Rendering Optimizations Guided by Head-Pose Estimates and their Uncertainty

Javier E. Martínez[1], Ali Erol[1], George Bebis[1], Richard Boyle[2], Xander Twombly[2]

[1]Computer Vision Laboratory, University of Nevada, Reno, NV 89557
[2]BioVis Laboratory, NASA Ames Research Center, Moffett Field, CA 94035
{javier, bebis, aerol}@cse.unr.edu, {rboyle, xtwombly}@mail.arc.nasa.gov

Abstract. In virtual environments, head pose and/or eye-gaze estimation can be employed to improve the visual experience of the user by enabling adaptive level of detail during rendering. In this study, we present a real-time system for rendering complex scenes in an immersive virtual environment based on head pose estimation and perceptual level of detail. In our system, the position and orientation of the head are estimated using stereo vision approach and markers placed on a pair of glasses used to view images projected on a stereo display device. The main innovation of our work is the incorporation of uncertainty estimates to improve the visual experience perceived by the user. The estimated pose and its uncertainty are used to determine the desired level of detail for different parts of the scene based on criteria originating from physiological and psychological aspects of human vision. Subject tests have been performed to evaluate our approach.

1 Introduction

Virtual environments (VEs) are effective computing technologies that allow deployment of various advanced applications including immersive training systems, surgical simulations, and visualization of large data sets among others. Development of such computing environments raises challenging research problems. To allow high degree-of-freedom (DOF) natural interaction, new input modalities based on direct sensing of the hand, eye-gaze, head and even the whole human body motion are being incorporated. To create an immersion effect, advanced display technologies such as 3D stereo displays or CAVE environments are being engineered and high quality real-time rendering algorithms are being developed.

Among different input modalities, head pose and/or eye-gaze estimation provide an effective input mainly for navigation tasks in VEs. During navigation, head pose information (i.e., 6 DOF) can help to optimize the computational load of rendering and increase visual quality at regions where the user is focusing on by estimating where the user is looking at. Technically, it is possible to employ adaptive level of detail (LOD) in rendering to improve the visual experience perceived by the user without a major increase in the computational load.

In this study, we present a real-time system for rendering complex scenes in an immersive virtual environment based on head pose estimation and perceptual level of detail (PLOD) [1]. In our system, the position and orientation of the head are estimated using stereo vision and markers placed on a pair of glasses that the user has to wear to view images projected on a stereo display device. The main innovation of our work is the incorporation of uncertainty estimates to improve the visual experience perceived by the user. The estimated pose and its uncertainty are used to determine the desired LOD for different parts of the scene based on criteria originating from physiological and psychological aspects of human vision. This work is part of a larger collaborative effort between our group and *BioVis* lab at NASA Ames to build a virtual simulator (i.e., Virtual Glove Box or VGX). VGX is intended to provide an advanced “fine-motor coordination” training and simulation system for astronauts to perform precise biological experiments in a Glovebox aboard the International Space Station [21][22].

In the next section, we present a brief review of previous work on PLOD. In Section 3, we describe of our system. The implementation details of head-pose estimation and PLOD calculation are presented in Sections 4 and 5 respectively. In Section 6, we report and discuss the results of our experiments. Finally, Section 7 contains our conclusions.

2 Previous Work

While the first work on PLOD dates back to '76 [1], most of the development has been done during the last decade. These advancements can be grouped into three areas, namely *criteria*, *mechanism* and *error measure*. The *criteria* are a set of functions that select areas from the objects that need to be drawn with a certain LOD. The *mechanism* is another set of functions that modify the geometry to achieve the desired LOD. They correspond to polygon simplification mechanisms that fall under four categories [5]: sampling, adaptive subdivision, vertex decimation and vertex merging. The *error measure* is an evaluation of the differences between the original object and the modified one, and it is used to control the mechanism. Measuring deviations from the original mesh to the modified mesh allows the quantification of the errors introduced when modifying the mesh. Common error measures in the literature include vertex-vertex, vertex-plane, vertex-surface, and surface-surface distances. Ideally, we would like these errors to be imperceptible to the user.

