Multi-baseline Stereo Using Surface Extraction

C. Lawrence Zitnick, Jon A. Webb

November 24, 1996
CMU-CS-96-196

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213-3890

Abstract

Traditionally, the matching problem in stereo vision has been formulated as an ill-posed problem. However, it has been shown that the matching problem can be well-posed in scenes with no occlusions or depth discontinuities. Unfortunately most real scenes do not obey these constraints. We overcome this by finding regions within the images in which the matching problem is well-posed. That is, we find image regions in which there are no occlusions or depth discontinuities. In general, a unique set of such regions does not exist. However, we will demonstrate that in almost all cases these regions can be found efficiently. Therefore the matching problem can be well-posed in almost all cases.

In order to find these corresponding regions we transform the problem from finding 2D image regions into identifying 3D surfaces. We have developed a method of 3D surface extraction which uniquely identifies correct 3D surfaces from a set of potential surfaces. In order to test the method we have built a four camera system with which we will present results from several scenes.

This research was partially supported by Visual Interface, Inc., and partially by the Advanced Research Projects Agency of the Department of Defense under contract number F19628-93-C-0171, ARPA order number A655, "High Performance Computing Graphics," monitored by Hanscom Air Force Base.

Keywords: Early vision, Stereo vision, Multi-baseline, Well-posed problems

1. Introduction

It is well known that computer graphics, the construction of 2D images from 3D models, is a well-posed problem: a unique solution does exist, and the solution depends continuously on the data. Unfortunately the same is not always true of the inverse problem of constructing a 3D model from 2D images[20]. Such a method is stereo matching which has typically been formulated as an ill-posed problem unless highly restrictive constraints[5,20] are imposed on the final 3D model. Such constraints typically include not allowing occlusions or discontinuities. In areas of occlusion there exists no solution to the matching problem, thus in general it is impossible for the matching problem to be well-posed. However, in areas which do not contain occlusions a solution does exist, allowing for the possibility of the problem to be well-posed over these regions of the image. It has been shown if a region of the image could be identified as containing no occlusions or discontinuities and the corresponding region of another image was identified, the matching problem between these regions is well-posed [5]. Unfortunately, in general there is no unique solution to identifying these corresponding regions, which once again creates an ill-posed problem. However, as we will demonstrate, there is a unique solution which can be efficiently found in most cases; making the stereo matching problem well-posed over regions in the images.

Our method works as follows: Assuming known epipolar geometry we can reduce the matching problem from a 2D to a 1D problem. In real images there are still typically many possible matches to each pixel in the reference camera. Now suppose we project every match into 3D space. Upon inspection of the set of 3D points we will find a collection of potential 3D surfaces. Since the correct surfaces within the set of potential surfaces are continuous, the 2D area in which they project on the images must not contain any occlusions or discontinuities. Thus, finding corresponding regions in images which contain no occlusions or discontinuities is equivalent to finding the correct surfaces from the set of potential surfaces.

Finding the correct potential surface is a surprisingly simple process. When surfaces within the scene are not occluded, the correct surface is always the potential surface with highest population, i.e. the highest number of 3D points, due to a simple (and non-heuristic) geometric constraint. Furthermore, even when a surface is partially occluded, if the distance between the surface and the surface occluding it is less than some distance $\varepsilon/2$, then the correct surface will still obey this constraint. As we shall demonstrate, ε is dependent on the camera baseline distance and repetition frequency of the texture in the scene. By proper stereo camera design ε can be made arbitrarily large.

In order to demonstrate our method with real systems, we built a system with four cameras along a similar baseline. Using four cameras instead of two, greatly improved the accuracy of the system while also increasing the reliability. By using a smaller baseline between two cameras we can increase ε, while using cameras with larger baseline distances increases the accuracy. Furthermore, since more cameras are used a greater number of false matches can be eliminated, thus eliminating several potential surfaces. Surprisingly, due to the elimination of false matches, increasing the number of cameras can actually decrease the algorithm's running time.

The main contribution of this paper is the method for uniquely finding corresponding regions between images in which the disparity is defined and continuous. Within these regions the stereo matching problem is well-posed as we will discuss in the next section. We will then discuss the relation between the image regions and the potential surfaces, along with how the correct surfaces are extracted from the set of potential surfaces. A description of the actual algorithm will follow which includes: how to extract multiple surfaces, how to create the initial set of 3D points or matches and how potential surfaces are created. Finally we will present results from three complex scenes, and accuracy measurements from two scenes with known geometry.

2. The Well-posed Problem within Regions of Continuous Disparity

Given a image region in which the disparity is continuous, i.e. there exists no occlusion or discontinuities, it is possible to formulate the matching problem as well-posed. We will assume the epipolar geometry between the cameras is known; for excellent reviews of epipolar geometry and stereo vision consult [1, chapter 13], [7, chapter 6], and [8, chapter 7]. The matching problem is then simplified into a 1D problem. Let R(x), and L(x) be the intensity values along the epipolar line in the right and left images respectively. Define the disparity as d(x). The matching problem can then be defined as the minimization of:

$$\|R(x) - L(x + d(x))\| \qquad (1)$$

Unfortunately in order for the solution to (1) to be unique R(x) and L(x) must be strictly monotonic. In real images this is rarely the case. Therefore we must apply another constraint. As [5, p. 47] proposes we can use a constraint of the Tikhonov type to obtain:

$$\|R(x) - L(x + d(x))\| + \lambda \|d'(x)\| \qquad (2)$$

Thus if $d'(x)$ is to exist the disparity must be continuous. As we stated earlier this implies the matching problem is only well-posed on regions with no occlusion or depth discontinuities. Therefore our task is now to find corresponding regions within the images which possess these two properties.

Formulating the matching problem in this manner is opposite most other methods. In other stereo methods using either two or more cameras, it is assumed the problem is ill-posed. Thus in order to find the correct match, more complicated methods of local correspondence are developed. A typical approach is to use SSD[15, 19], i.e. finding the minimum of the sum of squared distances around a pixel. While, using windows as in SSD can increase the chance of finding the correct match, it also creates problems such as jagged edges in the depth map[4,15,16]. Further attempts at local correspondence have also been attempted using adaptive window sizes[10,18], matching in the Fourier domain[9,12,25] and coarse to fine methods[3,6,13]. One of the most promising methods has been using multiple cameras, or multi-baseline stereo[14,15,19,21-23]. Using multiple cameras can reduce the number of false matches without increasing the complexity of the method for matching. In all of the above methods, it is assumed that matches are not unique, making the matching problem ill-posed. In our method we can uniquely find corresponding regions within the images in which the matching problem is well-posed - thus complicated local correspondence methods are not needed.

3. Extracting the Correct Surfaces from the Set of Potential Surfaces

To find corresponding regions with continuous disparity we first transform the problem from 2D images to 3D space. The forward method to find these regions between images would require us to pick two regions, l and r, within the left and right images respectively (figure 1.) We could then solve the matching problem to find a surface using (2). In order to decide if these two regions meet our requirements we would then have to examine the resulting surface; however we conjecture this step and thus the forward method would be quite difficult.

Instead we solve the inverse problem. First we project all matches into 3D space creating a set of 3D points. We then examine the set of 3D points to find subsets which would create continuous surfaces and minimize (2). These subsets of the 3D point set are called potential surfaces. The projection of the potential surfaces on the images creates regions similar to l and r. Therefore our task is transformed into finding the correct surfaces among the potential surfaces.

