Efficient Depth Estimation using Trinocular Stereo*

Charles Hansen† Nicholas Ayache and Francis Lustman

INRIA
Domaine de Voluceau, Rocquencourt
BP 105
78153 Le Chesnay Cédex

Abstract

We present recent advancements in our passive trinocular stereo system. These include a technique for calibrating and rectifying in a very *efficient* and *simple* manner the triplets of images taken for trinocular stereovision systems. After the rectification of images, epipolar lines are parallel to the axes of the image coordinate frames. Therefore, potential matches between the three images satisfy simpler relations, allowing for a less complicated and more efficient matching algorithm. We also describe a more robust and general control strategy now employed in our trinocular stereo system. We have also developed an innovative method for the reconstruction of 3-D segments which provides better results and a new validation technique based on the observation that neighbors in the image should be neighbors in space. Experiments are presented demonstrating these advancements.

1 Introduction

Computational stereo provides an attractive solution for the problem of recovering depth from 2-D images. One simply needs to determine a homologue (i.e. solve the correspondence problem) between the images and depth is easy to recover by using projective geometry. Computational stereo can be divided into two classes with respect to the primitives used for matching: intensity based and feature based. In recent years, feature based stereo has proven to be more reliable and robust than intensity based methods. However, such techniques have been plagued by slow execution time due to the necessity of detecting and then matching features. Indeed, the crucial and most time consuming portion of computational stereo is the matching process. Most systems make use of a variety of constraints, such as the epipolar constraint, to reduce the search space thereby improving the matching speed.

In computational stereo, tokens from one image are matched against tokens from another image. These tokens can be intensity levels or features extracted from the raw image. Recently, the problem of simplifying the matching process by use of a third camera has been investigated by a number of researchers[13,15, 10,8,9,12] (for a detailed reference see [14]). We have described elsewhere an innovative method for matching line segments which makes extensive use of the epipolar constraint by utilizing 3, rather than 2, cameras [5,4]. Although this method was fast and reliable, there was still the need to compute the intersection of the epipolar line with candidate segments. If we can align the epipolar lines such that they are parallel to each other and with the axes of the image planes (i.e. horizontal and vertical), we can dramatically reduce the computational effort for the search process. This problem has been addressed by others who have used mechanical means to achieve conjugate epipolar lines through camera placement. These methods have proved cumbersome and unreliable in practice due to the inaccuracy of the mechanical devices. Furthermore, such methods restrict the camera placement. In a previous paper, we formally developed a general rectification method requiring only the knowledge of the perspective transformation matricies of each camera[3]. Here we describe the results of experiments we have conducted with this rectification method and our trinocular stereo system.

In the following sections, we describe further enchancements to our trinocular stereo system. These improvements include a more robust and general control strategy. We have also developed a new method for the reconstruction of 3-D segments which provides better results and a novel validation technique based on the observation that neighbors in the image should be neighbors in space. Lastly, experiments demonstrating these advancements are presented.

2 Overview of Trinocular Stereo

Figure 1 illustrates the geometric constraints of trinocular stereovision. Camera i ($i = 1, 2$ or 3) is represented by its optical center C_i and its image plane $\mathcal{P}_i$. Given a scene point P its image I_i by camera i is given by the intersection of the line PC_i with the plane $\mathcal{P}_i$. This is the classical pinhole model. Points I_1, I_2 et I_3 form a triplet of *homologous* image points.

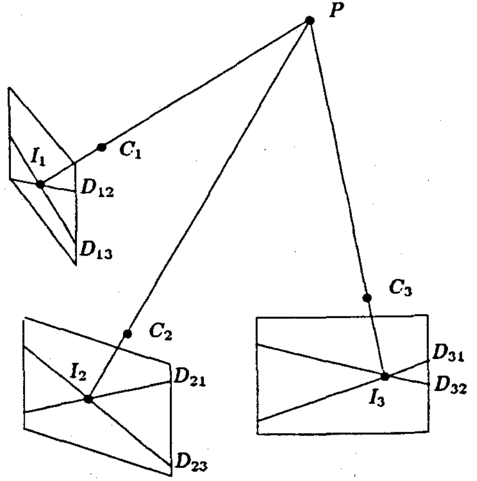

Figure 1: Geometric constraints of trinocular stereovision

Given a pair (i, j) of cameras and a physical point P, the *epipolar plane* $\mathcal{Q}_{ij}$ is defined by the triplet of points (C_i, P, C_j). The intersection of this epipolar plane with camera plane $\mathcal{P}_i$ is the *epipolar line* D_{ij}, while its intersection with camera plane $\mathcal{P}_j$ is the epipolar line D_{ji}. D_{ij} and D_{ji} are called *conjugated epipolar lines*. Any point I_i on D_{ij} (resp. I_j on D_{ji}) has its homologous image point I_j on D_{ji} (resp. I_i on D_{ij}). Therefore, using two

*this work was partially supported by esprit project $P940$.

†current address: Dept. of Computer Science, University of Utah

cameras, the search for homologous image points is a search along conjugated epipolar lines.