The most important part of a PLOD system is the set of criteria used to modulate the LOD. These criteria are related to or based on physiological and psychological aspects of human vision [2, 3]. We list below several important criteria [4]:

- **Contrast sensitivity:** The LOD is modulated depending on whether it is inside or outside of the Contrast Sensitivity Function (CSF) curve that shows the relationship between contrast and spatial frequency in human visual perception [4] .
- **Velocity:** The LOD is modulated proportionally to the relative velocity of the eye across the visual field.

- **Eccentricity:** The LOD is modulated proportionally to the angular distance of the object to the viewpoint.

- **Depth of field:** The LOD is modulated proportionally to the distance to the Panum's fusional area [2]. This is used only in connection with stereo-vision.

There are several examples of systems that make use of eye-gaze for guiding perceptually motivated simplifications including Reddy [10], Luebke [3], Williams [18] and Murphy [17]. Both [3] and [17] make use of an eye tracker to estimate the eye-gaze vector. In [3], the user’s head is placed in a chin rest to avoid having to calculate the position of the eyes. Only [17] tracks the head and the eyes simultaneously allowing the user to move in a more natural way.

3 System Design

Immersive VEs can be implemented in various operational environments, mainly determined by the output devices. In this study, we targeted a stereo display system [21]. An ideal system would require tracking both the head and eye-gaze simultaneously to allow arbitrary motion of the user; however, eye-gaze tracking could be very costly and intrusive. In our application, users need to wear a pair of polarized glasses which makes eye-gaze tracking challenging since the user's eyes are not visible. To keep things simple, we decided to obtain a rough estimate of eye-gaze by tracking the head and estimating its orientation. Developing a more accurate eye-gaze tracking system (e.g., by mounting small cameras on the frame of the glasses) is a part of our future work.

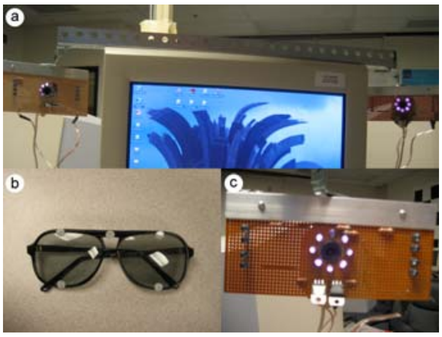

Fig. 1. Hardware setup: (a) camera setup on the computer, (b) eye-glasses with IR reflective markers, (c) camera close-up with IR LEDs.

To make head tracking fast and robust, we took advantage of the requirement that the users have to wear glasses by placing several markers on the frame of the glasses. This approach simplifies detecting the head without being intrusive. A challenging issue in designing the system was how to deal with illumination since any kind of external illumination could interfere with the stereo display device (e.g. projectorbased) and disturb the user. To deal with this issue, we decided to use IR LEDs for illumination and IR reflective markers as shown in Figure 1. A high-pass filter was installed on the cameras to block visible light from entering the camera sensor. The filter used in our setup was the Kodak Wratten 97c filter which has a cut-off limit of 800nm.

The system contains three modules as shown in Figure 2: (a) a vision module, (b) a PLOD module and (c) a rendering module. The vision module detects the position, orientation and uncertainty of the user's head and passes it to the PLOD module, which takes into account the physiological and psychological aspects of the human vision to calculate the LOD at which to draw the elements. Finally, the rendering module draws everything on the screen at the calculated LOD.

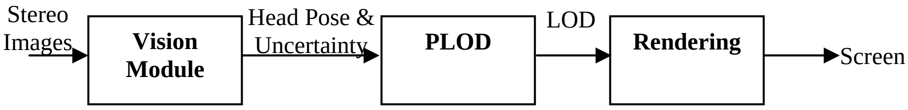

Fig. 2. Block diagram of the whole system.

