Real-Time 3D-Teleimmersion

K. Daniilidis[1], J. Mulligan[1], R. McKendall[1],
G. Kamberova[3], D. Schmid[1], R. Bajcsy[1,2]
[1] University of Pennsylvania
[2] NSF CISE Directorate,
[3] Washington University

Abstract

In this paper we present the first implementation of a new medium for telecollaboration. The realized testbed consists of two tele-cubicles hooked at two Internet nodes. At each telecubicle a stereo-rig is used to provide an accurate dense 3D-reconstruction of a person in action. The two real dynamic worlds are transmitted over the network and visualized stereoscopically. The full-3D information facilitates the interaction with any virtual object demonstrating in an optimal way the confluence of graphics, vision, and communication.

In particular, the remote communication and the dynamic nature of tele-collaboration put the challenge of optimal representation for graphics and vision. We treat the issues of limited bandwidth, latency, and processing power with a tunable 3D-representation where the user can decide over the trade-off between delay and 3D-resolution by tuning the spatial resolution, the size of the working volume, and the uncertainty of reconstruction. Due to the limited number of cameras and displays our system can not provide the user with a surround-immersive feeling. However, it is the first system that uses *3D-real-data* that are reconstructed *online* at another site. The system has been implemented with low-cost off-the-shelf hardware and has been successfully demonstrated in a local area network.

1 Introduction

Advances in networking and processor performance open challenging new directions for a remote collaboration via immersive environments. With the continuing progress in the bandwidth and the protocols of the information highway new education and business structures become feasible. The incorporation of graphical models in remote training is already a reality: Two astronauts from two different continents can already train together in a virtual space-shuttle [9]. However, nothing that they see is real: they see each other as their graphical avatars and the space shuttle is a virtual model. The demand for a collaboration among physicians for a common medical consultation during an operation or between engineers for virtual prototyping is increasing.

The purpose of this paper is to show in the context of teleimmersion the utility of an integrated approach coming from two fields:Computer Vision and Computer Graphics. We have embarked on the joint journey because we realized that the problem of teleimmersion, not only requires two different technologies: the data acquisition/reconstruction (the typical domain of Computer Vision) and the fast realistic and interactive data display (the typical domain of Computer Graphics) but also it requires rethinking some of the basic representations of the data in view of the constraints coming from the real time, low latency, high spatiotemporal resolution, and low cost demand.

While the Computer Vision community is mainly concerned with scene reconstruction to be used in different tasks such as navigation/manipulation or recognition, here the goal is different. In teleimmersion applications, the goal is *communication* amongst people who are geographically distributed but are meeting in the space of each local user augmented by the real live avatar of the remote partner. This is quite different from the conventional virtual reality. What is most important is not the realism but the usefulness with respect to the task in hand, for example, collaboration or entertainment. It is also different from traditional off-line versions of image-based rendering which just replace virtual with real worlds. Therefore, the challenging issue for computer vision beside the representation is the real-time processing - which was actually from the beginning a main issue for the visualization and the graphics community.

What will follow is a description of a fully integrated dynamic 3D telepresence system working over the network. The highlights of the system are:

1. Full reconstruction and transmission of dynamic *real* 3D-data in combination with any *virtual* object.

2. Real-time performance using off-the-shelf components (Intel Pentium II, Matrox Genesis, Diamond Fire GL boards).

3. Optimal balance between several quality factors (spatial resolution, depth resolution, work volume).

Why is 2D not enough ? Nowadays, most advanced teleconferencing and telepresence systems transmit 2D-images. In order to get additional views, the systems are using either panoramic systems and/or interpolate between a set of views [3, 16, 15]. We argue here, that for collaboration purposes the 3D-reconstruction can not be avoided. First, view morphing approaches are able to interpolate views over a very restricted range of weakly calibrated viewpoints. Second, even if a system is fully calibrated [15] we need a calibration between the observer tracker and the cameras. In a collaboration scenario, where multiple persons discuss real 3D properties of mechanical objects or even give instructions requiring 6DOF movements there is no camera placement constellation which can produce the required variability of viewpoints resulting from the head movements of a user and reflecting the feeling of distances. Therefore, we pursue a 3D image based rendering which involves the in vision well-known problem of stereo reconstruction.