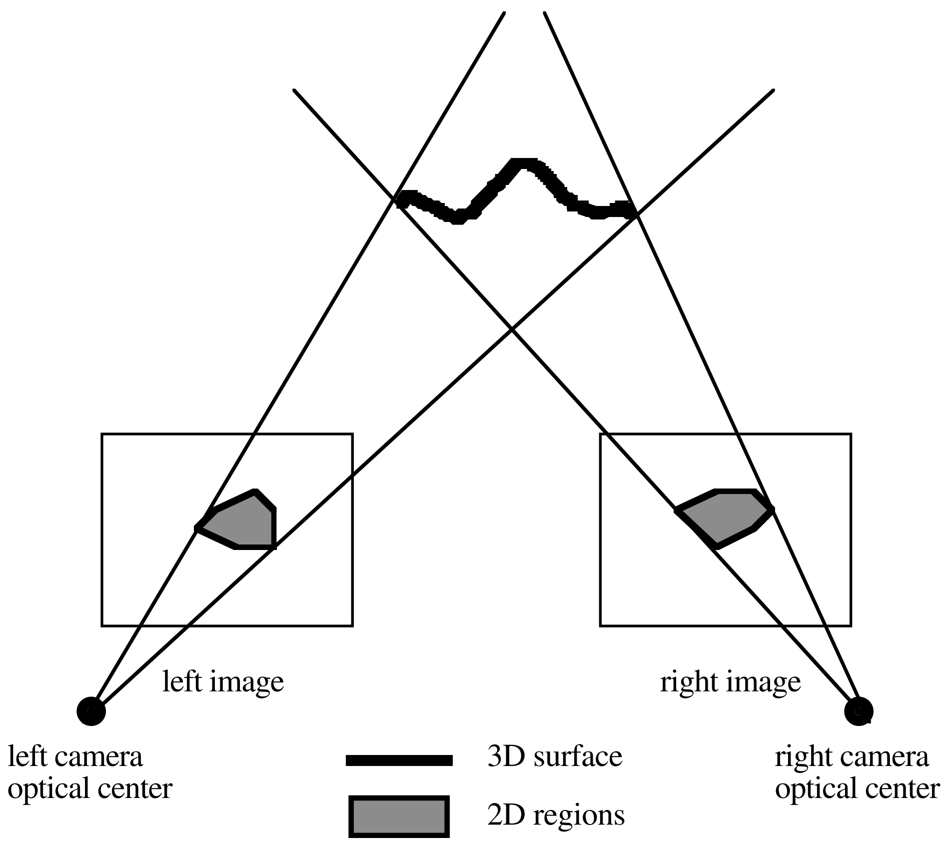


Figure 1: The projection of two corresponding image regions into 3D space

Given a value x, (1) will possess many possible solutions. Each of these solutions can be viewed as belonging to a different set of corresponding image regions which may overlap in one or both images. Analogously, in the 3D case, each solution will lie on a different potential surface. To better illustrate this point consider figure 2. In order to construct the potential surfaces from the matches we impose a constraint as in (2). First we assume that a particular match M is correct, we then combine all the neighboring matches which minimize (2) with respect to M into the same potential surface. Using this method, there exists seven potential surfaces in figure 2 of which only one can be correct, assuming the objects within the scene are opaque. Since the disparity in (2) is continuous and matches are unique within potential surfaces, the potential surfaces must internally obey the left-to-right constraint, i.e. matches must maintain there order between images. Thus only a true surface will entirely project onto a non-occluded object within an image. Therefore, in order to find the correct surface of a non-occluded object, we simply pick the potential surface with the greatest population. In figure 2 this would be the middle surface. Finally as illustrated by figure 2 notice each potential surface is separated by a certain distance ε.

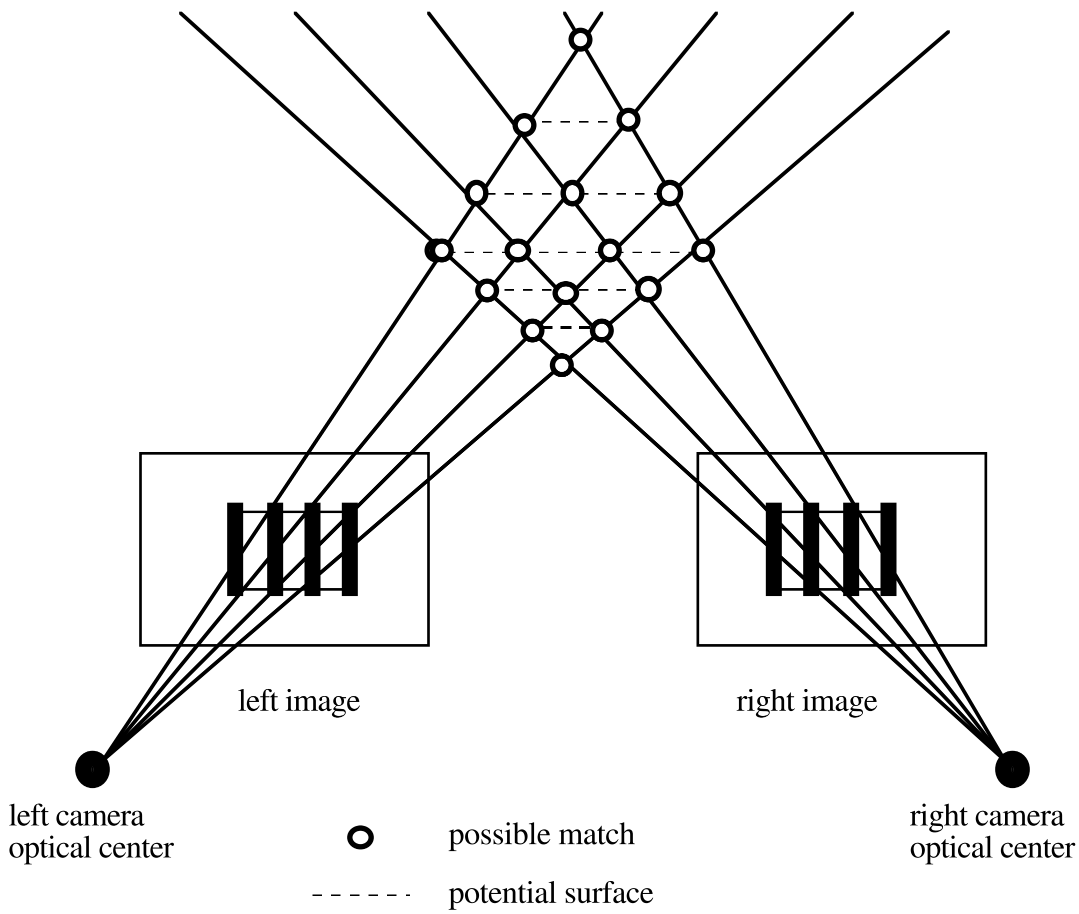


Figure 2: The projection of two images containing a repetitively textured object into 3D space. Notice the potential surface with the highest number of matches is the correct surface.

Now, let us consider the case of partially occluded objects. Unlike non-occluded objects the entire object is not visible from all cameras. Thus, the true surface has portions missing as seen from the reference camera. Therefore as shown in figure 3 it is possible that a false surface could contain more points or matches than the correct surface. However, it is very unlikely this case will occur for two reasons: If the distance between the occluded object and the occluding object is less than ε/2 it is impossible for this case to occur. Typically ε is fairly large, especially when smaller baseline distances between cameras are used. Second, in order for the false surface to contain more points than the true surface, the occluded object must have a highly repetitive texture. If a highly repetitive texture is not present, there will be fewer false matches and false surfaces will occur at greater depth intervals.

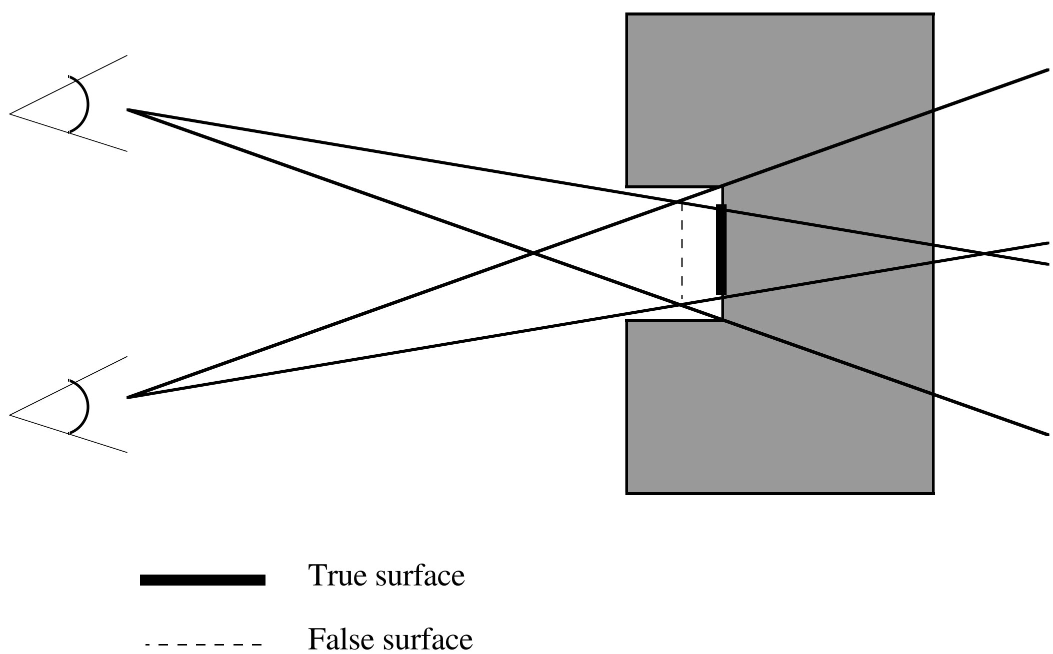
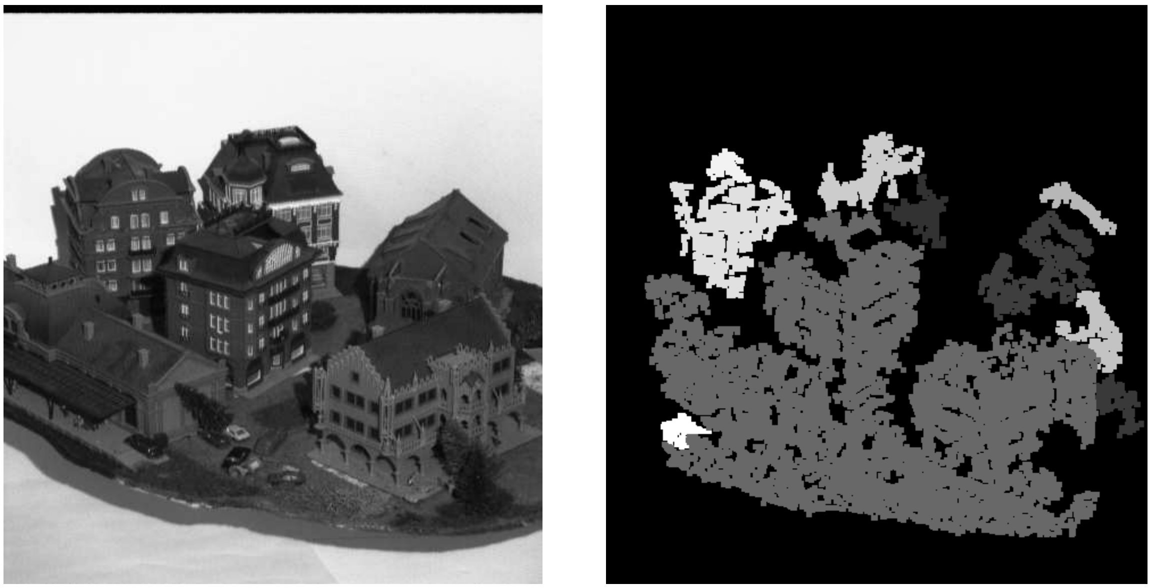