4 Head Pose Estimation

The vision module includes three processing steps: (a) marker extraction, (b) pose estimation, and (c) uncertainty estimation. First, the markers are extracted in each image. Then, the head pose is estimated by reconstructing the location of the markers in 3D through triangulation. Finally, uncertainty associated with the estimated pose is calculated.

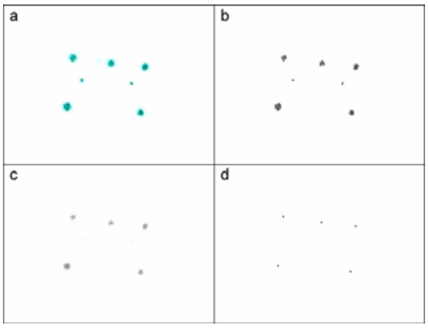

Fig. 3. Negative of the images from each stage of the process starting with: a) input image; b) thresholded with a value of 100; c) smoothed with a 9-pixel Gaussian filter; and d) final image showing the centers

4.1 Marker Extraction

The combination of IR illumination and IR reflective markers allows for fast and robust feature extraction. In the input images (see Figure 3(a)) the background is already suppressed due to the use of the filter that blocks visible light, allowing the detection and extraction of markers through a simple thresholding operation as shown in Figure 3(b). The thresholded image is then processed using a Gaussian filter to eliminate noise (see Figure 3(c)).

For each marker on the image, we estimate its center with sub-pixel accuracy (see Figure 3(d)). It should be noted that, it is still possible to get some extra blobs during segmentation due to light reflections on the eye-glasses; however, the special arrangement of markers (see Figure 4) can help us to eliminate them (e.g., by requiring that the upper 3 markers lie roughly on a line). This special marker configuration also allowed us to identify uniquely each marker on a single image (e.g. establish correspondences between the left and right images).

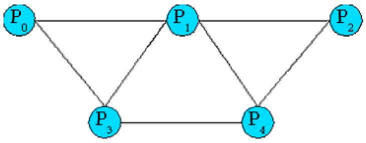

Fig. 4. Marker arrangement on the glasses.

4.2 Pose Estimation

Once the markers have been extracted in each image, the center of each marker can be used to calculate its 3D location using triangulation. In our approach, the location of the head is estimated by the location of the middle marker P_1 while its orientation is estimated by averaging the normal vectors corresponding to the three triangles shown Figure 4. We have validated the accuracy of our head pose estimate algorithm using a magnetic tracker with an accuracy of 1.8mm in the position and 0.5° in the orientation.

4.2 Uncertainty Estimation

It is possible to associate an uncertainty measure to both the position and orientation estimates of the head; however, we have observed that the uncertainty in orientation has a much higher effect on the LOD mainly due to the amplification of the error in the calculation of the point of interest on the screen (see Section 5). Therefore, we are only considered estimating orientation uncertainty.

Uncertainty calculation in stereo vision is a well studied topic. In general, it is possible to propagate calibration and feature localization errors to the estimates of 3D position and local orientations [15]. However, estimating orientation uncertainty analytically in our system was rather difficult; therefore, we implemented a random sampling approach.

Specifically, in matching two markers, we assume that the correspondences between pixels belonging to each marker are unknown. Using the epipolar constraint and the distance of the pixels from the center of the marker, we generate a cloud of 3D points for each marker. Then, each cloud is randomly sampled and all possible combinations of the samples are used to generate orientation estimates by computing the covariance matrix of the samples.

5 Perceptual Level of Detail

We assumed that the scene is represented by a triangular mesh corresponding to the coarsest LOD. For each triangle in the mesh, we calculate the desired LOD and increase the resolution (i.e. generate smaller triangles) accordingly, using adaptive subdivision.