2 Related Work

Reconstruction Here we are not going to review the huge amount of existing papers (confer the annual bibliographies by Azriel Rosenfeld) on all different aspects of stereo (the reader is referred to a standard review [5]). Application of stereo to image based rendering is very well discussed and reviewed in the recent paper by Narayanan and Kanade [13]. Although terms like virtualized reality and augmented reality are used in many reconstruction papers it should be emphasized that we address here a reactive telepresence problem whereas most image based rendering approaches try to replace a graphic model with a real one *off-line*.

Stereo approaches may be classified with respect to the matching as well as with respect to the reconstruction scheme. Regarding matching we differentiate between sparse feature based reconstructions (see the treatise in [6]) and dense depth reconstructions [14, 13] Approaches like in [2, 17] address the probabilistic nature of matching with particular emphasis on the occlusion problem. Area-based approaches [10] are based on correlation and emphasize like our approach the real-time issue.

An approach with emphasis on virtualized reality is [13]. This system captures the action of a person from a dome of 51 cameras. The processing is off-line and in this sense there is no indication how it could be used in telepresence beside the off-line reconstruction of static structures.

With respect to reconstruction, recent approaches can be classified in strongly and weakly (or self-calibrated) approaches. Self-calibration approaches [11] provide a metric reconstruction from multiple views with an accuracy which is suitable only for restricted augmented reality applications like video manipulation where the quality of depth is not relevant. Weakly calibrated approaches [8] provide a real time performance and is suitable for augmenting scenes only with synthetic objects. Our approach is the first that provides an optimal balance between depth accuracy and speed and therefore can be applied in teleimmersion.

3 System Description and Scenario

Before delving into the individual algorithms we describe here the system lay-out. Each side consists of

1. a stereo rig of two CCD-cameras,

2. a PC with a frame grabber,

3. a PC with an accelerated graphics card capable for stereo-glasses synchronization.

Both sites are hooked on the network and send their data using the TCP/IP protocol. The implementation is exactly the same for the local area network used now and for a wide area network experiment of the near future. The practical difference will be of course in the speed.

4 Stereo reconstruction

We elaborate next the main steps of the reconstruction algorithm and we place emphasis on the factors that affect the quality of reconstruction and the processing time. Our reconstruction uses two images but it is easily extensible to a polynocular configuration. We rely on the well known stereo processing steps of matching and triangulation given that the cameras are calibrated.

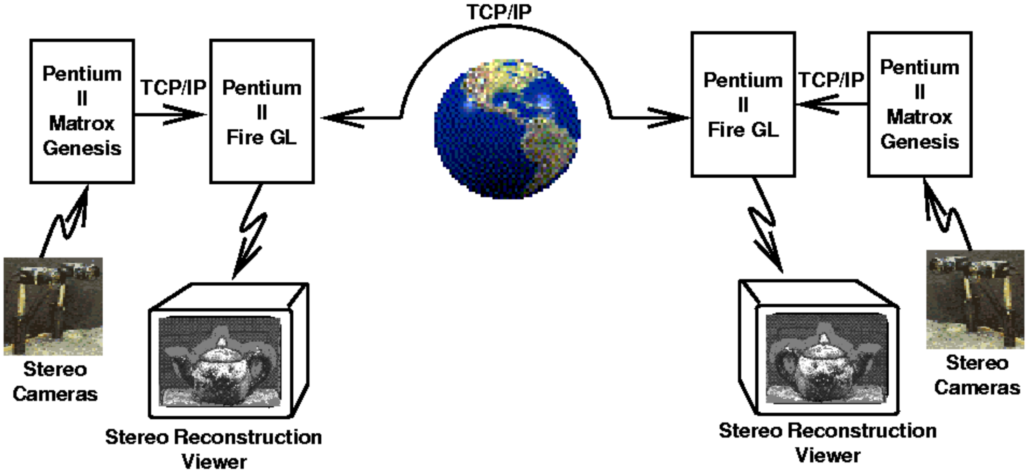
TCP/IP
Pentium II Matrox Genesis
TCP/IP
Pentium II Fire GL
Stereo Cameras
Stereo Reconstruction Viewer
Pentium II Fire GL
TCP/IP
Pentium II Matrox Genesis
Stereo Reconstruction Viewer
Stereo Cameras

