

## View-Independent Scene Acquisition for Tele-Presence

Jane Mulligan and Kostas Daniilidis  
 University of Pennsylvania, GRASP Laboratory\*  
 {janem,kostas}@grip.cis.upenn.edu

### Abstract

*Tele-immersion is a new medium that enables a user to share a virtual space with remote participants. The user is immersed in a rendered 3D-world that is transmitted from a remote site. To acquire this 3D description we apply bi- and trinocular stereo techniques. The challenge is to compute dense stereo range data at high frame rates, since participants cannot easily communicate if the processing cycle or network latencies are long. Moreover, new views of the received 3D-world must be as accurate as possible. We address both issues of speed and accuracy and we propose a method for combining motion and stereo in order to increase speed and robustness.*



**Figure 1. A user in Chapel Hill communicates with remote users from Philadelphia (left) and Armonk (right).**

### 1 Introduction

The power of today's general purpose and graphics processors and the high bandwidth of the recent Internet generations, provide the necessary infrastructure for mixed reality systems which can augment the user's senses and create the sense of tele-presence. In this paper we describe our contribution to the realization of a new mixed reality medium called tele-immersion. Tele-immersion enables users in physically remote spaces to collaborate in a shared space that mixes the local with the remote realities. The concept of tele-immersion involves all visual, aural, and haptic senses. To date, we have dealt only with the visual part, and in collaboration with the University of North Carolina (Henry Fuchs and co-workers) and Advanced Network and Services (Jaron Lanier), we have accomplished a significant step toward realization of visual tele-immersion.

Our accomplishment is best illustrated in Fig. 1

\*The financial support by Advanced Networks and Services and ARO/MURI-DAAH04-96-1-0007, NSF-CISE-CDS-97-03220, DARPA-ITO-MARS-DABT63-99-1-001 is gratefully acknowledged. We thank Jaron Lanier, Henry Fuchs, and Ruzena Bajcsy for their wonderful leadership in this project and Herman Towles, Wei-Chao Chen, Ruigang Yang (UNC) and Amela Sadagic (Advanced Netw. and Serv.) for the so productive collaboration.

taken during the first full scale demonstration at the University of North Carolina. The display set-up has been designed by Henry Fuchs[8]. A user wears passive polarized glasses and an optical tracker [10, Hi-ball] which captures the head's pose. On the two walls two realities, the Philadelphia, and the Armonk reality, respectively, are stereoscopically displayed from polarized pairs of projectors. The static parts of the two scenes are view-independent 3D descriptions acquired off-line. The 3D-descriptions of the persons in the fore-ground are acquired in real-time at the remote locations and transmitted over the network. The projections on the walls are dynamically rendered according to the local user's viewpoint, and updated by real-time real-world reconstructions to increase the feeling of sharing the same conference table.

There are two alternative approaches in remote immersion technologies we did not follow. The first involves video-conferencing in the large: surround projection of 2D panoramic images. This requires only a correct alignment of several views, but lacks the sense of depth and practically forbids any 3D-interaction with virtual/real objects. The second technology is closer to ours [3] and uses 3D-graphical descriptions of the remote participants (avatars). In the system description which follows, the reader will realize that

such a technique could be merged with our methods in the future if we extract models based on the current raw depth points. This is just another view of the model-free vs model-based extrema in the 3D-descriptions of scenes or the bottom-up vs top-down controversy. Assuming that we have to deal with persons, highly detailed human models might be applied or extracted in the future. However, the state of avatar-based tele-collaboration is still on the level of cartoon-like representations.

Comparing tele-immersion to classical augmented reality we find that real-time head tracking and display refresh rate pose minor problems. The challenging difference is, first, that the display used is a spatially augmented display and not an HMD and, second, that the mixed components are not pre-stored perfect virtual objects, but on-line acquired real range data. In addition these data are transmitted over the network before displayed. Sense of presence really depends on real-time responses and accurate depth estimation with respect to the viewer. In this paper, we will describe the real-time 3D acquisition of the dynamic parts of a scene which in Fig. 1 are the persons in the foreground. The approach we chose to follow is *view-independent* scene acquisition. Having acquired a scene snapshot at a remote site we transmit it represented with respect to a world coordinate system. Display from a new point of view involves only primitive transformations hard-wired in every graphics processor. In addition to real time, we want the new view to be error free so that the user does not experience wrong depths through her polarized stereo glasses. The basic question is how to achieve a perceptually best reconstruction in real-time. We have to emphasize that these criteria are stricter than those in navigation, for example. Navigational stereo targets a convex-hull based representation whereas the user here must be able to see features as detailed as a face profile reconstructed from frontal views.

