3D-Mosaicing from Stereo video

Kiran S. Bhat
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
kiranb@cs.cmu.edu

Abstract

We describe an algorithm to register images obtained from multiple viewpoints that are translated and rotated with respect to each other. Range information at these viewpoints are obtained from a stereo pair that is mounted on a mobile cart. We recover the rotation and translation parameters by minimizing the sum-squared difference of the image intensities. We use Levenberg Marquardt optimization techniques to perform the minimization. We register all the images into a single viewpoint, resulting in a 3D mosaic. Using the registration information, we can render images from novel camera viewpoints by applying view interpolation techniques.

1. Introduction

Image mosaicing techniques have been used widely in tele-reality applications like virtual walkthrough of real building interiors [Szeliski[1,2],Irani[3]]. Current techniques for creating a virtual walkthrough rely on building a panoramic mosaic of the office interior at several key locations,called “hot-spots” in the room [Eric Chen[4], Quicktime website[5]]. The user can pan & tilt a virtual camera at these hot-spots, and the system renders an appropriate view of the environment. Moreover, the user can jump between the hot-spots to get a global view of the environment. To realistically model a large environment using this technique, one needs to create a large number of mosaics from several hot-spots. This process can be extremely long, and involves a lot of human intervention. Moreover, since the transition from the hot-spots is discrete, the walkthrough is not smooth. The fundamental reason for these drawbacks is due to the fact that the standard image mosaicing techniques register images that are obtained from the same center of projection. In this work, we address the problem of registering images obtained from camera viewpoints that are translated and rotated relative to each other. Our technique requires the depth information from each viewpoint, which we obtain from stereo. After registration, we build a composite image from a single viewpoint, which we call a “3D mosaic”. This information can also be used to render views from novel camera viewpoints using standard view interpolation techniques[Chen and Williams[6]]. Our technique can be used for both building a panoramic mosaic at hot-spots and smoothly transition between them by controlling the new viewpoint. Note that our technique does not need a 3D model for rendering from a novel viewpoint. The performance of our registration and rendering algorithms depend critically on the stereo disparity results. We use the cooperative stereo matching algorithm developed by Zitnick and Kanade [Zitnick[7]] to obtain disparities from stereo. The algorithm for registering images from multiple viewpoints is discussed in section 2. Section 3,4 and 5 shows results of 3D mosaicing and rendering from novel viewpoints of a stereo sequence obtained from an office interior. Finally, we discuss some limitations of our work, and possible future work to improve the performance of the algorithm.

2. Registration

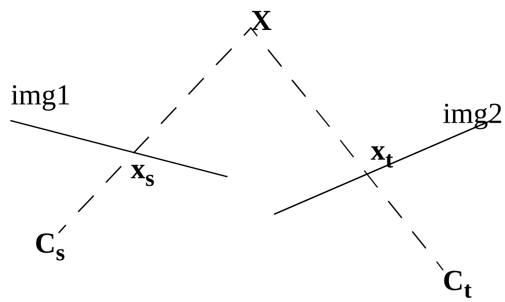

Figure 1: Scene Geometry

This section describes the relationship between corresponding pixels of two images that are rotated and translated with respect to each other. Let ***Img1*** and ***Img2*** be two images grabbed by one of the stereo cameras (say the left camera) at two spatial locations of the cart. $\mathbf{C_s}$ and $\mathbf{C_t}$ are the centers of projections of the two images. Let **X** be a 3D point in the world and let $\hat{x}_s$ and $\hat{x}_t$ be the projections of **X** in the two images. The relationship between **X** and $\hat{x}_s$ is given by the projection matrix $\Pi = [P| - PC]$

$X = C_s + \lambda P^{-1}\hat{x}_s$ and $X = C_t + \lambda P^{-1}\hat{x}_t$ where