Figure 1: Teleimmersion hardware set-up

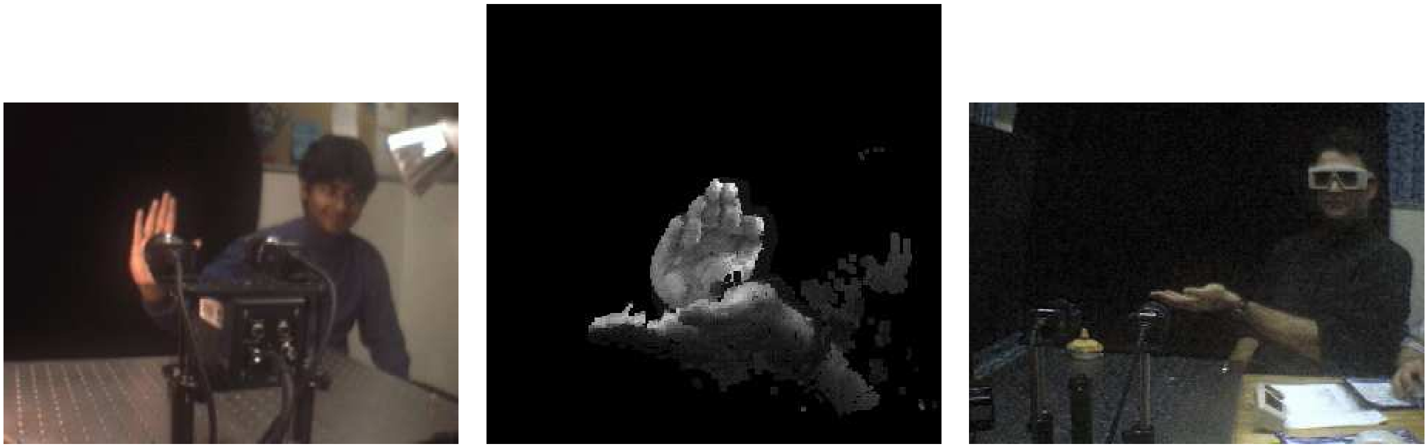

Figure 2: On the left and on the right we show the two users at each remote site, respectively. In the middle we show a 2D-image of the mixed 3D-reality that each user perceives.

Filtering It is well known that two image patches can be matched if they contain sufficient gray-value variation. Since most of the matching steps are time consuming we want to avoid them if we know a-priori that there is not sufficient image structure to match. Therefore, we compute the image gradient at each position by convolving the image with a Gaussian derivative. A subsequent thresholding extracts the image areas with a high gradient.

If the background is stationary we would like to avoid its reconstruction at each time-frame. A change detection method detects only the moving area on the image by thresholding the quotient of temporal derivative and spatial gradient.

Rectification When a 3D-point is projected onto the left and the right image plane of a fixating stereo-rig the difference in the image positions is both in horizontal and vertical directions. Given a point in the first image we can reduce the 2D-dimensional search to a 1D-dimensional if we know the so called *epipolar geometry* of the camera which is given from calibration. Because the subsequent step of correlation is area based and for reduction of time complexity we first perform a warping of the image that makes every epipolar line horizontal [1]. This image transformation is called *rectification* and results in corresponding points having coordinates (u, v) and $(u - d, v)$, in left and right rectified images, respectively, where d is the horizontal disparity.

Matching: disparity map computation The degree of correspondence is measured by a modified normalized cross-correlation [12],

$$c(I_L, I_R) = \frac{2\ cov(I_L, I_R)}{var(I_L)\ +\ var(I_R)}. \qquad (1)$$

where I_L and I_R are the left and right rectified images over the selected correlation windows. For each pixel (u, v) in the left image, the matching produces a correlation profile $c(u, v, d)$ where d ranges over a disparity range. The definition domain is the so called *disparity range* and depends on the depth of *working volume*, i.e. the range of possible depths we want to reconstruct. The time complexity of matching is linearly proportional to the size of the correlation window as well as to the disparity range.

We consider *all* peaks of the correlation profile as possible disparity hypotheses. This is different from other matching approaches which early decide on the maximum of the matching criterion. We call the resulting list of hypotheses for all positions a *disparity volume*. The hypotheses in the disparity volume are pruned out by a *selection procedure* that is based on the constraints imposed by

- Visibility: If a spatial point is visible then there can not be any other point in the viewing rays through this point and the left or right camera.

- Ordering: Depth ordering constrains the image positions in the rectified images. Both constraints can be formulated in terms of disparities without reconstructing the considered 3D-point [20, 5].

The output of this procedure is an integer *disparity map*. To refine the 3-D position estimates, a *subpixel correction* of the integer disparity map is computed which results in a subpixel disparity map. The subpixel disparity can be obtained either using a simple interpolation of the scores or using a more general approach as described in [4] which takes into account the distortion between left and right correlation windows, induced by the perspective projection, assuming that the surface can be locally approximated with a plane. The first approach is faster while the second gives a more reliable estimate of the subpixel disparity. We chose an extended version of the former which assumes preservation of the intensity value left and right. To achieve fast subpixel estimation and satisfactory accuracy we proceed as follows.