Figure 3: A true surface which is occluded can appear to be smaller than a false surface.

4. Finding Multiple Correct Surfaces From a Set of Potential Surfaces

Since correct surfaces are almost always the largest visible surface as seen from the reference camera, we can easily find one correct surface. We can simply count the number of points in each potential surface and pick the one with the most points as a correct surface. Picking the second surface becomes more complicated. After we pick the first surface we remove it from the set of potential surfaces, but all of the false surfaces corresponding to it are still included in the set. Hence, it is very likely that one of these false surfaces will be picked next if only the population of the potential surfaces are considered. Thus, we make an assumption about the objects within the scene: all objects are opaque. Therefore, after a potential surface is picked as a correct surface all of its points are projected onto the reference camera. Then in order for a future surface to be

correct it must not overlap any previously found correct surfaces as seen from the reference camera. This process of picking the next largest potential surface and checking if it overlaps previously found correct surfaces is repeated until only potential surfaces of some fixed size are left.
An example of picked surfaces is illustrated by figure 4. Notice that every pixel within a neighborhood of a taken pixel was also declared taken by the same surface. This helps guarantee that two surfaces will not overlap.



(a) (b)

Figure 4: Example of picked potential surfaces. (a) The reference image. (b) Surfaces picked, each color represents a different surface.

As stated earlier we continue picking surfaces until only surfaces of some fixed size are left. The likelihood of picking a false surface becomes greater as the surface size decreases. If there is low texture or objects are highly non-lambertain it is unlikely that many points or matches will be found. Thus, objects with low texture are generally broken up into many small surfaces. As a consequence our assumption that the potential surface with the most points is a true surface is no longer valid. Therefore all surfaces below a certain size, generally between 50 and 100 points depending on the scene are not picked.

5. The Set of Three-dimensional Points

Using only two cameras our method first seeks to find a collection of points in three-dimensional space such that all true (actual surface) points are included, but a number of false (false match) points may also be included—in fact, they form the great majority of points resulting from this step. Using the remaining cameras, we then refine the accuracy of the true points, and eliminate some of the false points. Finally in the second and third step we group points into potential surfaces and separate the false points from the true.

5.1 Properties of the set of points

In order to ensure that the second and third step can properly extract correct surfaces, the set of points resulting from the first step must possess several properties:

1. High accuracy for correct points
2. Minimum number of false points
3. Each point is the result of a weighted average of unique matches from each camera pair.
4. Only pixels with a local intensity slope greater than the greatest ratio of baseline distances are considered for matches.

As we shall see later, property 3 places constraints on properties 1 and 2, but property 3 is necessary to guarantee that true points are separated from false points by some fixed distance. Similarly property 4 is needed to ensure proper separation between correct and false points.

5.2 Camera Setup

We will assume that all cameras c_i for i from 0 to n are along a similar baseline. We will designate c_0 to be the reference camera. The cameras c_i for i from 1 to n will be ordered by their distance from the reference camera, i.e. c_1 is the closest camera to c_0(figure 5). Each camera pair consisting of cameras c_0 and c_i, will be designated as C_i. Finally, every camera c_i has a 4x3 perspective transform matrix T_i [7, chapter 3, 7] which relates three-dimensional points to homogenous image coordinates, i.e. we assume the pin-hole model for cameras.

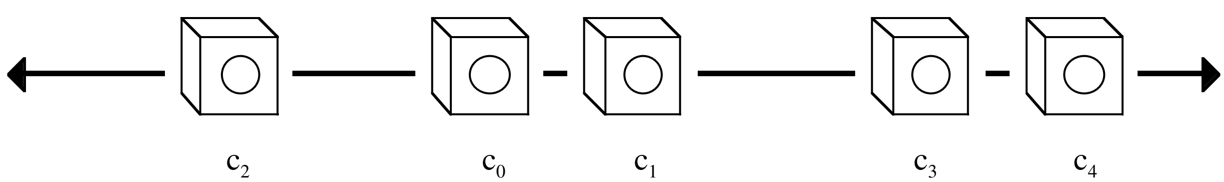


Figure 5: Cameras along a similar baseline.

5.3 Constructing a Set of 3D Points

Using the camera pair C_1 with the smallest baseline, the first step works as follows: Under perspective projection, any possible match to a pixel P in the reference image will lie on the corresponding epipolar line within c_1. When we examine the epipolar line more closely, as shown in Figure 6, we find that due to the discrete nature of images, its values must be calculated by interpolation. If we use a simple and effective technique like bilinear interpolation the values calculated along it look like the series of linear ramps shown in Figure 7.

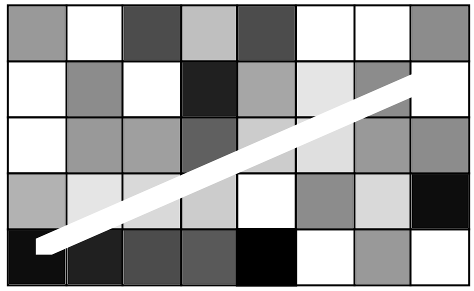


Figure 6: Magnified view of intensity image, and epipolar line.