$\lambda = \frac{1}{\delta}$ = (baseline)/(stereo disparity). The quantity δ is called the generalized disparity which is the ratio of stereo disparity to the stereo baseline. This quantity also measures the ratio of focal length of the camera to the z-depth of a point (f/z). In our formulation, since we have the stereo disparities at both the cart positions, δ is a known at each pixel in both images. Simplifying the above equations, we get

$$\hat{x}_t = \delta P_t(C_s - C_t) + P_t P_s^{-1}\hat{x}_s \tag{1}$$

The second term of the RHS describe the homography between the two images which can be parametrized by 8 parameters {m0,m1,..m7}. The first term is the displacement in the direction of the epipole which can be characterized by 3 parameters $\{t_x,t_y,t_z\}$ In matrix form:

$$P_t P_s^{-1} = \begin{bmatrix} m0 & m1 & m2 \\ m3 & m4 & m5 \\ m6 & m7 & 1 \end{bmatrix} \tag{2}$$

$$P_t(C_s - C_t) = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \tag{3}$$

Substituting (2) and (3) into (1), we get the following expressions:

$$\hat{x}_s = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} m0 & m1 & m2 \\ m3 & m4 & m5 \\ m6 & m7 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} + \delta \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \tag{4}$$

$$x' = \frac{(m0)x + (m1)y + m2 + \delta t_x}{(m6)x + (m7)y + 1 + \delta t_z} \tag{5}$$

$$y' = \frac{(m3)x + (m4)y + m5 + \delta t_y}{(m6)x + (m7)y + 1 + \delta t_z} \tag{6}$$

where $\hat{x}_s = [x', y', 1]^T$ and $\hat{x}_t = [x, y, 1]^T$. We observe that the mapping from $\hat{x}_t$ to $\hat{x}_s$ can be many to one, i.e., many pixels from Img1 can map to a single pixel in Img2. We handle the visibility problem by maintaining a Z buffer of the disparity values at each $\hat{x}_t$, and using the standard Z buffer algorithm.

To recover the parameters {m0...m7} and {t_x,t_y,t_z}, we minimize the sum of squared difference in image intensities of all corresponding pixels over the overlapping portions of the two images. The error function is given by

$$E(m, t) = \sum_i [Img2(\hat{x}_t) - Img1(\hat{x}_s)]^2 = \sum_i e_i^2$$
$$i \in OverlapRegions$$

To perform minimization, we use the Levenberg-Marquardt non-linear minimization algorithm. This algorithm requires the computation of the partial derivatives of e_i with respect to the unknown parameters {m0...m7,t_x,t_y,t_z}. They are computed as follows:

$$\frac{de_i}{dm0} = \frac{x_i}{D_i}\frac{\partial}{\partial x'}Img2$$

$$\frac{de_i}{dt_z} = -\frac{\delta}{D_i}\left[x'\frac{\partial}{\partial x'}Img2 + y'\frac{\partial}{\partial y'}Img2\right]$$

From these partials, we compute the approximate hessian matrix **A** and the weighted gradient vector **b** given by

$$a_{kl} = \sum_i \frac{\partial e_i}{\partial m_k}\frac{\partial e_i}{\partial m_l}, \quad b_k = -2\sum_i e_i \frac{\partial e_i}{\partial m_k}$$

We update the parameters $\{\mathbf{m,t}\}$ by an amount $(A+\mu I)^{-1}b$, where μ is a adaptive parameter. μ is initialized to a very small number (say 0.001) initially. At any step, if the current error is greater than the error at the previous step, we multiply μ by a factor of 10, else we divide it by the same factor.We implemented the registration algorithm in a coarse to fine scheme using a gaussian pyramid scheme. We built a gaussian pyramid comprising of 3 levels, where each level is obtained by low-pass filtering and subsampling the next higher level by a factor of two. At the coarsest level, we solve for the translation parameters $\{t_x,t_y,t_z\}$ alone. At the middle level, we solve for the eight homography parameters {m0..m7} with fixed values for translation parameters obtained from the previous level. At the highest level, we solve for all the eleven parameters. We find that the coarse to fine scheme helps a lot in converging to the correct local minima. At each level, the algorithm attains a local minima in at most 10 steps.