As one can see on figure 1, a scene point P produces three pairs of homologous epipolar lines. When the image points (I_i, I_j, I_k) form a triplet of *homologous* image points, then I_i is necessarily located at the intersection of the epipolar lines D_{ij} and D_{ik} respectively defined by I_j and I_k. Therefore the search for homologous image points between two images can now be reduced to a simple verification at a precise location in the third image. For instance checking that (I_1, I_2) form a pair of homologous image points consists in verifying the presence of I_3 at the intersection of D_{31} and D_{32}.

An overview of the calibration used in our system can be found in Appendix A.

2.1 Overview of Previous System

We have previously presented an original trinocular approach to stereovision[5,6]. Our scheme was a four step process (for details see [5]):

1. **Preprocessing** - Acquire an image and extract the features to be matched. In our case, linear segments, which are polygonal approximations to edge pixels, are used.

2. **Hypothesis Prediction** - For each segment in the first image, locate a match, within local geometric constraints, from the candidate segments in the second image. By employing the trinocular geometry described above, use the third image to verify or refute the hypothesis.

3. **Hypothesis Validation** Following the hypothesis prediction step, about 10 percent of the hypothesized triplets are incorrect. The validation phase uses local constraints in the image, the disparity gradient, to validate the matches. After hypothesis validation, less than 1 percent of the matches are erroneous.

4. **3-D reconstruction** Reconstruct the 3-D segment from the common portion of the triplet of matched 2-D segments.

We demonstrated the effectiveness of utilizing the third camera with this approach. Yet, further advancements in terms of efficiency and reliability are possible.

One such modification changes the order of the preceeding steps. We have found through experimentation that if we reconstruct after the hypothesis step, the validation procedure is more effective. This allows correct matches, which previously were considered spurious by the validation criteria, to be properly validated. This is explained in the section on validation.

3 Rectification of the images

Through the epipolar geometry, the search for candidate matches is reduced from a 2 dimensional search to a 1 dimensional search; matches must lie on the epipolar line. Although quite simple, one still needs to compute the epipolar line attached to the midpoint of a segment in the first image and subsequentially, compute the intersections with candidate segments. Even when optimized, this method is computationally expensive, when applied to real data, due to the number of intersections which must be computed. We could enhance the process if we could achieve parallel conjugate epipolar lines. Furthermore if we align these parallel epipolar lines with the image coordinate frame, computation becomes minimal.

Since all epipolar lines pass through the epipole, we can achieve parallel epipolar lines, in the image, by rejecting the epipole to infinity. This can be accomplished by reprojecting the original image onto a new image plane which is parallel to the vector formed by the two optical centers (see figure 2). For three cameras, it suffices to chose a plane parallel to the plane containing the optical centers of the three cameras. Details of this process, as well as the mathematical derivation, are rigorously addressed in another paper[3].

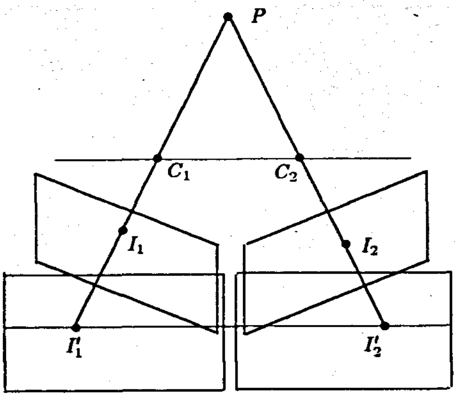

Figure 2: Rectification of two images

Once we have parallel epipolar lines, our next objective is to judiciously chose our new image coordinate frames such that we have the advantageous relationship:

$$\begin{aligned} u_1 &= u_3 \\ v_1 &= v_2 \\ u_2 &= v_3 \end{aligned}$$

Figure 3 illustrates this concept. This is analogous to attaching the UV coordinate frame of the images to the epipolar lines. Since the horizontal and vertical lines are parallel conjugate epipolar lines, this reduces the search to simply looking along the horizontal or vertical for a potential match.

The rectification matricies are given in the appendix. Rectification requires only 6 multiplications, 6 additions and 2 divides per end-point. Figure 4 shows a triplet of images of a typical room scene and Figure 5 shows the rectified triplet.