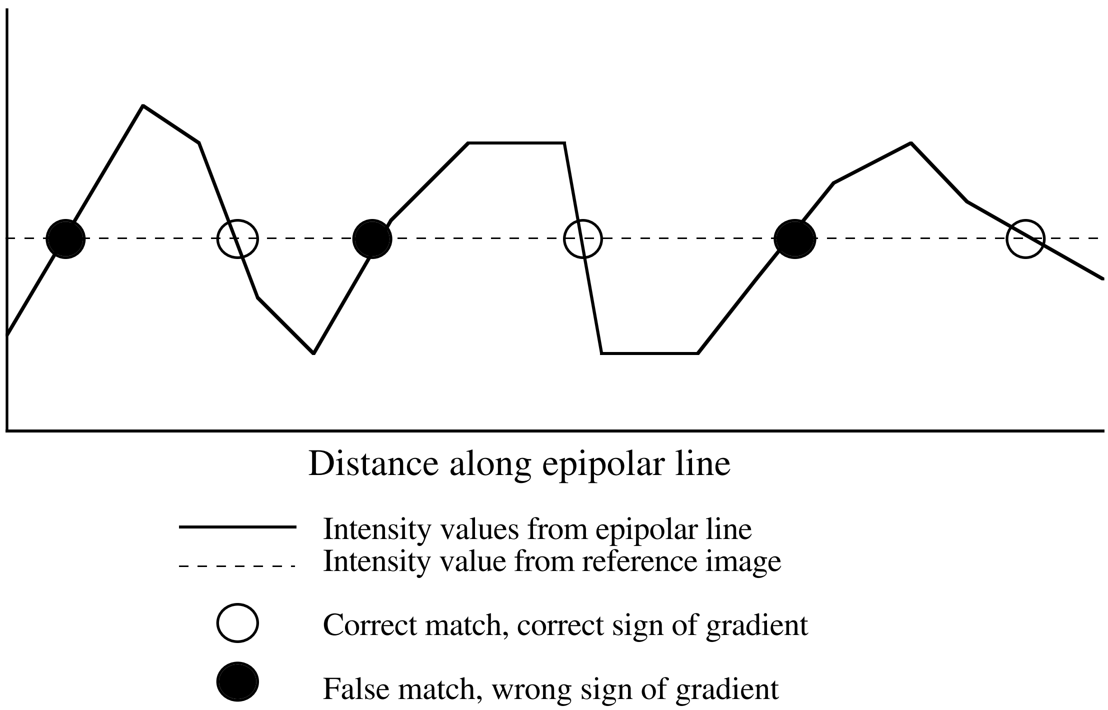


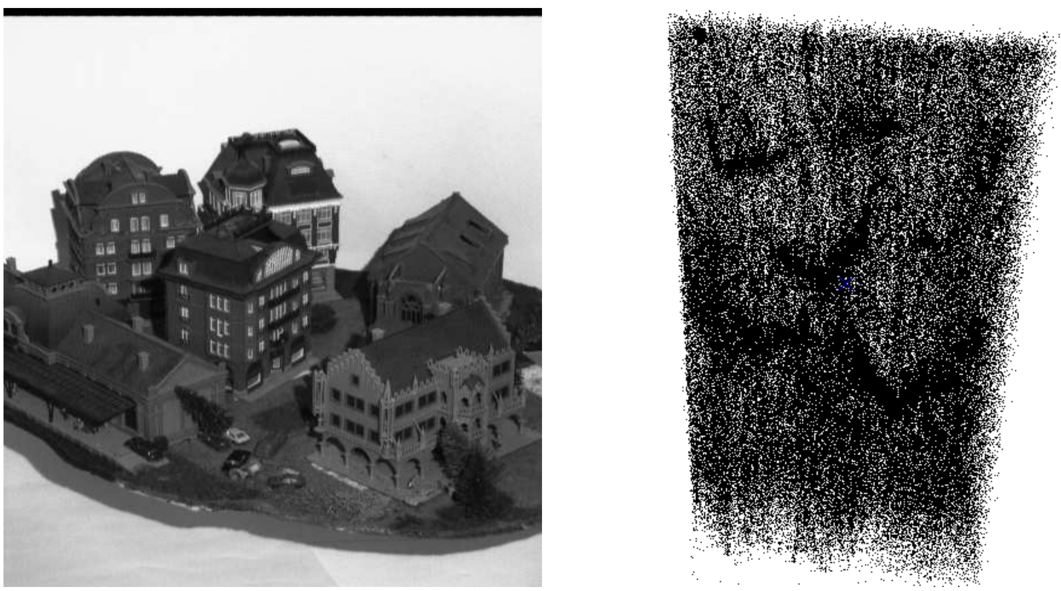
Figure 7: Interpolated values of the epipolar line L, with corresponding possible matches to pixel P in the reference image.

In order to find possible matches simply find the intersection points of the linear ramps shown in Figure 7 with the constant line at the pixel value for P from the reference image. We can eliminate half of these intersection points by including the sign of the gradient at P when we do the intersection. The result is a collection of possible matches for P that includes both its true match and a number of false matches. This process can be done in time proportional to the number of pixels in the epipolar line. The time for this step is on the order of $C*N^2*T$, where C is the number of cameras included in the local set, N is the image resolution, and T is the number of pixels in the epipolar line.

Note that, in common with every other stereo vision technique of which we are aware, we are assuming that all corresponding pixels possess the same appearance in all cameras. This is the Lambertian objects assumption, essentially assuming that all world surfaces are microscopically rough, like flat wall paint.

When using a camera pair with a short baseline distance, correct matches are rarely eliminated, since images from nearby cameras are so similar. Furthermore, the number of false matches is reduced since the epipolar line is foreshortened with the camera pair of shortest baseline distance. Unfortunately, smaller baseline distances cause inaccurate depth estimates.

Sample output from using one camera is illustrated in figure 8. Over 222,000 points were found using a camera pair approximately 1.5m from the model town with a baseline distance of 72.2mm.



(a) (b)

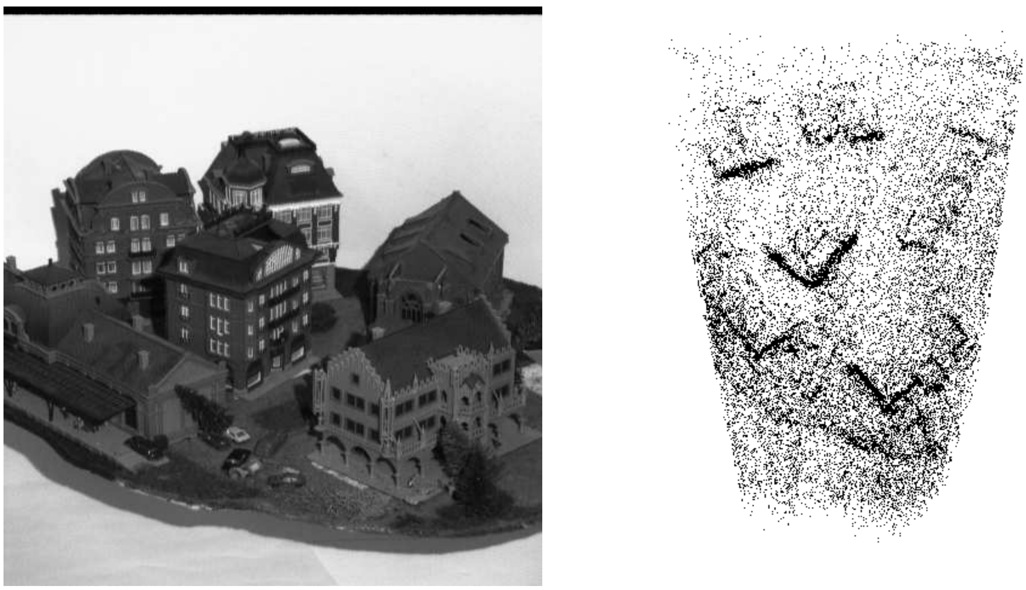
Figure 8: Example of point set after first step using the camera pair with the shortest baseline. (a) The intensity image (486x528) from the reference camera. (b) The resulting point set viewed from above.

The next step iterates among the remaining camera pairs in order of baseline distance to refine the depth estimates and remove many false points. Since we know the approximate depth from the first step we can limit our search for a new match in another camera pair, C_i, to a window around the previous match with radius $E_{i-1}+E_i$, the sum of the error values for the camera pairs. If a match is found within this window a new depth estimate is created by averaging the depth estimates together using baseline lengths as weights. This ensures that estimates with larger baselines and thus higher accuracy are weighted more. With the refined depth estimate we repeat the process using the next camera pair until all camera pairs are used. If a match is not found within the window we conclude the point found using previous camera pairs is a false point and remove it from the point set.

The time for this step is on the order of $D*N^2*U$, where D is the number of cameras in the distant set, N is the image resolution, and U is the number of possible matches for a pixel from the first step.

The resulting point set should possess the first two properties stated earlier. Using camera pairs with larger baseline distances increases the accuracy of the good points satisfying property 1, while using smaller baselines eliminates many false points property 2. Unfortunately, there is a limit to the size ratio of large baselines to small resulting from property 3. In order to better understand this we will discuss the error associated with each camera pair.

An example of the point set after all cameras are used is illustrated by figure 9. The number of points has been reduced to 34,000 with the correct points still members of the set.



(a) (b)

Figure 9: Example of point set after step two using all camera pairs. (a) The intensity image (486x528) from the reference camera. (b) The resulting point set viewed from above.

5.4 Estimation of Accuracy

Given a three dimensional point p, with corresponding point P_0 in c_0, there are two constraints on error. The first relates to the camera characteristics and location. The second relates to the scene itself, i.e. the amount of texture within the scene. For this section we assume the point p is the result of correct matches from each camera pair.