We will not review<sup>1</sup> the huge number of existing papers (see the annual bibliographies by Azriel Rosenfeld) on all aspects of stereo (the reader is referred to a standard review [1]). Application of stereo to image based rendering is very well discussed and reviewed in the recent papers by Narayanan and Kanade [5] and Szeliski [9].

## 2 System's Overview and Architecture

A tele-immersion telecubicle is intended both to acquire a 3D model of the local user and environment for rendering and interaction at remote sites, and to

<sup>1</sup>Regarding related work the reader is referred to <http://www.cis.upenn.edu/janem/techrep00.pdf>.



Figure 2. Camera configuration, user view.

provide an immersive experience for the local user via head tracking and stereoscopic display projected on large scale viewscreens.

In the current set-up none of the participating sites has a full version of the telecubicle. Instead, the display site is as illustrated in Figure 1 and the acquisition site consists only of a camera cluster as illustrated in Figure 2. A cluster consists of 7 firewire cameras arranged on an arc at 15° separation to ‘surround’ the user and prevent any break presence due to a hard edge where the reconstruction stops. These cameras are used to calculate binocular or trinocular stereo depth maps from overlapping pairs or triples.

We are assuming remote sites have a static background scene which can be reconstructed and transmitted in advance. As a result we need a method to segment out the static parts of the scene online. We have chosen to implement the background subtraction method in [4]. A sequence of  $N$  (2 or more) background images  $B_i$  are acquired in advance of each session. From this set we compute a pixelwise average background image  $\bar{B} = \frac{1}{N} \sum_i B_i$ . We then compute the average pixelwise difference between  $\bar{B}$  and  $B_i$ ,  $\bar{D} = \frac{1}{N} \sum_i (\bar{B} - B_i)$ . During a session each primary image  $I$  is subtracted from the static mean background and thresholded  $I_B = (\bar{B} - I) > T \times \bar{D}$ . A series of erosions and dilations are performed on  $I_B$  in order to sharpen the background mask.

The reconstruction algorithm begins by grabbing images from 2 or 3 strongly calibrated cameras. The system rectifies the images so that their epipolar lines lie along the horizontal image rows to reduce the search space for correspondences, and so that corresponding points lie on the same image lines.

Real-time stereo matching suggests one of several correlation based matching metrics. In particular we have focussed on Sum of Absolute Differences (SAD), because of the speed provided by hardware specific operations, and Modified Normalized Cross Correlation (MNCC), which produces superior depth maps in the binocular case. In general the SAD calculation is:  $corr_{SAD}(I_L, I_R) = \sum_W |I_L - I_R|$  for a window  $W$  in rectified images  $I_L$  and  $I_R$ . The disparity  $d$  determines



Figure 3. Trinocular triple.



Figure 4. Rendered reconstructions, profile view. (a) Binocular MNCC; (b) trinocular MNCC.

the relative window position in the right and left images.

A better correspondence metric is modified normalized cross-correlation (MNCC),  $corr_{MNCC}(I_L, I_R) = \frac{2 \text{COV}(I_L, I_R)}{\sigma^2(I_L) + \sigma^2(I_R)}$ . 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 metrics above produce a correlation profile  $c(u, v, d)$  where disparity  $d$  ranges over acceptable integer values. Selected matches are maxima (for MNCC) or minima (for SAD) in this profile.

The trinocular epipolar constraint is a well known technique to refine or verify correspondences and improve the quality of stereo range data. A match in a pair of images is correct if the epipolar lines for the original point  $[u, v]$  and the hypothesized match  $[u - d, v]$ , intersect in the third camera image [1]. Trinocular camera triples are usually arranged in a right angle, allowing matching along the rows and columns of the reference image [6, 2]. Our surround camera configuration does not allow us to arrange or rectify triples of camera image planes such that they are coplanar, and therefore it is more expensive for us to exploit the trinocular constraint.