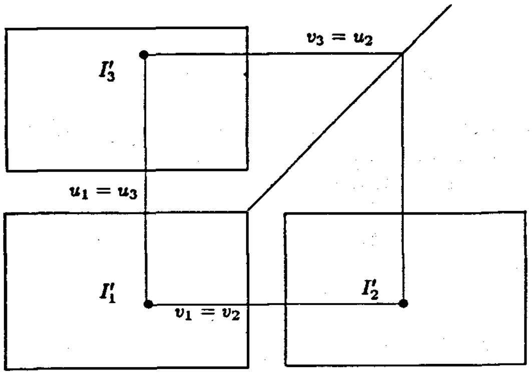

Figure 3: After the rectification of three images

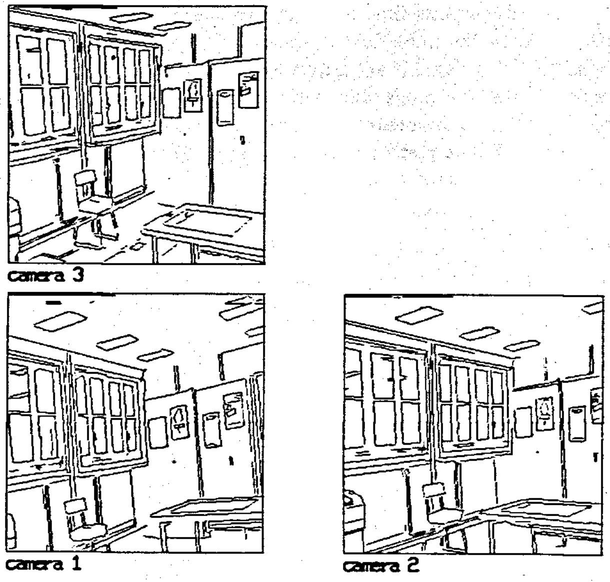

Figure 4: Typical Room Scene

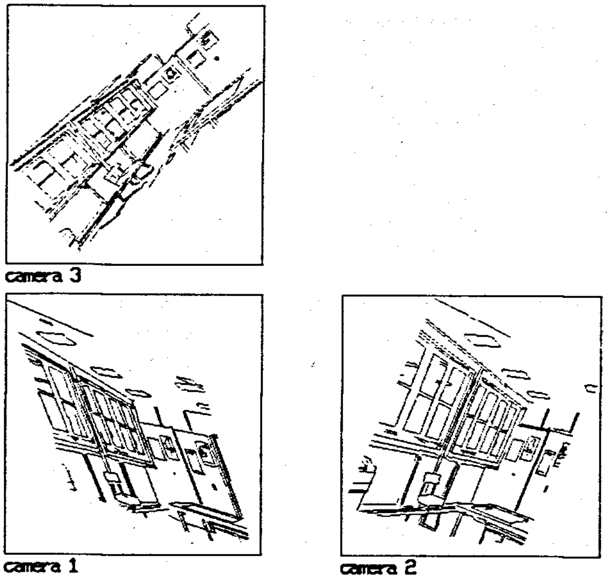

Figure 5: Rectified Room Scene

Rectification drastically reduces the amount of computation necessary in the hypothesis formation step of the stereo process. We have found that the rectification process decreases the matching time by more than a factor of 2. Table 1 shows the results of rectifying the segments prior to matching for several different scenes. Processing times are given for a Sun 3/50 workstation.

4 Reconstruction

Given a hypothesis, that is a matched triplet of segments from the 3 images, we want to reconstruct a 3-D segment in space which corresponds to the matched 2-D segments. In our previous paper, this process was performed by locating the common portion of the 3 segments using the epipolar geometry and then projecting the endpoints to find the physical line corresponding to the imaged segments [5]. This is shown in Figure 6. Actually, this figure is misleading since in practice 3 planes never intersect in a line. In practice, we are in the situation depicted by Figure 7. The 3-D segment we would like to recover is contained in the volume defined by the intersections of the planes. We decompose this problem in the following two subproblems:

1. How to reconstruct 3D lines from their 2D images.
2. How to compute the corresponding 3D endpoints.

	number of segments			*processing times*		
	cam1	*cam2*	*cam3*	*previous matcher*	*rectification*	*new matcher*
scene1	548	531	536	11.4208	0.6968	5.8930
scene2	393	405	371	6.7562	0.4644	3.4694
scene3	203	199	205	1.6766	0.1988	0.7470
scene4	262	240	284	2.5066	0.2818	1.1454
scene5	283	266	280	2.6228	0.2984	1.1122
scene6	312	337	336	3.3864	0.3482	1.5770
scene7	305	352	338	3.2536	0.3814	1.5106

Table 1: Speed up of Rectification

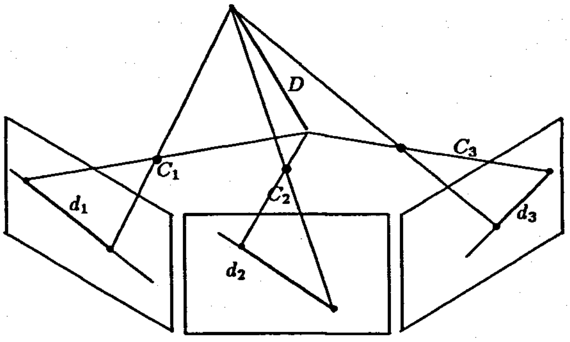

Figure 6: Previous Method

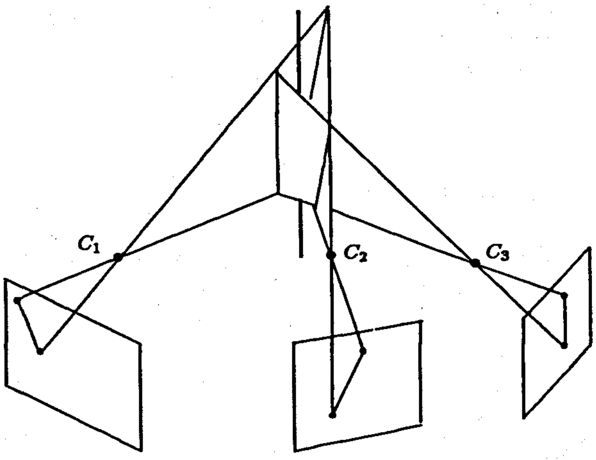

Figure 7: Actual Case of Three Planes

4.1 Building 3D lines from their 2D images

More formally, given three 2D lines d_i, one seeks the 3D line D whose projections d_i' on cameras i ($i = 1,2,3$) *best* approximate the 2D lines d_i (cf. figure 8).