Let ϵ be the unknown subpixel correction. For corresponding pixels in the left and right images,

$$I_L(A(u,v)) = \alpha I_R(u-d+\epsilon, v) = \alpha(I_L(u-d, v)+\epsilon\nabla I_L(u-d, v)) \qquad (2)$$

where the coefficient α takes into account possible differences in camera gains. By taking a first order linear approximation of (2) over the correlation window we obtain the equivalent of a differential method for computing the optical flow. We use an FIR-filter-approximation of the image gradient appearing in the above formula. The disparity map is the input to the reconstruction procedure.

Reconstruction Each of the stereo rigs is calibrated before the experiment using a standard "strong" calibration technique [18]. The calibration estimates the two 3x4 projection matrices for the left and the right camera. Given the disparity at each point and the calibration matrix the coordinates of a 3D-point can be computed.

From the disparity maps and the camera projection matrices the spatial position of the 3D points are computed based on triangulation [6]. The result of the reconstruction (from a single stereo pair of images) is a list of spatial points.

The error in the reconstruction depends on the error in the disparity and the error in the calibration matrices. Since the action to be reconstructed is close to the origin of the world coordinate system the depth error due to calibration is negligible in comparison to the error in the disparities. What is mainly of concern is the number of outliers in the depth estimates resulting in the huge peaks usually appearing near occlusion or texture-less areas.

Once we have extracted the depth of the remote user, we augment the local user's world by putting the extracted real avatar of the remote user in it. We can further augment it with a synthetic object like a teapot by placing the synthetic object in the local user's world.

5 System Components and Performance Analysis

We give here a full transparent description of the system hardware as well as the algorithm so that it is easily reproducible and its performance can be fairly compared.

The current version at the local site (called A in Fig. 1) has an Intel Pentium-II 450 MHz and a Matrox-Genesis Frame Grabber. The latter includes the TI C80 processor as a component. The CD-cameras are the Sony XC-77. For visualization we use the Diamond FireGL-4000 board and the CrystalEyes stereo-glasses. In the current version we use an even slower Pentium II (266 MHz) at the remote site. The network part involves a 100 Mbps hub at each site.

We next describe step-by-step the algorithm and the involved parameters at each step.

1. **Rectification:** Given the calibration matrices the left and the right images are digitized and rectified using the functions supported by the C80 DSP in the frame grabber. At this step the sampling resolution must be set.

2. **Filtering:** The rectified images are filtered with a spatial and a temporal gradient. The temporal gradient is just a frame difference and the size of the spatial gradient mask is set to 7x7 which is the maximum of the filtering supported by the frame-grabber board. The result of this step is a binary mask indicating where disparity should be computed.

3. **Correlation:** The rectified images and the mask "leave" the frame-grabber board and enter the main memory. The normalized correlation is computed using 5x5 correlation windows. The search area depends on the effective focal length measured in pixels, the depth of the working volume, and the baseline of the stereo system. This is easily illustrated in the case of cameras with parallel axes where the disparity is equal the product of the focal length and the baseline divided by the depth. The effective focal length depends on the sampling, the CCD-cell size, and the real focal length. The search area is one of the main tunable parameters but a typical value used also in a comparison below is 30.

4. **Selection:** The normalized correlation values are thresholded and the selection starts according to the criteria described above. High thresholds exclude the computation of any depth whereas low thresholds allow erroneous estimation of depth at places without gray-value structure.

5. **Subpixel disparity:** The disparity is refined using the flow-like subpixel estimation.

6. **Reconstruction:** The 3D-points are computed with the help of the calibration matrices.

7. **Coloring:** The 3D-points are colored with the original image color using again the calibration matrices.

We next present a listing of the timing of every step described above for two exemplary parameter set-ups resulting in a frame-rate of 2Hz and 0.5Hz, respectively. The fast set-up has half of the resolution of the original slow set-up as well as half of the working volume. The working volume in the slow set-up is 50cm at a distance of 1m of the camera.

We do not mention the effective bandwidth of our network connection and the display speed because both of them are orders of magnitude faster than the reconstruction. In the fastest rate and dependent on the volume of data we typically require the visualization of 50K 3D points per second whose transmission results to 2 frames per sec × 50K points per frame × 15 bytes per points (3 float coordinates plus 3 RGB values) equal to 12Mbits per sec.