Following Okutomi and Kanade’s observation [7], we optimize over the sum of correlation values with respect to the true depth value rather than disparity. Essentially we treat the camera triple  $\langle L, C, R \rangle$  as two independent stereo pairs  $\langle L, C_L \rangle$  and  $\langle C_R, R \rangle$ . As a result of using foreground segmentation we need consider only one half to one third of the pixels in the reference image  $C_R$ . This makes it feasible to calculate the entire correlation profile for each pixel one at a time. To calculate the sum of correlation scores we precompute a lookup table of

Step	SAD	MNCC	Tri-SAD	Tri-MNCC
Rectify	49	50	49	48
Background	18	18	18	18
Matching	182	261	390	791
Reconstruct	6	6	7	6
Total	446 ms	520 ms	662 ms	1067 ms
fps	2.2	1.9	1.5	0.9

Table 1. Frames per second (fps) values include 160 ms capture time and 6 ms network transmission overhead.

the location in  $C_L$  corresponding the current pixel in  $C_R$  (based on the right-left rectification relationship). We also compute a linear approximation for the disparity  $\widehat{d}_L = M(u_{C_R}, v_{C_R}) \times d_R + b(u_{C_R}, v_{C_R})$  at  $[u_{C_L}, v_{C_L}]$  which arises from the same depth point as  $[u_{C_R}, v_{C_R}, d_R]$ . As we calculate the correlation score  $corr_R(u_{C_R}, v_{C_R}, d_R)$ , we look up the corresponding  $[u_{C_L}, v_{C_L}]$  and compute  $\widehat{d}_L$ , then calculate the correlation score  $corr_L(u_{C_L}, v_{C_L}, \widehat{d}_L)$ . We select the disparity  $d_R$  which optimizes  $corr_T = corr_L(u_{C_L}, v_{C_L}, \widehat{d}_L) + corr_R(u_{C_R}, v_{C_R}, d_R)$ .

The method can be summarized as follows:

#### Pixelwise Trinocular Stereo

**Step 1:** Precompute lookup table for  $C_L$  locations corresponding to  $C_R$  locations, and approximation lookup tables  $M$  and  $b$

**Step 2:** Acquire image triple  $\langle L, C, R \rangle$

**Step 3:** Rectify  $\langle L, C_L \rangle$  and  $\langle C_R, R \rangle$  independently.

**Step 4:** Calculate foreground mask for  $C_R$

**Step 5:** for every foreground pixel

**Step I:** for every disparity  $d_R \in D_r$

**Step i:** compute  $corr_R(u_{C_R}, v_{C_R}, d_R)$

**Step ii:** lookup  $[u_{C_L}, v_{C_L}]$

**Step iii:** compute  $\widehat{d}_L = M(u_{C_R}, v_{C_R}) \times d_R + b(u_{C_R}, v_{C_R})$

**Step iv:** compute  $corr_L(u_{C_L}, v_{C_L}, \widehat{d}_L)$

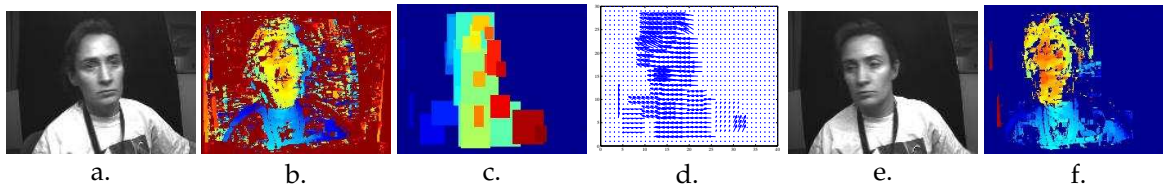
**Step v:**  $corr_T = corr_L + corr_R$

**Step vi:** Update  $corr_{best}, d_{best}$

**Step 6:** Goto 2

As expected SAD implementations were faster than MNCC based implementations. All implementations ran on a quad-PIII 550 MHz server in 1 second or less, including image acquisition and transfer and transmission of reconstructions to the renderer. Timings for the various systems are presented in Table 1.