For doing this, one uses minimal representation of lines. Therefore, assuming d_i is not parallel to the v axis, [1] it is represented by the parameters (α_i, μ_i) such that the equation of d_i in the image plane of camera i is

$$\alpha_i u_i + v_i + \mu_i = 0$$

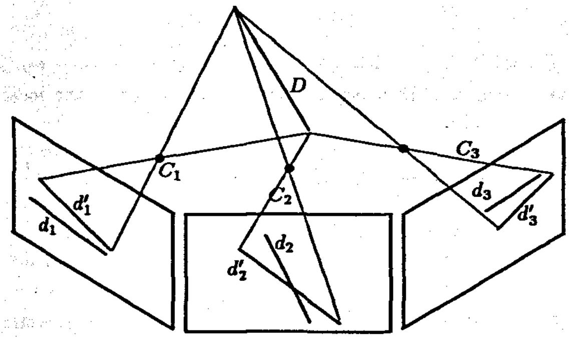

Figure 8: Building 3D lines from their 2D images

Assuming that D is not perpendicular to the z axis, [2] it is represented by the parameters (a,b,p,q) such that D is defined by the equations

$$\begin{cases} x = az + p \\ y = bz + q \end{cases} \tag{1}$$

One assumes that the perspective transformation of each camera is represented by a 3x4 matrix T_i computed during a preliminary calibration stage [7]. If we denote by t^i_{jk} the element of rank (j,k) in the perspective matrix T_i, saying that the projection of D on camera i is d_i is equivalent to saying that the following two equations hold (see appendix):

$$a(\alpha_i t^i_{11} + t^i_{21} + \mu_i t^i_{31}) + b(\alpha_i t^i_{12} + t^i_{22} + \mu_i t^i_{32}) + (\alpha_i t^i_{13} + t^i_{23} + \mu_i t^i_{33}) = 0 \tag{2}$$
$$p(\alpha_i t^i_{11} + t^i_{21} + \mu_i t^i_{31}) + q(\alpha_i t^i_{12} + t^i_{22} + \mu_i t^i_{32}) + (\alpha_i t^i_{14} + t^i_{24} + \mu_i t^i_{34}) = 0 \tag{3}$$

This system provides two independant linear equations on the unknowns (a,b) and (p,q) respectively: therefore two images are enough to solve for (a,b,p,q) exactly. Given three images, the system becomes overconstrained, and one must define an error criterion.

To do so, we consider the uncertainties on the parameters of the 2D lines, and we take them into account explicitely by computing a recursive weighted least square solution (Kalman Filter approach). This approach provides not only a better estimate of (a,b,p,q) (compared to a simpler least-square) but also an estimate of its quality under the form of a 4x4 symetric covariance matrix W_D. The interested reader is referred to [1,11,2].

4.2 Computing 3D endpoints

Having computed the parameters of a supporting 3D line, one must use the endpoints of the 2D image segments to define the endpoints of a 3D segment. For each endpoint I_i of a 2D segment in image i we compute the 3D line L_i supported by C_iI_i and the 3D point P_i of D which is closest to D_i (the common perpendicular).

Therefore, given the two endpoints a_i and b_i of a 2D segment, one obtains the endpoints A_i and B_i of a 3D segment *supported by* D. This is illustrated by figure 9.

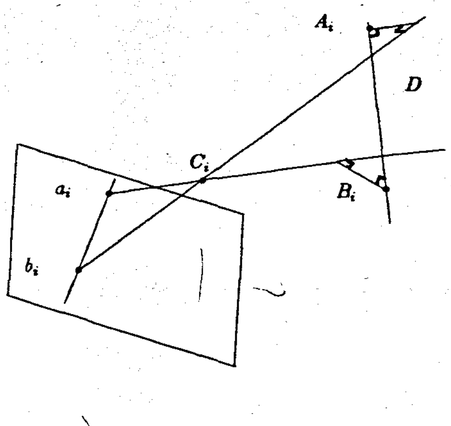

Figure 9: Building 3D segments from 2D segments

This operation is repeated for the endpoints of the corresponding segment in images 2 and 3, and one keeps the 3D segment on D which is the intersection of A_1B_1, A_2B_2 and A_3B_3.

This algorithm for reconstruction gives us better results than our previous method. The reconstructed 3-D segments are in the same position but the lengths of the segments are longer as seen in Figure 10. Notice the improvements with the segments above the window, the desk, and the lights on the ceiling.