Registration results

Figure 2: Registration Results

Figure 2 show the registration results obtained for a pair of consecutive images taken from the stereo cart using the 11-parameter algorithm. The images in the first row are the input images to the registration algorithm, and the image in the bottom row shows the result of Img2 (the right image in the first row) warped to Img1 (the left image in the first row). We can see that the registration is pretty accurate in the regions of overlap, and the warped image matches Img1 almost perfectly.

3. Single Viewpoint Mosaic

Given that we can register two images taken at different centers of projection and having different orientations with respect to each other, we can warp one of the images into the viewpoint of the other image and obtain a composite image. We call this composite image as a *3D mosaic*. Figure 3 shows an example of a 3D mosaic created from a sequence of images, two of which are also shown.

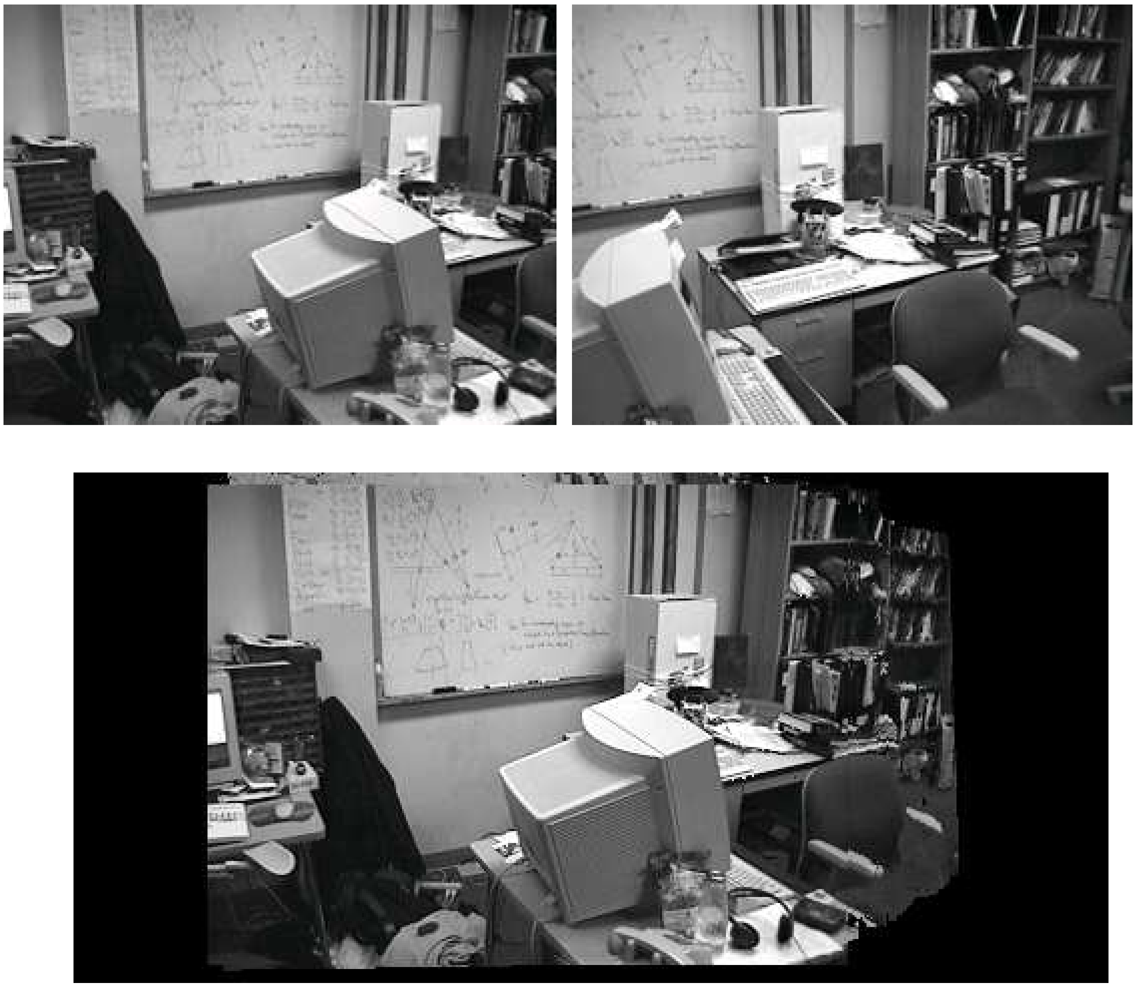

Figure 3: 3D mosaic obtained from two views

In building this mosaic, we warp Img2 (top right image) onto the viewpoint of the Img1 (top left image). The blending scheme uses the contributions of Img2 only outside the boundaries of Img1. In these regions, we maintain correct visibility by maintaining a Z buffer of the disparity values. We observe that the registration is fairly accurate, and the resulting 3D mosaic correctly depicts the scene from the viewpoint of the first camera. However, some of the regions look blurry, and we think that this is caused due to the noise in the disparity map obtained from stereo algorithms.

4. Rendering from novel camera viewpoint

One potential application of our registration technique is to render images from novel camera viewpoints. Figure4(a) shows two images taken at two distinct camera viewpoints. We compute the registration between these two views using the 11 parameter transformation. Figure 4(b) shows a real image captured at an intermediate camera viewpoint and figure 4(c) shows a rendered image at this intermediate view using data from images in fig. 4(a) using view interpolation techniques. Also shown in fig.4(d) is the error between the real image and the rendered image. This figure shows that it is possible to create accurate rendering at novel viewpoints using our technique without explicitly building a 3D model of the scene.

Figure 4(a): Img1

Img2

Figure 4(b): Intermediate “real” view

Figure 4(c): Intermediate “rendered” view

Figure 4(d): Difference Image

5. Results

We present some additional results of single viewpoint mosaic and rendering from novel viewpoints. Since we perform forward warping, there are a few holes in the resulting