Define l as the 3D line resulting from the projection of P_0, and the epipolar line L as the projection of l onto c_i. The first step is to find t_i, the three dimensional distance along l traveled per pixel in L. Given the intrinsic and extrinsic parameters for the cameras we can compute T_i the perspective transformation for camera c_i[11]. Given a depth estimate d along the line l we can compute t_i by:

$$t_i = \frac{1}{T_i\big(l(d)\big)'} \tag{3}$$

Given t_i we can now find the error by using local slopes within the images. Since we are only finding an approximation for the error we assume that all intensities are linear. Let the three dimensional point p project onto point P_i in c_i. Since we assume that intensities are linear we can easily find the slope of the intensity value s_i, along the epipolar line L at P_i. Thus the error for the camera pair C_i is simply:

$$E_i = \frac{t_i}{s_i} \tag{4}$$

After each depth estimate is averaged together weighted by baseline lengths we can combine the different errors, E_i for i from 1 to n to find the error E_t, for the final depth estimate. Define b_i to be the baseline distance for the camera pair C_i.

$$E_t = \frac{\sum_{i=1}^{n} E_i b_i}{n \sum_{i=1}^{n} b_i} \tag{5}$$

5.5 The Relation of Accuracy and Reliability to Baseline Distances and Minimum Intensity Slopes

We will now examine property 3 of the point set; each point is the result of a weighted average of unique matches from each camera pair. That is, given an 3D point estimate p after using camera pair C_i, there should be only one possible match using camera pair C_{i+1} within a window of $E_i + E_{i+1}$ around p.

Therefore the distance between possible matches in C_{i+1} must be greater than $2(E_i+E_{i+1})$. Let p_i be a possible match within $E_i + E_{i+1}$ of p. By examining the epipolar line around P_i, the projection of p_i on c_i, we find the closest possible match P_i', to P_i. Obviously, the 2D distance d_i, from P_i to P_i' must always be at least 2 pixels, if matches have the same sign of the gradient. Consequently, if property 3 is to hold, the following must be true:

$$2\left(E_i + E_{i+1}\right) < d_i t_{i+1} \tag{6}$$

$$2\left(\frac{t_i}{s_i} + \frac{t_{i+1}}{s_{i+1}}\right) < d_i t_{i+1} \tag{7}$$

$$\frac{t_i}{t_{i+1}} < \frac{d_i s_i}{2} + \frac{s_i}{s_{i+1}} \tag{8}$$

If we assume $\frac{s_i}{s_{i+1}} \approx 1$

$$\frac{t_i}{t_{i+1}} = \frac{d_i s_i}{2} + 1 \qquad (8)$$

If we assume $d_i = 2$

$$\frac{t_i}{t_{i+1}} = s_i + 1 \qquad (9)$$

Since epipolar line length is proportional to baseline length, we can simplify (9) even further. If C_i has a baseline distance of b_i and C_{i+1} has a baseline distance of b_{i+1} then:

$$\frac{b_i}{b_{i+1}} = s_i + 1 \qquad (10)$$

Thus, the ratio of baseline distances between camera pairs should not be greater than the local slope around pixels. Therefore, as stated by property 4, a pixel should only be considered for a possible match if the local slope around the pixel is greater than the ratio of baseline distances.

5.6 The Distance Between Correct and False Points

As we stated earlier property 3 of the set of points helps guarantee that false points are separated from true points by some fixed distance. Property 3 of the point set states that each three dimensional point is the weighted averaged of unique matches from each camera pair. Thus for each match found using the camera pair of smallest baseline distance there is at most one match found per other camera pair. Therefore in order to determine the distance between points we need only look at the camera pair of smallest baseline distance C_1. As stated earlier there is at least 2 pixels between matches in camera c_1. Thus ε, the minimum distance between correct and false points, is easily computed as:

$$\varepsilon = 2t_1 - E_1 \qquad (11)$$

The distance traveled per two pixels in c_1 minus the error associated with C_1.

6. The Potential Surfaces

So far we have discussed a method for extracting correct surfaces from a set of potential surfaces. Now we will discuss the potential surfaces themselves. What is a potential surface? It is a group of three dimensional points with the potential of being the correct surface of some object in the scene. If an object within a scene has a continuous surface as seen from the reference camera, then all points which correspond to that object are in the same potential surface. Since there is usually adequate separation between correct and false points, object surfaces need not be smooth in order for potential surfaces to be successfully created. However, if objects lack texture there may be many potential surfaces which correspond to it. Thus the population of extracted correct surfaces can typically range from only 50 points to over 10,000.

6.1 Properties of the Potential Surfaces

In order to find correct points, our strategy is to link together 3D points that could belong to the same opaque surface, and then to extract the true opaque surfaces. These potential surfaces consist of three dimensional points possessing the following properties:

1. A potential surface contains either true (actual surface) or false points, never a mixture.
2. Two correct points from the same unobscured object are in the same potential surface.
3. Two points within the same potential surface must not overlap as seen from the reference camera.

6.2 Method for Grouping Points into Potential Surfaces

Creating surfaces with these properties is straightforward. As we have shown false points are separated from correct points by some fixed distance ε.

Define the projection of 3D points p and p' onto the reference camera to be P and P'. If P' is within a 2D window, W, around P then p and p' are defined to be within the same potential surface if:

Define z and z' to be the distance from p and p' to the reference camera.

$$|z - z'| < \delta \qquad (12)$$

Since it is common for points to be missing on the true surface, especially at intensity peaks and areas of low texture, W is greater than 1, in practice it is usually set between 3 and 5. As δ increases the ability of finding surfaces at extreme angles to the reference camera increases, but the likelihood of adding false points to true surfaces also increases. Therefore, we must reach a balance between δ, W and the greatest surface normal θ allowed within a potential surface. In order to ensure potential surfaces are reliably created the following must be true:

1. $\delta < \varepsilon$ (13)

2. Define p_w to be the inverse projection of P+W at the same depth as p, i.e. $T_0(p_w) = P + W$.

$$\theta < \tan\left(\frac{\varepsilon - \delta}{p_w - p}\right) \qquad (14)$$

Since (ε-δ) tends to be fairly large with respect to (p_w-p), δ can be set to a wide range of values without much affect to the final depth map. Therefore, we generally set δ to be less than $\varepsilon/4$ to minimize the probability of the method failing.

6.3 Probability of Creating Potential Surfaces Failing

This process of creating potential surfaces is completely reliable, except at surface boundaries. In the interior of a potential surface the use of camera pairs with short baseline distances forces the false matches to occur at fairly wide distances from each other, with the result that the surface creating process never combines the interior parts of a false and a true surface.

At the surface boundary, this constraint does not apply, but we can compute the probability of including a bad point in a good surface by noting the distance between the false points. The probability that first camera pair found a match within W is:

$$\frac{\delta W}{d_1 t_1} \tag{15}$$

Then the probability of all camera pairs agreeing on the match is:

$$\frac{\delta W}{d_1 t_1} \prod_{i=2}^{n} \frac{E_i + E_{i+1}}{d_i t_i} \tag{16}$$

7. Passive Lighting Experiments

To test the reliability of our algorithm we constructed a system using four cameras (Sony XC-75, 486x528) along a horizontal baseline, figure 5.1. The intrinsic and extrinsic parameters of the cameras were found using the same method as [11]. The average focal length for the cameras was 19mm, minimizing any barreling distortion. As shown in table 5.1 the baseline ratio between C_1 and C_2 is 2.6, thus pixels with local intensity slopes less than 3 were not considered for possible matches. To obtain better results a Lapacian of Guassian with σ=3 pixels was used to normalize the images. This was nessecary due to non-lambertain objects, and the gains and offsets of the cameras not being calibrated. Typical running time for the three scenes was 30 seconds for the first step and 30 seconds for the last two steps on an Indigo 2 xz. The objects within the scenes were about 1.5m from the reference camera. The average theoretical error(5) for the scenes is 0.47mm. The probability(16) of adding false points or matches onto correct surfaces is 0.13%. The average distance between false and correct surfaces was 83.9mm, with the minimum distance ε = 29.6mm.

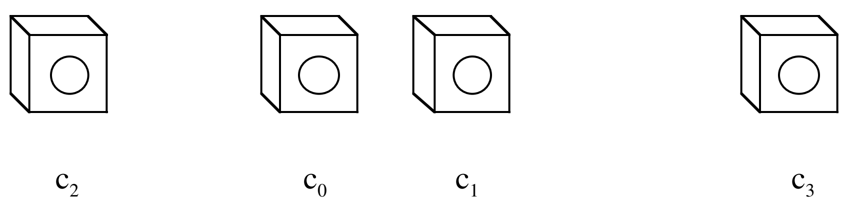


Figure 10: Camera setup for experimental results.

	Camera 1	Camera 2	Camera 3
Baseline Distance(mm)	72.19	187.47	336.19
mm/pixel	16.77	6.42	3.66

Table 1: Distance from each camera to reference camera, and the average three dimensional distance traveled in the workspace per pixel in each camera.