5 Validation

After the hypothesis prediction step, about 10 percent of the hypothesized matches are erroneous. This is due to the existence of verifying segments in the third image which fulfill both the geometric and epipolar constraints and is generally caused from artifacts in the scene. We must employ a validation step to filter out these bad hypotheses. To do this, we use the following two constraints:

1. **Uniqueness Constraint** - This allows, at most, one hypothesis for each matched segment. We constrain uniqueness only within epipolar bands so that errors with segmentation do not cause problems.

2. **Regularity Constraint** - If we assume that objects in the scene are smooth, the two segments belonging to the same object which are neighbors in the image will also be reconstructed as neighbors in space (except at a few depth discontinuities).

The uniqueness constraint is quite simple. If a segment matches more than one segment in either of the other two images and the segments overlap within the epipolar band, then this constraint is violated. This is shown for a camera pair in Figure 11. Recall that the epipolar lines are horizontal thus the epipolar bands are delimited by the endpoints of the segments. Although S_1^1 matches both S_2^1 and S_2^2, the match is considered valid since S_2^1 and S_2^2 do not overlap within an epipolar band. Presumably, S_2^1 and S_2^2 belong to a broken edge. This can be caused by errors in the preprocessing step. Whereas the match S_1^2 with S_2^3 and S_2^4 violates

[1] one uses the symmetric parametrization for lines parallel to the v axis

[2] one uses two complementary parametrization respectively for lines perpendicular to the zx or zy planes.

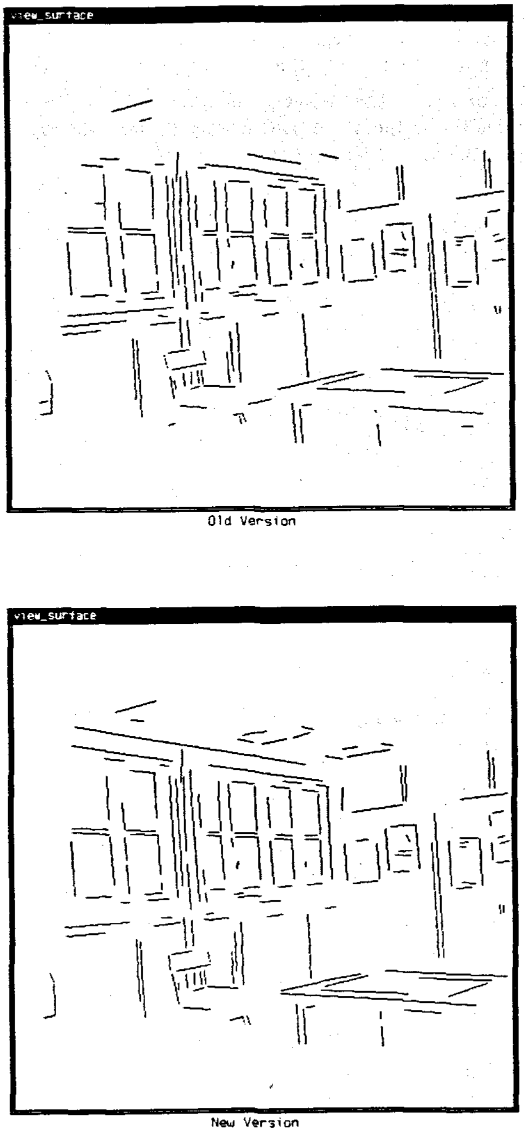

Figure 10: Results of old and new

the uniqueness constraint since they do overlap within the epipolar band.

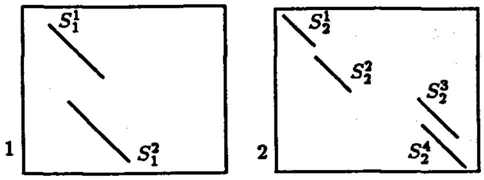

Figure 11: Uniqueness Constraint

The regularity constraint we now use differs substantially from the previous version. In our previous system, we computed the local neighborhood graph for the matched segments and used the disparity gradient measure to discriminate between good matches and erroneous ones. Since the disparity gradient measure was only computed for the mid-point of a segment, this method sometimes discarded good matches and occasionally allowed bad matches to pass the validation test.

We have developed a new method based on the assumption that neighboring segments in the image are likely to be neighboring segments in 3-space. We define the following acceptance criterion, Ψ:

$$\Psi = \frac{\sum \xi}{N} \tag{4}$$

where:

$$\xi_i = \left\| \begin{array}{ll} 1 & \text{if } i \text{ is a } \textit{supporting} \text{ neighbor} \\ 0 & \text{otherwise} \end{array} \right. \tag{5}$$

$$N = \text{total number of neighbors} \tag{6}$$

A neighbor is any segment which is physically close in the image (within a local 2-D neighborhood). We compute the local 2-D neighborhood as before[4]. A *supporting neighbor* is a neighbor in the image whose reconstructed segment lies close to the segment under consideration. Thus, given a hypothesis, a matched triplet of 2-D segments and their reconstructed 3-D segment S, compute the ratio of the matched 2-D neighbors (image neighbors) whose reconstructed 3-D segment is sufficiently close to S and the total number of neighbors.