Step	Fast setup	Slow setup
Total time	506ms	2080ms
Rectification	26ms	110ms
Filtering	32ms	90ms
Correlation and Selection	358ms	1460ms
Subpixel disparity	42ms	270ms
Reconstr. and Coloring	48ms	150ms

Table 1: Timings of each processing step in two different qualities

The real power of the system lies in the accuracy of the depth estimation without sacrificing time. We achieve a relative depth error of less than 0.1% at a distance of 1m (less than 1mm)[1]. This is demonstrated on the accompanying video where depth discontinuities subtle structures like veins on hands are easily recognizable.

The comparison to the performance of other stereo algorithms is difficult since we have to consider both depth accuracy and speed. Furthermore, depth accuracy is measurable only on objects with known ground-truth which are difficult to compare with human figures.

There exist considerably faster systems all of them based on rough depth estimates. The Stereo Vision Machine II from SRI [7] and the Interval stereo processor [19] use a DSP C60 and an FPGA array, respectively, achieving a video frame rate (30Hz) of processing. However, their depth accuracy is not useful for close range systems because it is based on integer disparity estimation. Pentium-II based machines are the SRI-SVM-I [7] and the Point-Grey Cyclops trinocular systems which achieve 12 and 3 frames per second, respectively, are also providing us with only integer-valued disparities. It should

[1] Accompanying papers analyzing the depth error on known model objects are not cited here in order to avoid jeopardizing the blind review

be here emphasized that the Genesis frame grabber board used in our approach may have a DSP on board but still has to be considered as off-the-shelf hardware, first due to its low price and second due to the conveniency of its programming.

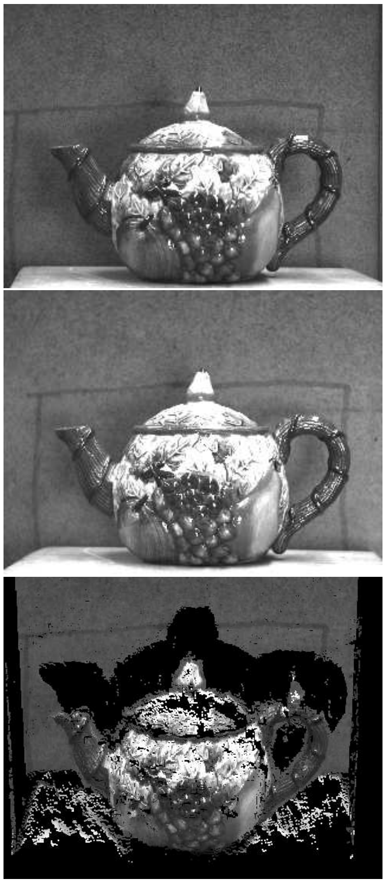

Figure 3: Left and right original image of a real teapot and rendering of the real data from another view.

We continue with showing real reconstructions. In Fig. 3 we show the original images. The depth reconstruction shown on the right of Fig. 3 and in particular the rendering from multiple head poses in the video demonstrates the reconstruction of details on the sculped surface of the real teapot but also the absence of depth values at places of specularity.

In Fig. 4 we show a snapshot of the reconstruction of a live movement of a person's head and hands whereas in the video these live data are reconstructed in a remote site and rendered locally from different viewpoints.

The third experiment shown only in the video realizes a virtual handschake rendered in a virtual space. The depth is perceived with the changing of position as well as with all occluding cues.

In the last video experiment the scenario is dynamic and involves real and virtual data. The local user holds a virtual ball in its hand and passes it to the hand of the remote user as soon as the two hands come closer. The fingertips as well as the passo-over time point are automatically detected. Unlike network-based games, here, real-3D replications of the players are projected instead of artificial avatars. Even with this simple set-up we can give a very good idea of the entertainment of the future.

6 Conclusions and the Future

We have presented a first real-time implementation of 3D-tele-immersion. The stereo-reconstruction uses state of the art stereo matching and the fusion of the two worlds is asynchronous which facilitates higher flexibility in the display site. The implementation enables the tuning of quality and working volume vs. speed. The user can choose an acceptable balance among size of working volume, depth quality, and spatial resolution.

As with many other prototypes in the history of technology it opens a bunch of challenges for all disciplines of graphics, vision, and communication. Teleimmersion is already recognized as one of the key-applications for Internet-2.