7.1 Model Town Scene

The results using the model town illustrates the reliablity of the algorithm. No false regions where picked, and only a small number of false points were included with the true regions. Areas of low texture produce no points as expected.

Camera 0

Camera 1

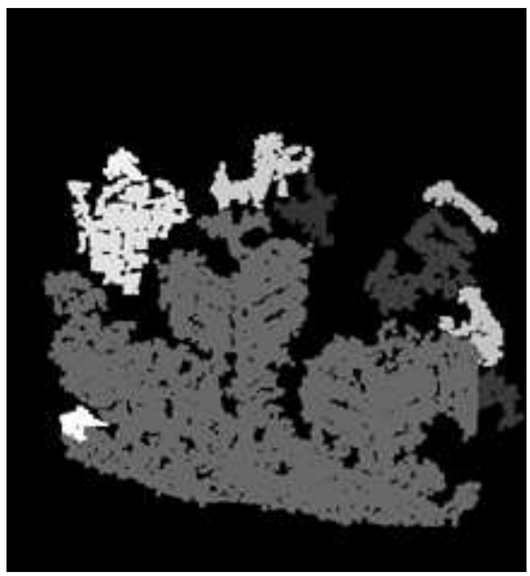
Camera 2

Camera 3

Figure 11: Intensity images from the four cameras

(a)

(b)



(c)

Figure 12: Results from Algorithm. (a) Depth Map. (b) Final point set viewed from above. (c) Regions picked.

Number of points after second step	Final number of true points	Number of potential surfaces	Number of potential surfaces with more than 80 points.	Number of potential surfaces picked as correct.
33,785	15,785	2610	11	10

7.2 Left-to-Right Constaint

The next scene illustrates two advantages of the algorithm. First, the vertical stick changes order within the images relative to the stripped paper. Since the left-to-right constraint does not apply to our algorithm unlike most edge-based algorithms[17] both the stripped paper and stick are found. The missing sections of the paper are due to occlusion by the wooden stick in one of the images. Second, the stripped pattern on the piece of paper is a repetitive pattern. With traditional multi-baseline stereo algorithms this would typically produce gross errors, however, no errors resulted using our algorithm.

Camera 0

Camera 1

Camera 2

Camera 3

Figure 13: Intensity images from the four cameras

(a) (b)

(c)

Figure 14: Results from Algorithm. (a) Depth Map. (b) Final point set viewed from side. (c) Regions picked.

Number of points after second step	Final number of true points	Number of potential surfaces	Number of potential surfaces with more than 80 points.	Number of potential surfaces picked as correct.
24,065	12,815	1336	11	9

7.3 Shirt and Books

Even though there is low texture on many areas of the shirt (lowest object) correct points are still found. However, the algorithm failed to find points on one of the fingers due to low texture. Finally notice how no points were found at the center of the left book. This is caused by spectular reflection.

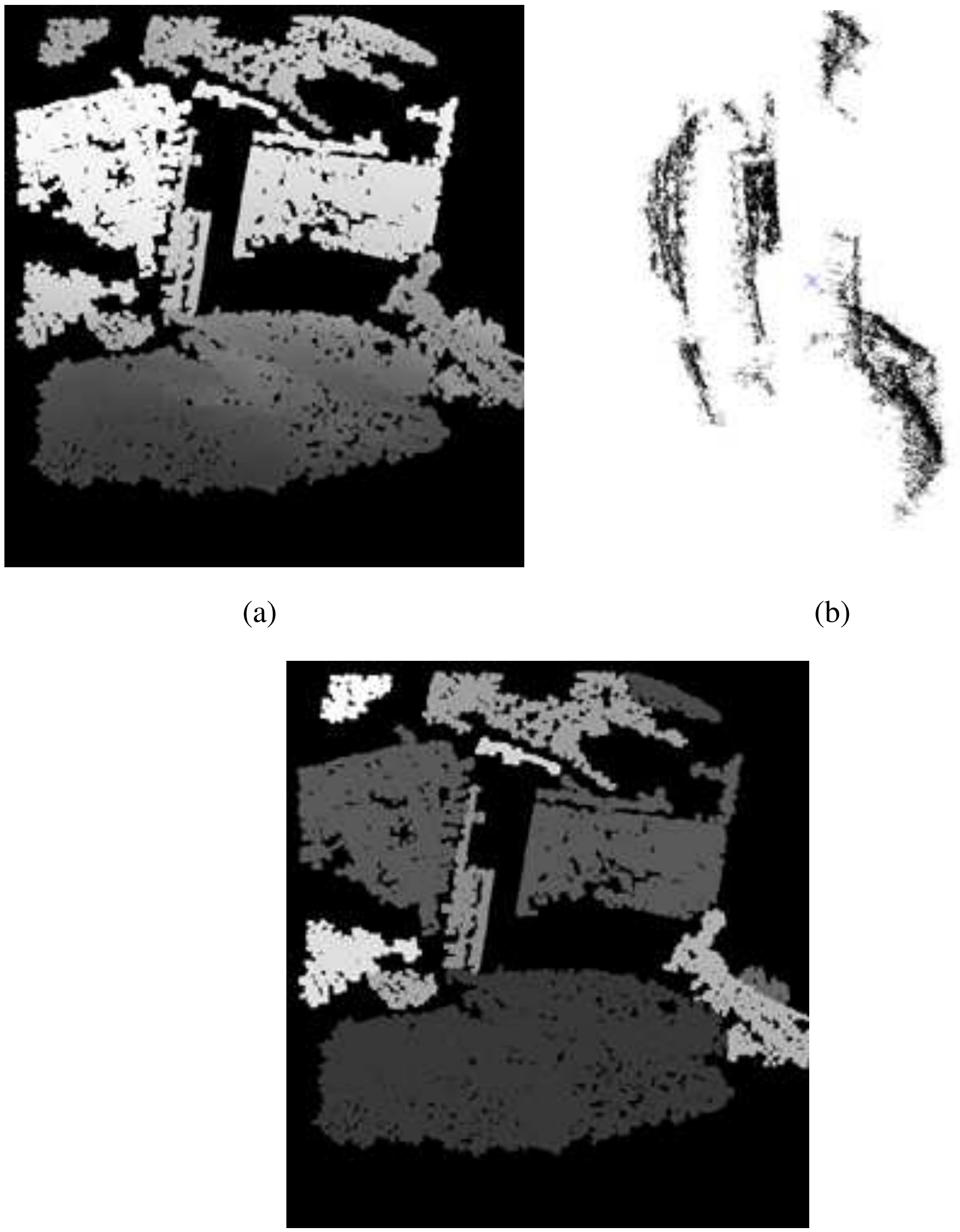
Camera 0

Camera 1

Camera 2

Camera 3

Figure 15: Intensity images from the four cameras



(a) (b)

(c)

Figure 16: Results from Algorithm. (a) Depth Map. (b) Final point set viewed from side. (c) Regions picked.