Since we have already reconstructed all neighbors, it is straightforward to compute the distance between 3-D segments although it is not computationally efficient to do so. To determine whether a neighboring segment is *supporting* or not, we use a rectilinear parallelpiped containing S as a clipping box to rapidly determine 3-D neighbors. The box is constructed, in 3-space, at an experimentally determined distance from the segment, in our case, 50 cm. It suffices to compute the intersections of the 3-D segments in question with this box. Using this criterion, much less than 1 percent of our final matches are incorrect.

6 Experiments and Results

This stereo matching technique has been tested on a number of indoor scenes. We only present the following typical results.

Three images of a room are taken simultaneously with our previously calibrated three camera system mounted on our mobile robot. These 3 scenes were taken by rotating the mobile robot. A triplet is digitized, and from these 512x512 images, edge points are extracted and chains of connected edge points are built and approximated by a set of linear segments, oriented with respect to the contrast sign across the segment. These images are then rectified for processing efficiency. At this point, the preprocessing is complete. Hypotheses are generated. For each hypothesis, a 3-D segment is constructed. Using these 3-D segments, we can validate the matches to eliminate spurious hypotheses. Figure 12 shows the original segments used for matching. Figure 13 shows the rectified segments. Figure 14 show the results of matching and validation.

7 Conclusion

In this paper, we have presented the recent advancements in our trinocular stereo system. These included a technique for calibrating and rectifying in a very *efficient* and *simple* manner the triplets of images taken for trinocular stereovision systems. After the rectification of images, epipolar lines are parallel to the axes of the image coordinate frames: therefore, potential matches between two or three images satisfy simpler relations, allowing for a simpler and more efficient matching algorithm.