In the calculation of the desired LOD, we are interested in finding the highest spatial frequency that a person can see at a particular location under conditions determined by the triangle's contrast, eccentricity and angular velocity. The spatial frequency of a triangle is found by measuring the maximum angle between the vertices of the triangle projected on the screen plane with respect to the head position. The contrast level for a triangle is obtained by rendering the triangle at the coarsest level and examining the color content of the projection. To calculate the eccentricity, the triangle is represented by its geometric center. Uncertainty estimates mainly affect the eccentricity values.

When the user is looking at the screen, the direction of his (her) head intersects the plane formed by the screen at a point called the Point of Interest (POI). Uncertainty in head orientation affects the location of the POI, which in turn affects the eccentricity values of the triangles. We have incorporated orientation uncertainty in the eccentricity calculations by modifying the triangle’s location with respect to the POI. Specifically, given a point P and the orientation uncertainty matrix Σ, an uncertainty corrected point P_u is calculated as follows:

$$P_u = e^{-\frac{1}{2}(P-POI)^T \Sigma^{-1}(P-POI)}\left(POI - P\right) + P \tag{1}$$

where point P is shifted towards POI proportionally to the probability that P itself is the POI.

The highest spatial frequency is determined by solving a system of equations given the contrast, angular velocity and the modified eccentricity values of triangles. Once the highest spatial frequency is known, it can be related to a certain LOD that is determined by the implementation. In our system, the depth of subdivision is taken to be the LOD measure.

6 Subject Tests

The objective of our tests was to quantify the improvement obtained by incorporating uncertainty correction on a perceptually oriented display system. The test application displays a terrain section or height map (see Figure 5), on which PLOD optimizations are applied. The user is shown three test cases (i.e. different views of the terrain), each containing three scenarios.

The first scenario presents the user with common optimizations found in the literature; namely velocity, contrast and eccentricity. The second scenario uses a constant uncertainty correction to modify the way eccentricity behaves. The constant uncertainty matrix is chosen to contain the maximum uncertainty values obtained using our algorithm on a large number of experiments. The third scenario uses uncertainty corrections like before; however, the covariance matrix is continuously updated through the sampling algorithm presented in Section 4.2.

In each case, the user was asked to judge the amount of changes perceived all over the screen while browsing the map by moving his/her head. The judgment of the user is constrained to be high, medium or low/no changes. This judgment is obviously very subjective but it helps establishing a baseline for comparing the results of different types of tests. We are only interested in the relative change rather than the absolute values of the responses.

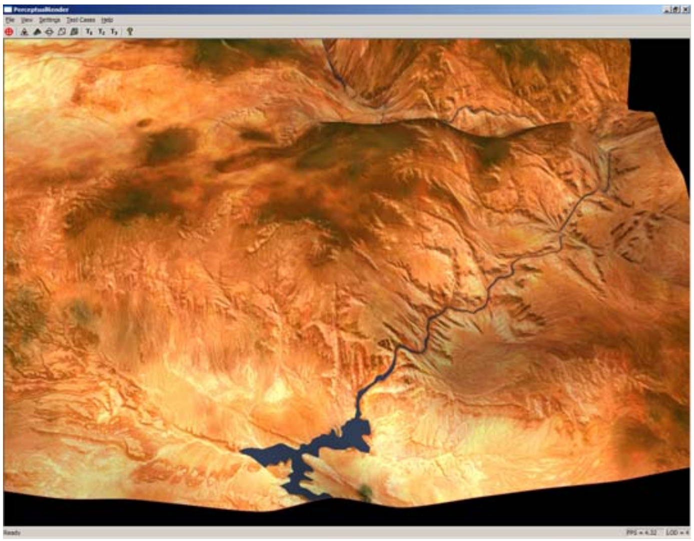

Fig. 5. Terrain view for test case 2.