The main challenge for the vision as well as the graphics community is the issue of representation. Like the explosion of coding techniques for transmission of 2D images after the introduction of WWW we anticipate breakthroughs in problems related with representation.

The wide use of 3D-data from reconstruction raises demand for a higher quality of shape representation. We are working on the critical problems of occluding contours and specularities arising in stereo reconstruction. The dynamics of the scene necessitate shape representations that will be easily updatable using some simple assumptions on temporal coherence. Even if we use multiple cameras to obtain a surround capture we need surface parametrizations that can be also spatially registered in a simple and robust way. Last but not least, the 3D-data have to be transmitted over the network. The challenge for progressive 3D wavelet-like representations which simultaneously address the critical issues above remains open.

References

[1] N. Ayache and C. Hansen. Rectification of images for binocular and trinocular stereovision. *Proc. of 9th International Conference on Pattern Recognition*, 1:11–16, 1988.

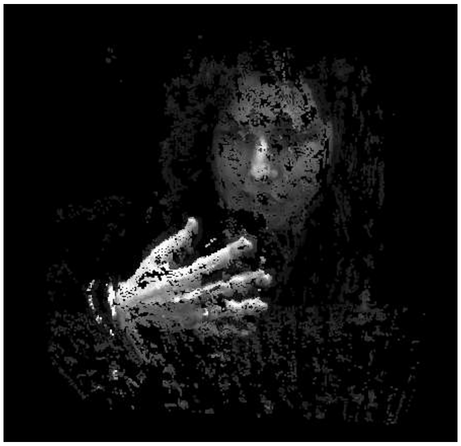

Figure 4: Rendered view of 3D-data from a reconstruction of a moving person

[2] P. Belhumeur. A bayesian approach to binocular stereopsis. *Intl. J. of Computer Vision*, 19(3):237–260, 1996.

[3] E. Chen and L. Williams. View interpolation for image synthesis. In *ACM SIGGRAPH*, 1993.

[4] F. Devernay. Computing differential properties of 3-D shapes from stereoscopic images without 3-D models. *INRIA, Sophia Antipolis, RR-2304*, 1994.

[5] U. Dhond and J. Aggrawal. Structure from stereo: a review. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1489–1510, 1989.

[6] O. Faugeras. *Three-dimensional Computer Vision*. MIT-Press, Cambridge, MA, 1993.

[7] K. Konolige. Small vision system: Hardware and implementation. *Eighth International Symposium on Robotics Research, Hayama, Japan*, 1997.

[8] K. Kutulakos and J. Vallino. Calibration-free augmented reality. *IEEE Trans. on Visualization and Computer Graphics*, 4(1):1–20, 1998.

[9] M. Macedonia and S. Noll. Real-time 60hz distortion correction on a silicon graphics ig. *IEEE Computer Graphics and Applications*, 5:76–82, 1998.

[10] L. Matthies. Stereo vision for planetary rovers: Stochastic modeling to near real-time implementation. *International Journal of Computer Vision*, 8:71–91, 1992.

[11] S. Maybank and O. Faugeras. A theory of self-calibration of a moving camera. *Intl. J. of Computer Vision*, 8(2):123–151, 1992.

[12] H. Moravec. Robot rover visual navigation. *Computer Science: Artificial Intelligence*, pages 105–108, 1980/1981.

[13] P. Narayanan, P. Rander, and T. Kanade. Constructing virtual worlds using dense stereo. *Proc, Intl. Conf. Computer Vision ICCV98*, pages 3–10, 1998.

[14] M. Okutomi and T. Kanade. A multiple-baseline stereo. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15(4):353–363, 1993.

[15] D. Scharstein and R. Szeliski. Stereo matching with non-linear diffusion. *Proc. Int. Conf. Computer Vision and Pattern Recognition*, 1996.

[16] S.M. Seitz and C.R. Dyer. Towards image-based scene representation using view morphing. In *ACM SIGGRAPH*, 1996.

[17] C. Tomasi and R. Manduchi. Stereo without search. *Proc. European Conf. Computer Vision*, 1996.

[18] R. Tsai. A versatile camera calibration technique for high accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Trans. Robotics and Automation*, 3:323–344, 1987.

[19] J. Woodfill and B. Von Herzen. real time stereo vision on the parts reconfigurable computer. In *IEEE Workshop on FPGAs for Custom Computing Machines*, 1997.

[20] A. Yuille and T. Poggio. A generalized ordering constraint for stereo correspondence. AI Lab Memo 777, MIT, 1984.