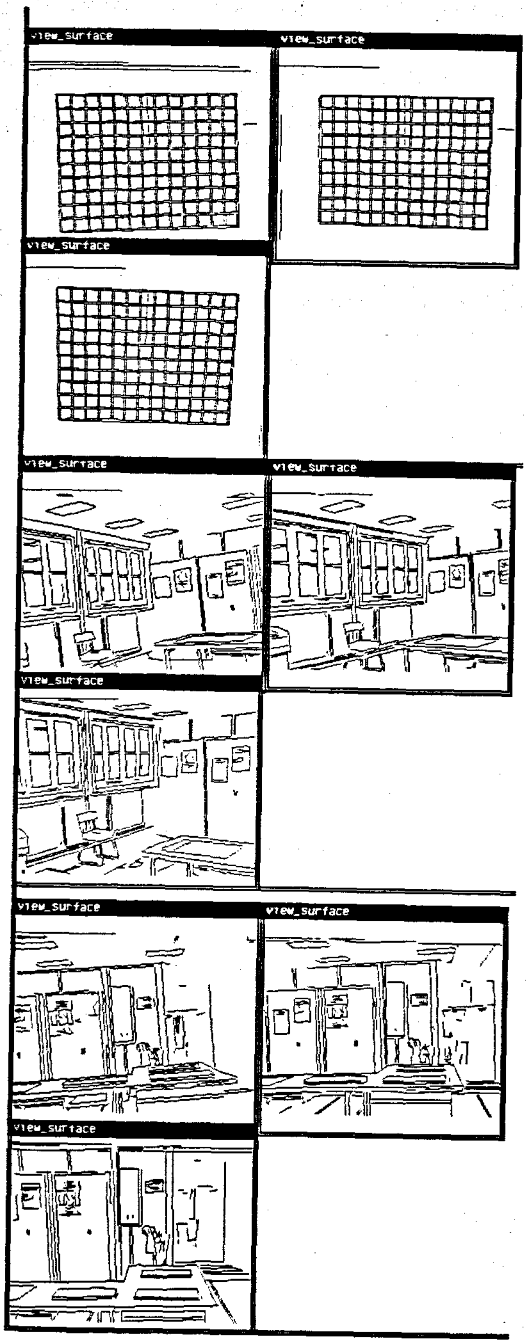

Figure 12: Segments Used for Matching

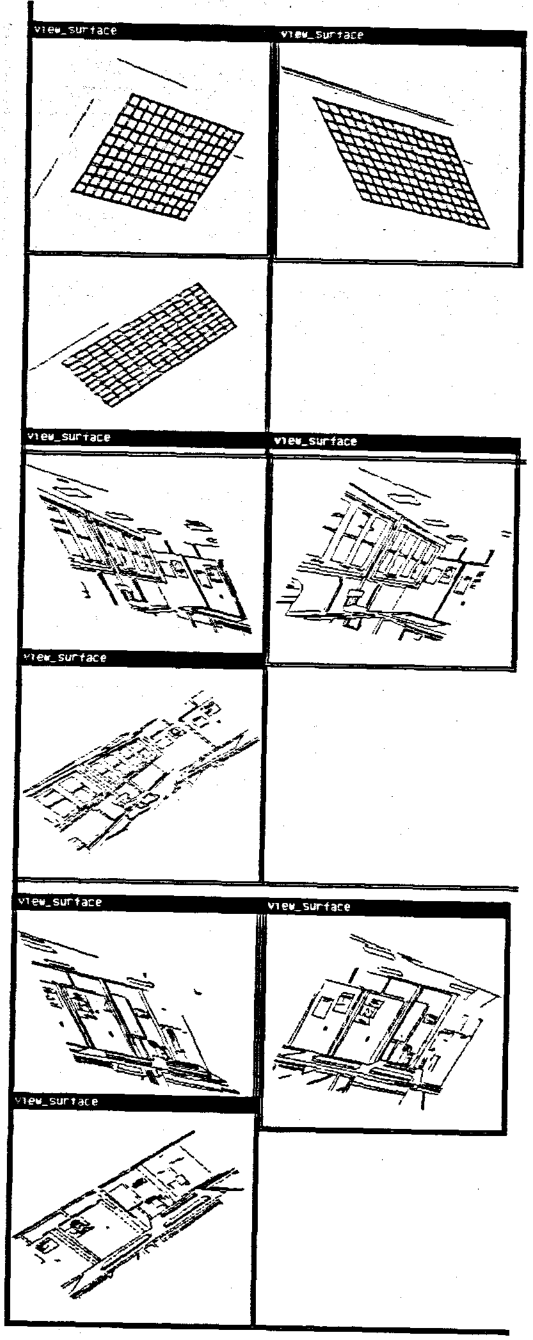

Figure 13: After Rectification

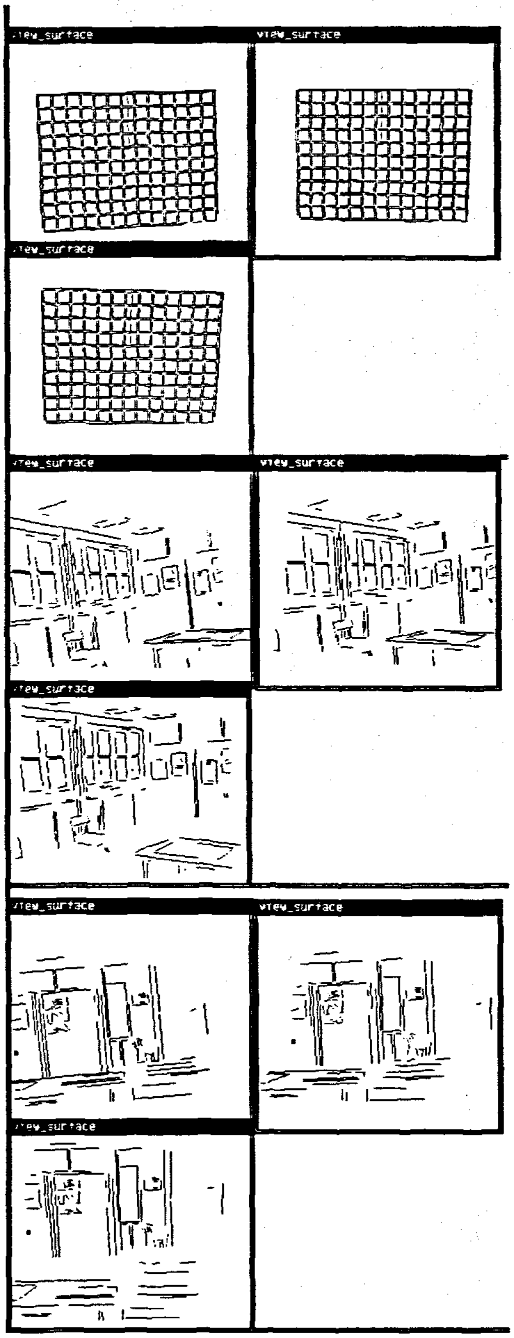

Figure 14: Results of Trinocular Stereo

We also described a more robust and general control strategy now employed in our system. We have shown an improved method for reconstruction of the 3-D segments which provides better results than our previous system. We have also shown a new validation technique based on the observation that neighbors in the image should be neighbors in space. Results from experiments demonstrating these ideas have been presented.

A Computing a 3D line D from its 2D projections d_i

One assumes that the perspective transformation of each camera is represented by a 3x4 matrix T_i computed during a preliminary calibration stage [7]. T_i is used to relate the projective coordinates $(x, y, z, 1)^t$ of a 3D point P to the projective coordinates $I_i^* = (U_i, V_i, S_i)$ of its image in camera i by:

$$I_i^* = \begin{pmatrix} U_i \\ V_i \\ S_i \end{pmatrix} = T_i \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

If P is not in the focal plane of camera i, then the image coordinates of I_i are given by:

$$I_i = \begin{pmatrix} u_i \\ v_i \end{pmatrix} = \begin{pmatrix} U_i/S_i \\ V_i/S_i \end{pmatrix}$$

If C_i is the optical center for camera i, then three 3x3 rectification matrices, called R_1, R_2 and R_3, are defined as (for details see[3]):

$$R_i = \begin{pmatrix} (C_{i-1} \times C_i)^t \\ (C_i \times C_{i+1})^t \\ (C_1 \times C_2 + C_2 \times C_3 + C_3 \times C_1)^t \end{pmatrix} [t_2^i \times t_3^i \quad t_3^i \times t_1^i \quad t_1^i \times t_2^i]$$

where $i+1 = 1$ if $i = 3$ and $i-1 = 3$ if $i = 1$.