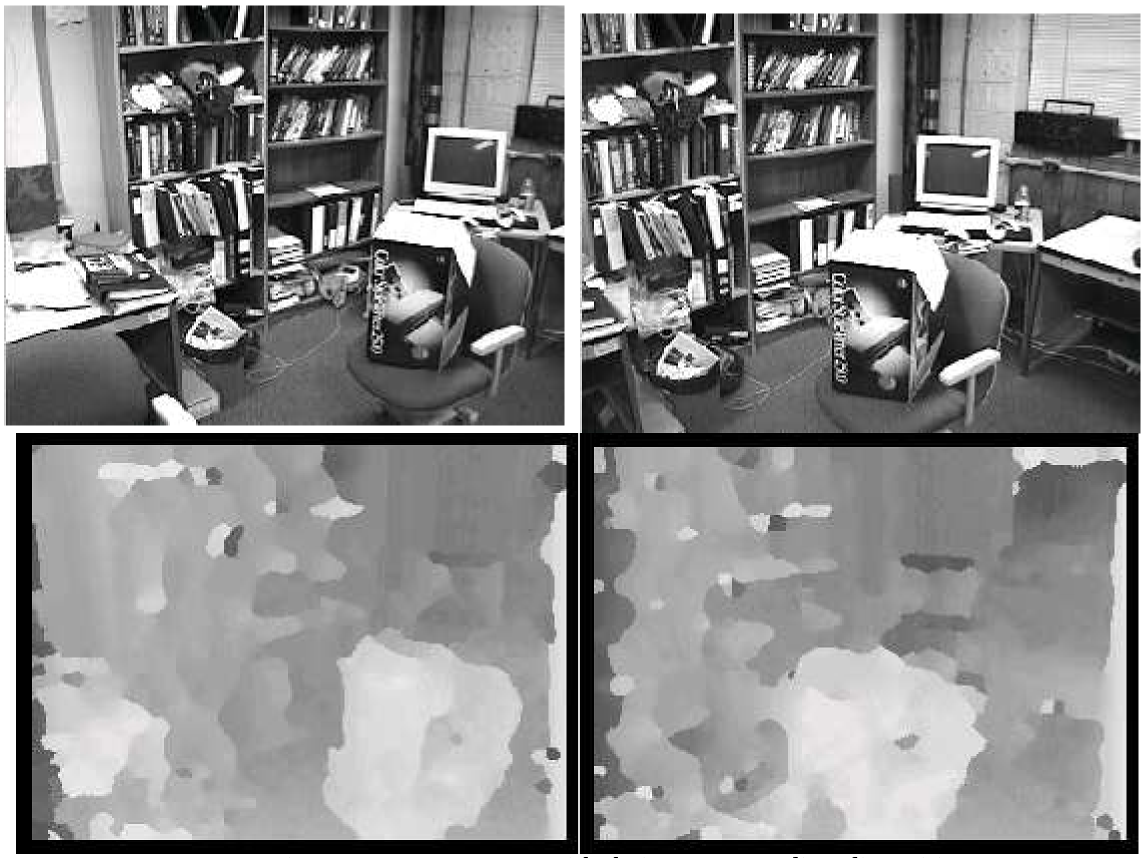

Figure 5: Input Images with their corresponding disparities

warped image (the new view). We fill these holes by median filtering the warped images (which blurs the images), and then performing a image sharpening operation. So, the resultant images do not look “exactly” like the original images, but look slightly blurred.