For tele-immersion we are further interested in the quality and density of depth points. Although the computation times were greater, the high quality of



**Figure 5. Frame 12 (left) of stereo sequence (a), and computed disparity image (b); Extracted windows (c) and calculated flow per window (d); Frame 18 of sequence (e) and region based disparity map (f).**

trinocular depth maps makes them a desirable alternative to faster but noisier SAD range images. Figure 3 illustrates a trinocular triple and Figure 4 (a) and (b) the resulting rendered depth maps for binocular MNCC (right pair) and trinocular MNCC respectively. The improvement in depth map from use of the trinocular constraint is evident in the reduction of noise speckle and refinement in profile detail.

### 3. Motion-based Enhancements

The dominant cost in stereo reconstruction is that of the correlation match itself, in general proportional to  $N \times M \times D$  for images of size  $N \times M$  and  $D$  tested disparity values. By using background subtraction in our application we have reduced the number of pixels considered by the search to one half to one third of the total  $N \times M$ . To reduce the matching costs further, we would like to reduce  $D$ , the number of disparities considered for each of the remaining pixels. However, to maintain a seamless immersive experience, we cannot greatly restrict the motion of subjects in the stereo workspace which will depend on  $D$ .

For online stereo reconstruction at high frame rates there will be considerable similarity between successive images. We can exploit this temporal coherence in order to further optimize our online calculations. We propose a simple segmentation of the image, based on finding regions of the disparity image which contain only a narrow range of disparity values. Using a per region optical flow calculation we can estimate the location of the region in future frames, and bound its disparity search range  $D_i$ .

Our method for integrating disparity segmentation and optical flow can be summarized as follows:

**Step 1:** Bootstrap by calculating a full disparity map for the first stereo pair of the sequence.

**Step 2:** Use flood-fill to segment the disparity map into rectangular windows containing a narrow range of disparities.

**Step 3:** Calculate optical flow per window for left and right sequences.

**Step 4:** Adjust window positions, and disparity ranges according to estimated flow.

**Step 5:** Correlation matching using assigned disparity range, selecting 'best' correlation value over all windows and disparities associated with each pixel location.

**Step 6:** Goto Step 2.

Most time critical systems using correlation matching will benefit from this approach as long as the expense of propagating the windows via optical flow calculations is less than the resulting savings over the full image/full disparity match calculation.

Figure 5 a), b), and c) illustrate an image, its disparity map and the rectangular regions extracted via flood-fill; d) shows the regions with their flow values marked and e) and f) show a later frame and the disparity map calculated from propagated regions of similar disparity.

### References

- [1] U. Dhond and J. Aggrawal. Structure from stereo: a review. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1489–1510, 1989.
- [2] O. Faugeras. *Three-dimensional Computer Vision*. MIT-Press, Cambridge, MA, 1993.
- [3] J. Leigh, A. Johnson, M. Brown, D. Sandin, and T. Defanti. Visualization in teleimmersive environments. *Computer*, 32(12):66–73, 1999.
- [4] F. C. M. Martins, B. R. Nickerson, V. Bostrom, and R. Hazra. Implementation of a real-time foreground/background segmentation system on the intel architecture. In *IEEE ICCV99 Frame Rate Workshop*, Kerkyra, Greece, 1999.
- [5] P. Narayanan, P. Rander, and T. Kanade. Constructing virtual worlds using dense stereo. In *Proc. Int. Conf. on Computer Vision*, pages 3–10, 1998.
- [6] Y. Ohta, M. Watanabe, and K. Ikeda. Improving depth map by right-angled trinocular stereo. In *ICPR'86*, volume I, pages 519–521, Paris, France, 1986.
- [7] M. Okutomi and T. Kanade. A multiple-baseline stereo. *IEEE PAMI*, 15(4):353–363, 1993.
- [8] R. Raskar, G. Welch, M. Cutts, A. Lake, L. Stesin, and H. Fuchs. The office of the future: A unified approach to image-based modeling and spatially immersive displays. In *ACM SIGGRAPH*, pages 179–188, 1998.
- [9] R. Szeliski. Stereo algorithms and representations for image-based rendering. In *British Machine Vision Conference*, pages 314–328, Nottingham, England, September 1999.
- [10] G. Welch and G. Bishop. Scaat: Incremental tracking with incomplete information. In *ACM SIGGRAPH*, 1997.