The image of a generic point $P = (x, y, z)^t$ of D by camera i is $I_i' = (u_i', v_i')^t$ such that:

$$\begin{aligned} u_i' &= \frac{(at_{11}^i + bt_{12}^i + t_{13}^i)z + pt_{11}^i + qt_{12}^i + t_{14}^i}{(at_{31}^i + bt_{32}^i + t_{33}^i)z + pt_{31}^i + qt_{32}^i + t_{34}^i} \\ v_i' &= \frac{(at_{21}^i + bt_{22}^i + t_{23}^i)z + pt_{21}^i + qt_{22}^i + t_{24}^i}{(at_{31}^i + bt_{32}^i + t_{33}^i)z + pt_{31}^i + qt_{32}^i + t_{34}^i} \end{aligned}$$

where t_{jk}^i is the element of rank (j, k) in the perspective matrix T_i.

Saying that I_i' belongs to d_i means that

$$\alpha_i u_i' + v_i' + \mu_i = 0$$

If the preceding relation has to be verified for any $P \in D$, except C_i, then the following two equations must hold:

$$\begin{aligned} \alpha_i(at_{11}^i + bt_{12}^i + t_{13}^i) + (at_{21}^i + bt_{22}^i + t_{23}^i) + \mu_i(at_{31}^i + bt_{32}^i + t_{33}^i) &= 0 \\ \alpha_i(pt_{11}^i + qt_{12}^i + t_{14}^i) + (pt_{21}^i + qt_{22}^i + t_{24}^i) + \mu_i(pt_{31}^i + qt_{32}^i + t_{34}^i) &= 0 \end{aligned}$$

By reorganizing the coefficients, one can see that these equations are the equations 2 and 3.

References

[1] N. Ayache. *Construction et Fusion Représentations Visuelles 3D: Applications á la Robotique Mobile.* PhD thesis, Paris-Orsay, 1988.

[2] N. Ayache and O.D. Faugeras. Maintaining Representations of the Environment of a Mobile Robot. In *International Symposium on Robotics Research*, August 1987. Santa-Cruz, California.

[3] N. Ayache and C. Hansen. Rectification of images for binocular and trinocular stereovision. In *Proc. International Conference on Pattern Recognition*, October 1988. 9th, Beijing, China.

[4] N. Ayache and F. Lustman. Fast and Reliable Passive Stereovision Using Three Cameras. In *Proceedings of International Workshop of Industrial Applications of Machine Vision and Machine Intelligence*, IEEE, February 1987. Tokyo, Japan.

[5] N. Ayache and F. Lustman. Fast and Reliable Passive Trinocular Stereovision. In *Proc. First International Conference on Computer Vision*, pages 422–427, IEEE, June 1987. London, U.K.

[6] N. Ayache and F. Lustman. Trinocular Stereovision, Recent Results. In *Proc. International Joint Conference on Artificial Intelligence*, August 1987. Milano, Italy.

[7] O.D. Faugeras and G. Toscani. The Calibration Problem for Stereo. In *Proceedings CVPR '86, Miami Beach, Florida*, pages 15–20, IEEE, 1986.

[8] A. Gerhard, H. Platzer, J. Steurer, and R. Lenz. Depth Extraction by Stereo Triples and a Fast Correspondence Estimation Algorithm. In *Proc. International Conference on Pattern Recognition*, pages 512–515, IEEE, October 1986. Paris, France.

[9] E. Gurewitz, I. Dinstein, and S. Sarusi. More on the Benefit of a Third Eye for Machine Stereo Perception. In *Proc. International Conference on Pattern Recognition*, pages 966–968, IEEE, October 1986. Paris, France.

[10] M. Ito and A. Ishii. Range and Shape Measurements Using Three-View Stereo Analysis. In *Proc. International Conference on Computer Vision and Pattern Recognition*, pages 9–14, IEEE, 1986. Miami Beach, Florida.

[11] F. Lustman. *Vision stéréoscopique et perception du mouvement en vision artificielle.* PhD thesis, Paris-Orsay, 1987.

[12] Y. Ohta, M. Watanabe, and K. Ikeda. Improving Depth Map by Right-angled Trinocular Stereo. In *Proc. International Conference on Pattern Recognition*, pages 519–521, IEEE, October 1986. Paris, France.

[13] M. Pietikainen and D. Harwood. Depth from Three-Camera Stereo. In *Proc. International Conference on Computer Vision and Pattern Recognition*, pages 2–8, IEEE, 1986. Miami Beach, Florida.

[14] M. Pietikainen and D. Harwood. Progress in trinocular stereo. In *Proceedings NATO Advanced Workshop on Real-time Object and Environment Measurement and classification, Maratea, Italy*, August 31 - September 3 1987.

[15] M. Yachida, Y. Kitamura, and M. Kimachi. Trinocular Vision: New Approach for Correspondence Problem. In *Proc. International Conference on Pattern Recognition*, pages 1041–1044, IEEE, October 1986. Paris, France.