUIC '03: Proceedings of the Fourth Australasian user interface conference on User interfaces 2003*, pages 39–46, Darlinghurst, Australia, Australia, 2003. Australian Computer Society, Inc.

[Shn96] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *VL '96: Proceedings of the 1996 IEEE Symposium on Visual Languages*, page 336, Washington, DC, USA, 1996. IEEE Computer Society.

[SPP00] A. Schenk, G. P. M. Prause, and H.-O. Peitgen. Efficient semiautomatic segmentation of 3d objects in medical images. In *MICCAI '00: Proceedings of the Third International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 186–195, London, UK, 2000. Springer-Verlag.

[vWN03] J. J. van Wijk and W. A. A. Nuij. Smooth and efficient zooming and panning. *INFOVIS*, 2003.

[vWN04] J. J. van Wijk and W. A. A. Nuij. A model for smooth viewing and navigation of large 2d information spaces. *IEEE Trans. Vis. Comput. Graph.*, 10(4):447–458, 2004.

[WGCO07] L. Wolf, M. Guttmann, and D. Cohen-Or. Non-homogeneous content-driven video-retargeting. In *Proceedings of the Eleventh IEEE International Conference on Computer Vision (ICCV-07)*, 2007.

[ZSS97] S. Zhai, B. A. Smith, and T. Selker. Improving browsing performance: A study of four input devices for scrolling and pointing tasks. In *INTERACT '97: Proceedings of the IFIP TC13 Interantional Conference on Human-Computer Interaction*, pages 286–293, London, UK, UK, 1997. Chapman & Hall, Ltd.