Our experiments were performed using 19 test subjects. Comparisons between different scenarios were performed, tabulating the increase and decrease rate of one test scenario versus the other. Our results are shown in Tables 1-4. In all tables, changes in user's satisfaction across the three scenarios are listed in the first column. In all cases, a change in user's satisfaction could be an increase, no change or a decrease. Table 1 shows the average over all cases, while Tables 2, 3 and 4 show the results for test cases 1, 2 and 3 respectively.

Table 1. Satisfaction comparison between test scenarios across all test cases

All cases	Increase	No Change	Decrease	Total
Fixed vs. None	63.16%	29.82%	7.02%	100.00%
Variable vs. None	26.32%	54.39%	19.30%	100.00%
Variable vs. Fixed	8.77%	33.33%	57.89%	100.00%

Table 2. Satisfaction comparison between test scenarios for test case 1

Case 1	Increase	No Change	Decrease	Total
Fixed vs. None	52.63%	47.37%	0.00%	100.00%
Variable vs. None	21.05%	52.63%	26.32%	100.00%
Variable vs. Fixed	10.53%	21.05%	68.48%	100.00%

Table 3. Satisfaction comparison between test scenarios for test case 2

Case 2	Increase	No Change	Decrease	Total
Fixed vs. None	63.16%	31.58%	5.26%	100.00%
Variable vs. None	21.05%	63.16%	15.79%	100.00%
Variable vs. Fixed	0.00%	42.11%	57.89%	100.00%

Table 4. Satisfaction comparison between test scenarios for test case 3

Case 3	Increase	No Change	Decrease	Total
Fixed vs. None	73.68%	10.53%	15.79%	100.00%
Variable vs. None	36.84%	47.37%	15.79%	100.00%
Variable vs. Fixed	15.79%	36.84%	47.37%	100.00%

From Table 1, we can see that the use of fixed uncertainty greatly improves performance. In the case of fixed uncertainty, only 7% of the time people perceived worse performance compared to not having uncertainty optimizations enabled. The results for dynamic uncertainty are not as good as those for fixed uncertainty. About 55% of the time people did not notice any differences between using variable uncertainty and not using it. The direct comparison between dynamic and static uncertainty shows that dynamic uncertainty performance is clearly perceived as worse 57% of the time. Similar results can be observed for all test cases as shown in Tables 2-4.

Further analysis of our system's performance revealed that the main reason for the underperformance of the variable uncertainty approach was the jitter in the uncertainty covariance matrix. In particular, the calculation of the covariance matrix was not very stable and its values oscillated. These oscillations made the triangles that lie on the outer edges of the high resolution region to change levels back and forth from one level to the next. Since the human eye has an increased sensitivity to movements on the periphery compared to the center, this effect made the users more aware of changes in the periphery. The main reason for the oscillations was probably our sampling strategy. For the sake of high processing speed, we assumed a uniform distribution over the cloud of points which might not be a valid assumption. Several techniques that can be used to solve this problem including Monte Carlo, Shifted Hammersley, Latin Hypersquare, Equal Probability Sampling and others.

Another important observation was the increase in rendering speed when using the PLOD compared to rendering the same terrain at the highest LOD. The frame rate increased from 5 fps to 15 fps on a Pentium® 4 2.56MHz processor with 1 GB of RAM.

7 Conclusions

We have presented a real-time system that combines a vision module that estimates the user's head pose with a PLOD module that optimizes image rendering based on perceptual parameters. The system was implemented on a fairly modest PC using off the shelf components and it was able to improve the frame rate significantly compared to rendering the same terrain at full resolution. Subject tests were performed to assess the benefits of using uncertainty estimates in conjunction with other parameters. Our results indicated that uncertainty estimates help in making optimizations more seamless to the user. An approach for calculating orientation uncertainty was presented and employed as part of the vision module. However, the jitter in the uncertainty calculations prevented us from achieving the same level of performance compared to using fixed parameters. More details about this work can be found in [23]. Future work includes further investigation of these issues as well as estimating eye-gaze more accurately.