Figure 6(a): 3D mosaic

Figure 6(b): 3D mosaic from a different viewpoint

Figure 7: Rendered views from novel viewpoints

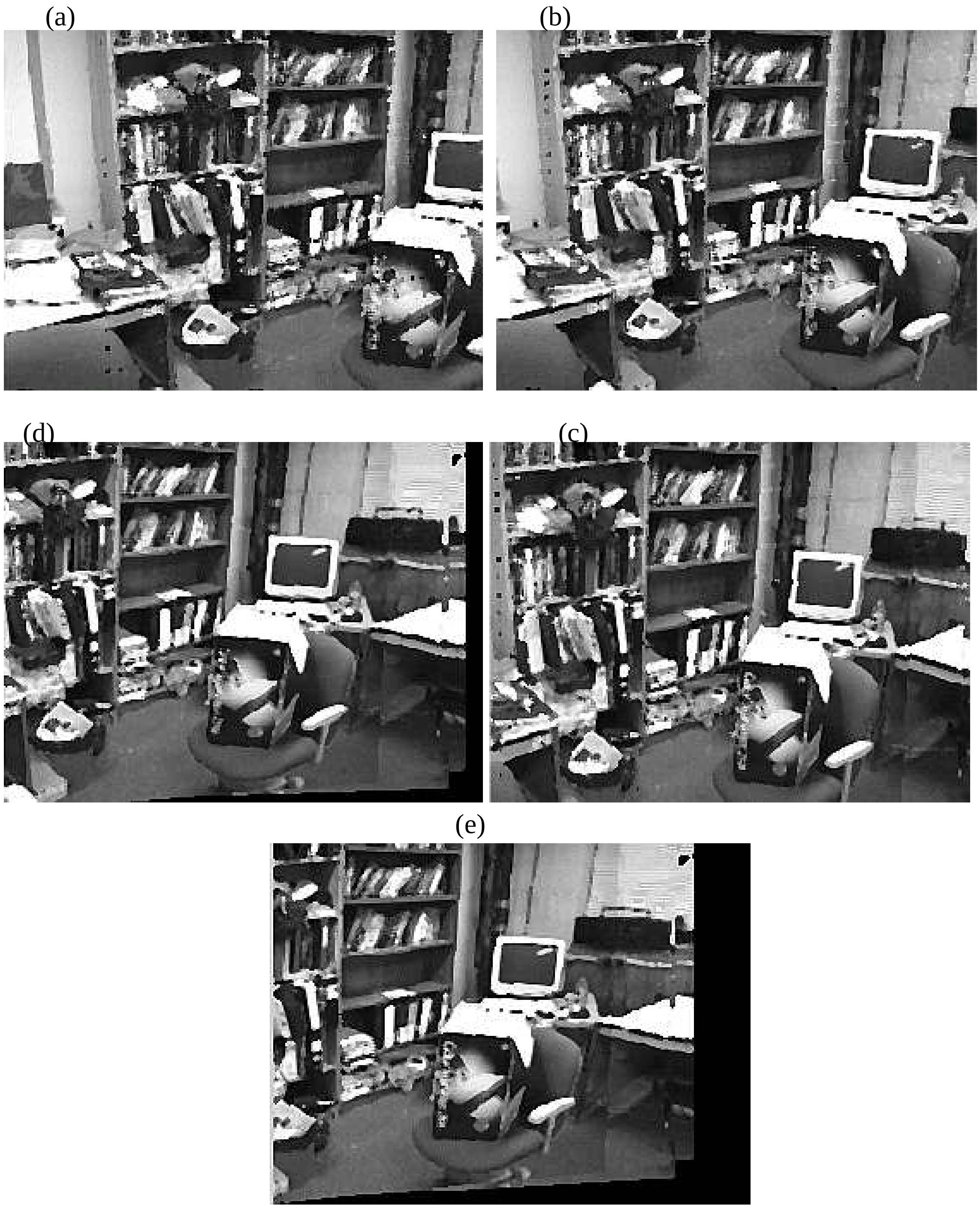

Figure 8(a-e): Rendered views from a rotating camera. The “virtual” camera rotates from left to right in this sequence.

6. Conclusions and future work

We present a technique to register images obtained from multiple viewpoints. We can build a composite 3D image mosaic by warping the multiple views into a single viewpoint. We also demonstrate some rendering results from novel viewpoints. Since the input images are obtained from cameras that can rotate and translate relative to each other, we need just six parameters (rigid body motion) to characterize the registration. We however use 11 parameters for this registration, with five redundant parameters. Formulating the problem with 6 parameters could improve the registration performance, since we are now optimizing over a lower dimensional space (6 as opposed to 11 dimensions). Another limitation of our approach is stereo. The quality of the rendered scene depends directly on the accuracy of the stereo results.

Future work: We would like to implement the 6 parameter formulation for registration, and compare the results with the current formulation. Since we have multiple registered views of a scene, we can correct for errors in the stereo disparity maps