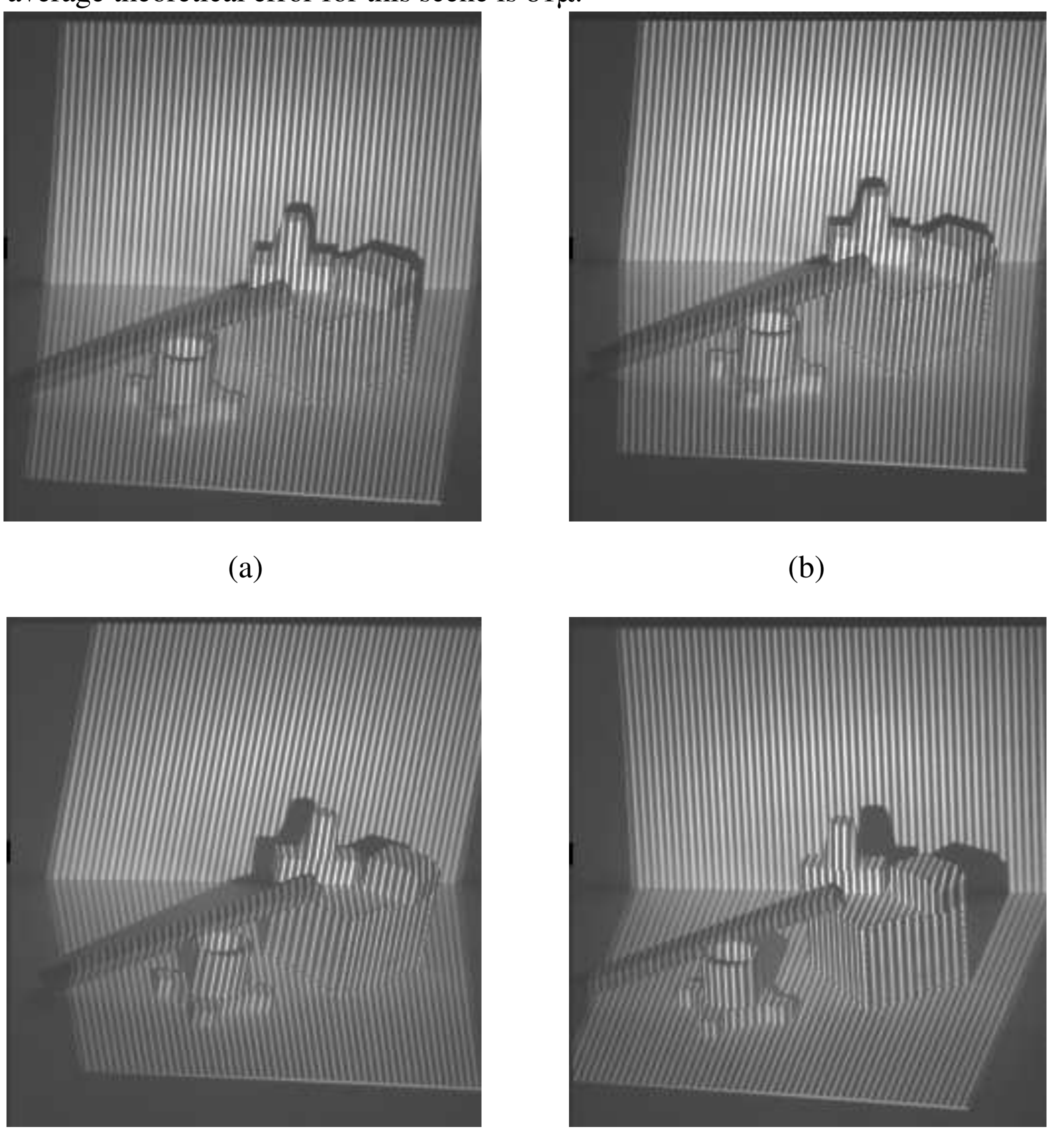
Number of points after second step	Final number of true points	Number of potential surfaces	Number of potential surfaces with more than 80 points.	Number of potential surfaces picked as correct.
41,743	21,980	889	13	12

8. Active Lighting Experiments

In order to find the best case results of our algorithm we used active lighting on two scenes in order to create texture. The projected vertical line pattern varies from light to dark linearly. Since the gains and offsets of each camera vary, we have linearly adjusted the intensity values based on the local minimum and maximum pixel values. The camera setup is identical to that of the previous section. To measure the results we fit the points to a plane or cylinder to measure the average error, standard deviation and maximum error. The average local intensity slopes for the planar and cylinder scenes are 29.7 and 35.6 respectively. We believe the difference between the theoretical error and average error is due to the objects not being perfectly Lambertain and our assumption that the intensities vary linearly.

8.1 Plane Experiments

The average theoretical error for this scene is 81μ.



(a) (b) (c) (d)

Figure 17: Intensity images from the four cameras

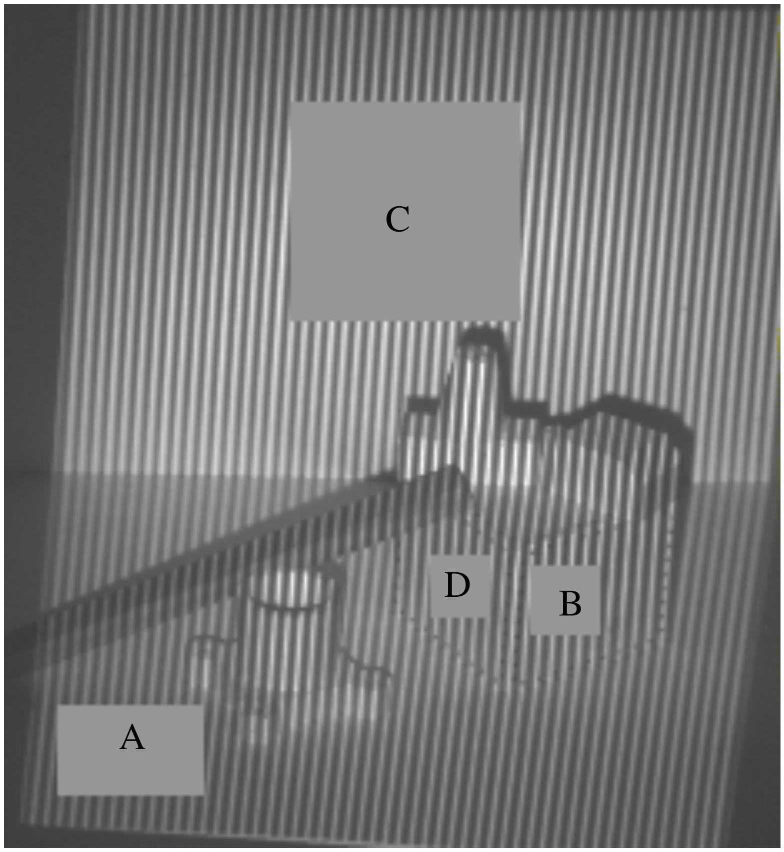


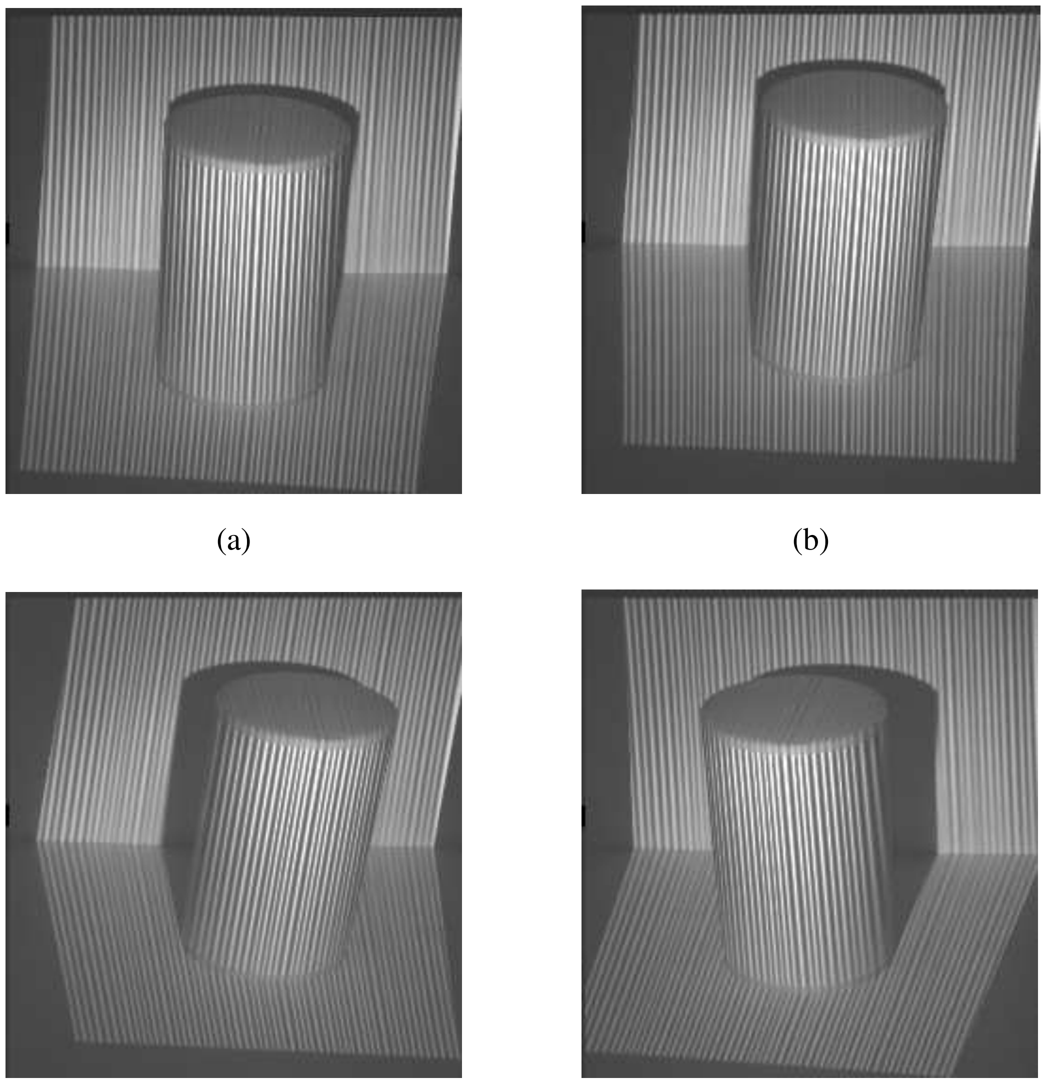
Figure 18: Planar regions tested

Region	Number of Points	Surface Normal	Average Error	Standard Deviation	Maximum Error
A	3,022	(-0.999, -0.014, 0.0088)	258μ	204μ	1480μ
B	1,155	(-0.007, -0.554, -0.833)	267μ	195μ	1352μ
C	12,846	(0.045, -0.036, -0.998)	300μ	238μ	2198μ
D	890	(0.010, 0.828, -0.561)	218μ	166μ	1305μ

Table 2: Number of points, surface normal, average error, standard deviation and maximum error of the planar regions

8.2 Cylinder Experiment

The average theoretical error for this scene is 69μ.