Acknowledgment

This work was supported by NASA under grant #NCC5-583

References

1. J. H. Clark. "Hierarchical Geometric Models for Visible Surface Algorithms". Communications of the ACM, vol. 17(2), pages 547-554, 1976.
2. T. Oshima, H. Yamamoto, and H. Tamura. "Gaze-Directed Adaptive Rendering for Interacting with Virtual Space". Proceedings of 1996 IEEE Virtual Reality Annual International Symposium, pages 103-110, 1996.
3. David Luebke et al. "Perceptually Driven Simplification Using Gaze-Directed Rendering". University of Virginia Technical Report CS-2000-04.
4. David Luebke et al. Level of Detail for 3D Graphics. Morgan Kaufmann Publishers, 1st ed., 2003.
5. David Luebke. "A Developer's Survey of Polygonal Simplification Algorithms". IEEE Computer Graphics &Applications, May 2001.

6. Stephen Junkins and Allen Hux." Subdividing Reality". Intel Architecture Labs White Paper, 2000.
7. J. Warren and S. Schaefer. "A factored approach to subdivision surfaces". Computer Graphics and Applications, IEEE, pages 74-81, 2004.
8. David H. Eberly. 3D Game Engine Design. Morgan Kaufmann Publishers, 2001.
9. J. L. Mannos and D. J. Sakrison. "The Effects of a Visual Fidelity Criterion on the Encoding of Images". IEEE Transactions on Information Theory, vol. 20(4), pages 525-535, 1974.
10. M. Reddy. "Perceptually Modulated Level of Detail for Virtual Environments". PhD. Thesis. CST-134-97. University of Edinburgh, Edinburgh, Scotland, 1997.
11. Andrew T. Duchowski. "Acuity-Matching Resolution Degradation Through Wavelet Coefficient Scaling". IEEE Transactions in Image Processing, vol. 9(8), pages 1437-1440, August 2000.
12. Wilson S. Geisler and Jeffrey S. Perry, "Real-time Simulation of Arbitrary Visual Fields". ACM Symposium on Eye Tracking Research & Applications, 2002.
13. Jeffrey S. Perry and Wilson S. Geisler. "Gaze-contingent real-time simulation of arbitrary visual fields". Proceedings of SPIE: Human Vision and Electronic Imaging, San Jose, CA, 2002.
14. Qiang Ji and Robert M. Haralick. "Error Propagation for Computer Vision Performance Characterization". International Conference on Imaging Science, Systems, and Technology, Las Vegas, June, 1999.
15. Don Murray and James J. Little. "Patchlets: Representing Stereo Vision Data with Surface Elements". Workshop on the Applications of Computer Vision (WACV), 2005.
16. David A. Forsyth and Jean Ponce. Computer Vision A Modern Approach. Prentice Hall, 1st ed., 2003.
17. Hunter Murphy and Andrew T. Duchowski. "Gaze-Contingent Level Of Detail Rendering". EuroGraphics Conference, September 2001.
18. Nathaniel Williams et al. "Perceptually guided simplification of lit, textured meshes". Proceedings of the 2003 Symposium on Interactive 3D graphics, Monterrey, CA, 2003.
19. K. Arun et al. "Least-squares fitting of two 3-D point sets", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 9(5), pages 698-700, 1987.
20. Robyn Owens. Lecture notes, http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/OWENS/LECT11/node5.html
21. "VirtualgloveBox", http://biovis.arc.nasa.gov/vislab/vgx.htm
22. "Effective Human-Computer Interaction in Virtual Environments", http://www.cse.unr.edu/CVL/current_proj.php
23. Javier Martinez, "Rendering Optimizations Guided by Head-Pose Estimates and Their Uncertainty", M.S. Thesis, Dept of Computer Science and Engineering, University of Nevada, Reno, August 2005.