caused due to occlusions. Currently, our 3D mosaic is obtained by projecting all images onto a single viewpoint. We can extend this to an orthographic projection. We can also construct an LDI of the 3D mosaic (Layered depth mosaic!) by incorporating multiple depths at each pixel.

References

1. R.Szeliski. Image mosaicing for tele-reality applications. In IEEE Workshop on Applications of Computer Vision, pp 44-53, December 1994.

2. R. Szeliski and H. Shum. Creating full view panoramic image mosaics and environment maps. Computer Graphics (SIGGRAPH’97), pp. 251-258, 1997.

3. M. Irani, P. Anandan, J. Bergen, R. Kumar and S. Hsu. Mosaic representations of video sequences and their applications. Signal Processing: Image Communication, special issue on Image and Video semantics: Processes, Analysis and Applications, Vol 8, No. 4, pp 327-351, May 1996

4. S.E. Chen. Quicktime VR - an image based approach to virtual environment navigation. Computer Graphics (SIGGRAPH ‘95), pp 29-38, August 1995.

5. Quicktime VR demos- http://www.apple.com/quicktime/qtvr/

6. S. Chen and L. Williams. View interpolation for image synthesis. Computer Graphics (SIGGRAPH ‘93), pp 279-288, August 1993.

7.C. Zitnick and T. Kanade. A Cooperative Algorithm for Stereo Matching and Occlusion Detection. tech. report CMU-RI-TR-99-35, Robotics Institute, Carnegie Mellon University, October, 1999.