(a) (b)

(c) (d)

Figure 19: Intensity images from the four cameras

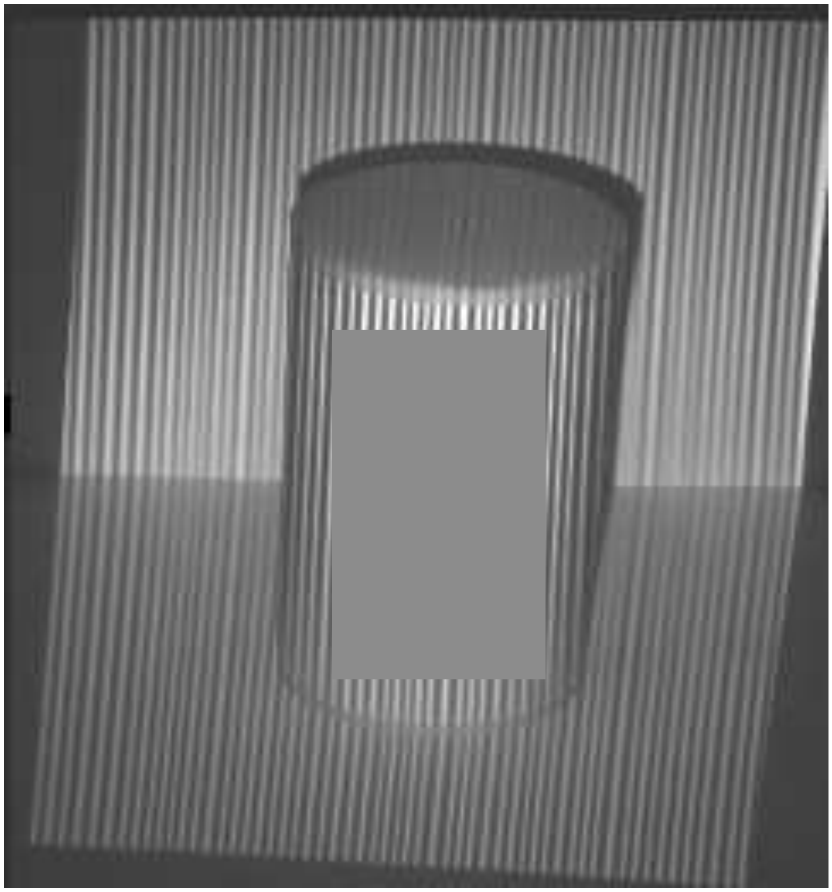


Figure 20: Area of cylinder tested

Number of Points	Theoretical Error	Average Error	Standard Deviation	Maximum Error
9,560	69µ	179µ	169µ	1719µ

Table 3: Number of points, theoretical error, average error, standard deviation and maximum error of the cylinder data

9. Conclusion

Within this paper we have demonstrated a method for uniquely identifying corresponding image regions in which the matching problem is well-posed. In order to identify the image regions we have transformed the problem into one of identifying correct 3D surfaces from a set of potential surfaces. These correct surfaces can then be projected onto the images to identify the regions, however this is typically not necessary since the correct 3D surfaces are actually the desired output. We have shown there exists a method for uniquely extracting the correct surface from the set of potential surfaces when the surface is not occluded by another surface by greater than a distance $\varepsilon/2$. Furthermore we have shown the construction of potential surfaces is stable since false surfaces are separated from correct by at least ε.

In order to demonstrate the effectiveness of our algorithm we have built a four camera system. We have results from three complex scenes showing the resulting correct surfaces. Accuracy measurement were also done on two scenes with an average error of 250μ and 179μ.

References

[1] Ballard, D.H. and Brown, C.M. *Computer Vision*, Englewood Cliffs, N.J., 1982.

[2] Barnard, S.T. and Fischler, M.A. “Computational Stereo” *Computing Surveys*, 14(4) 1982:p.553-572.

[3] Barnard, S.T. “Stochastic Stereo Matching Over Scale” *Int’l Journal of Computer Vision*, 1989:p. 17-32.

[4] Belhumer, P.N. “A Binocular Stereo Algorithm for reconstructing Sloping, Creased and Broken Surfaces” *Int’l Conf. on Computer Vision*, 1993:p. 431-438.

[5] Bertero, M., Poggio, T. and Torre, V. “Ill-posed Problems in Early Vision” Artificial Intelligence Lab. Memo, No. 924, MIT, Cambridge, MA, 1987.

[6] Chen, J. and Medioni, G. “Parallel Multiscale Stereo Matching Using Adaptive Smoothing” *ECCV*, 1990:p. 99-103.

[7] Faugeras, O.D. *Three-Dimensional Computer Vision: A Geometric Viewpoint.* 1993, MIT Press.

[8] Horn, B.K.P. *Robot Vision,* MIT Press-McGraw-Hill, Cambridge-New York, 1986.

[9] Jenkin, M.R.M. and Jepson, A.P. “Recovering Local Surface Structure Through Local Phase Difference Measurements” *Computer Vision, Graphics and Image Processing: Image Understanding*, 59(1):p. 72-93.

[10] Kanade, T. and Okutomi, M. “A stereo Matching Algorithm with an Adaptive Window: Theory and Experiment” *IEEE Transactions on Pattern Anaylsis and Machine Intelligence*. 16:9, Sept. 1994.

[11] Kang, S.B, Webb, J.A., Zitnick, C.L. and Kanade, T. “A Multibasline Stereo System with Active Illumination and Real-time Image Acquisition” *Proc. of Int’l Conf. on Computer Vision*, 1995.

[12] Maimone, M.W. “Charaterizing Stereo Matching Problems using Local Spatial Frequency” CMU-CS-96-125, May 1996.

[13] Marr, D. and Poggio, T. “A Theory of Human Stereo Vision” *Proc. Roy. Soc. London*, Vol. B 204, 1979:p. 301-328.

[14] Moravec, H.P. “Obstacle Avoidance and Navigation in the Real World by a seeing Robot Rover” STAN-CS-80-813, Sept. 1980.

[15] Nakahara, T. and Kanade, T. “Experiments in Multiple-Baseline Stereo” CMU-CS-93-102.

[16] Nakamura, Y., Matsuura, T., Satoh, K. and Ohta, Y. “Occlusion Detectable Stereo - Occlusion Patterns in Camera Matrix” *IEEE Conf. on Computer Vision and Pattern Recognition*, 1996.

[17] Ohta, Y. and Kanade, T. “Stereo by Intra- and Inter-Scanline Search Using Dynamic Programming” *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. PAMI-7, No.2 , March 1985:p. 139-154.

[18] Okutomi, M. and Kanade, T., “A Locally Adaptive Window for Signal Matching” *International Journal of Computer Vision*, 7:2, 1992:p. 143-162.

[19] Okutomi, M. and Kanade, T. “A Multiple-Baseline Stereo” *IEEE Trans. on PAMI*, 1993. 15(4):p. 353-63.

[20] Poggio, T. and Torre, V. “Ill-Posed Problems and Regularization Analysis in Early Vision” Artificial Intelligence Lab. Memo, No. 773, MIT, Cambridge, MA, 1984.

[21] Ross, B. “A Practical Stereo Vision System” *Proc. of the IEEE Int’l Conf. on Computer Vision and Patt. Recog.* 1993:p. 148-153.

[22] Stewart, C.V. and Dyer, C.R. “The Trinocular Stereo Algorithm for Overcoming Binocular Matching Errors” *Proc. Second Int’l Conf. on Computer Vision*, 1988:p134-138.

[23] Webb, J. “Implementation and Performance of Fast Parallel Multi-baseline Stereo Vision” *Proc. of Image Understanding Workshop*, 1993:p. 1005-1012.

[24] Weng, J. “A Theory of Image Matching” *3rd Int’l Conf. of Computer Vision*, 1990:p. 200-209.

[25] Xiong, Y. “High Precision Image Matching and Shape Recovery” CMU-RI-TR-95-35, Sept. 1995.