Computer Vision and Image Understanding, 63:3, pp. 542-567, May 1996

A Maximum Likelihood Stereo Algorithm*

Ingemar J. Cox, Sunita L. Hingorani, Satish B. Rao
NEC Research Institute
4 Independence Way
Princeton, NJ 08540
ingemar|sunita|satish@research.nj.nec.com

Bruce M. Maggs
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
bmm@cs.cmu.edu

Abstract

A stereo algorithm is presented that optimizes a maximum likelihood cost function. The maximum likelihood cost function assumes that corresponding features in the left and right images are Normally distributed about a common true value and consists of a weighted squared error term if two features are matched or a (fixed) cost if a feature is determined to be occluded. The stereo algorithm finds the set of correspondences that maximize the cost function subject to ordering and uniqueness constraints.

The stereo algorithm is independent of the matching primitives. However, for the experiments described in this paper, matching is performed on the *individual pixel intensities*. Contrary to popular belief, the pixel-based stereo appears to be robust for a variety of images. It also has the advantages of (i) providing a *dense* disparity map, (ii) requiring *no* feature extraction and (iii) *avoiding* the adaptive windowing problem of area-based correlation methods. Because feature extraction and windowing are unnecessary, a very fast implementation is possible.

Experimental results reveal that good stereo correspondences can be found using only ordering and uniqueness constraints, i.e. without *local* smoothness constraints. However, it is shown that the original maximum likelihood stereo algorithm exhibits multiple global minima. The dynamic programming algorithm is guaranteed to find one, but not necessarily the same one for each epipolar scanline causing erroneous

*Portions of this work were originally reported at the 1992 British Machine Vision Conference [11] and the 1994 Int. Conf. on Computer Vision and Pattern Recognition [10].

correspondences which are visible as small local differences between neighboring scanlines.

Traditionally, regularization, which modifies the original cost function, has been applied to the problem of multiple global minima. We developed several variants of the algorithm that avoid classical regularization while imposing several global *cohesivity constraints*. We believe this is a novel approach that has the advantage of guaranteeing that solutions minimize the original cost function and preserve discontinuities. The constraints are based on minimizing the total number of horizontal and/or vertical discontinuties along and/or between adjacent epipolar lines, and local smoothing is avoided. Experiments reveal that minimizing the sum of the horizontal and vertical discontinuities provides the most accurate results. A high percentage of correct matches and very little smearing of depth discontinuities are obtained.

An alternative to imposing cohesivity constraints to reduce the correspondence ambiguities is to use more than two cameras. We therefore extend the two camera maximum likelihood to N-cameras. The N-camera stereo algorithm determines the "best" set of correspondences between a given pair of cameras, referred to as the *principal* cameras. Knowledge of the relative positions of the cameras allows the 3D point hypothesized by an assumed correspondence of two features in the principal pair to be projected onto the image plane of the remaining $N-2$ cameras. These $N-2$ points are then used to verify proposed matches. Not only does the algorithm explicitly model occlusion between features of the principal pair, but the possibility of occlusions in the $N-2$ additional views is also modelled. Previous work did not model this occlusion process, the benefits and importance of which are experimentally verified. Like other multi-frame stereo algorithms, the computational and memory costs of this approach increase linearly with each additional view. Experimental results are shown for two outdoor scenes. It is clearly demonstrated that the number of correspondence errors is significantly reduced as the number of views/cameras is increased.

1 Introduction

Stereo algorithms need to determine the set of correct correspondences between features in (at least) two images. While the epipolar constraint of stereo reduces the search space to one dimension (along the epipolar lines), the correspondence problem is still difficult. There are a number of reasons for this, including, (1) feature detection is not perfectly reliable, so false features may be detected in one or other of the images, (2) features in one image may be occluded in the other image and (3) establishing the similarity between two features is confounded by noise in the two images. It should also be recognized that camera calibration is an important component of any stereo algorithm. However, in this paper, we ignore the calibration issues and assume that the epipolar lines are given.

The motivation for this work was four-fold. First, was to derive a maximum likelihood (ML) formulation of the stereo problem from a sensor fusion perspective and in this regard we were strongly influenced by the work of Pattipati *et al* [29]. The ML estimate does not require knowledge of a prior probability density function (which may be difficult to estimate) and this distinguishes it from the Bayesian maximum a posteriori (MAP) estimate. Of course, for diffuse priors, the ML and MAP estimates will coincide. Following the sensor fusion methodology of Pattipati *et al* [29] allows the cost of matching or occluding a feature to be derived based on measurable physical and statistical characteristics of the scene and the cameras. It should be noted that the occlusion process is explicitly modeled. This stereo framework is developed in Section (2) and is, at the algorithmic level, independent of the feature primitives.

Second, was an interest in re-evaluating pixel-based stereo in which matching is performed on the *individual pixel intensities*. In this respect, the work is related to the intensity-based stereo work of Horn [17] and Gennert [16] and most recently Belhumeur [5, 4][1]. A pixel-based algorithm has the benefits of (i) providing a *dense* disparity map, (ii) requiring *no* feature extraction and (iii) *avoiding* the adaptive windowing problem of area-based correlation methods.[2] However, there is a commonly held belief that since "stereo projections do not preserve photometric invariance", pixel-based stereo is "in general doomed to failure" [14]. The experimental results described herein contradict this opinion. In practice, the corresponding intensities in left and right views are usually quite close. This will certainly be true for frontal planar surfaces viewed under Lambertian reflection. Gennert [16] modelled the photometric variances due to stereo projections and showed that the intensities of two corresponding points are approximately related by a spatially varying multiplicative term that is a function or surface orientation and reflectance models. Using this model, Gennert developed a stereo algorithm that matched individual pixel intensities based on a complex cost function consisting of a linear combination of a brightness matching error, and disparity, multiplier and vertical smoothness penalties together with several matching constraints. Minimization of this functional is difficult, computationally expensive and convergence cannot be guaranteed. Nevertheless interesting results were reported, that support the premise that pixel-based stereo is practical. The maximum likelihood framework described in Section (2) assumes that corresponding pixels are normally distributed about some true common value.[3] Experiments described in Section (5) revealed that changes in illumination conditions and differences in camera responses were the principal source of errors to the normal assumption. These effects appear to dominate over the photometric variances modelled by Gennert. The changes in illumination and or camera responses were modeled by constant multiplicative and additive factors that can be easily estimated and compensated for *automatically* prior

[1] Whose work is contemporaneous with ours

[2] Note however, that the underlying framework of Section (2) does not rely on pixel matching. Indeed, the measurement vector, $\mathbf{z}_i$ may also consist of a set of intensities, i.e. window-base, or edge parameters, i.e. feature-based, provided the underlying statistical assumption of Normal distributions is met.

[3] This assumption is implicit in Belhumeur's work as well.

to the stereo matching. This simple correction procedure improves the performance of the ML algorithm and is a significant contribution to the practical application of pixel-based stereo algorithms.

The third motivation was an attempt to design a stereo algorithm with as few constraints as possible. This is in contrast to Belhumeur's work which is motivated in part to building more sophisticated Bayesian priors or world models. Both approaches are interesting. Algorithms exploiting more sophisticated models would be expected to perform better on imagery that satisfies the *a priori* assumptions, but their applicability may be confined to a restricted class of images. Conversely algorithms exploiting only a minimal number of constraints may be applicable over a wider class of scenes though their performance may sometimes be inferior to more specialized algorithms. The initial ML algorithm imposes no smoothness constraints at all, only uniqueness and order, yet performs surprisingly well.

The maximum likelihood stereo algorithm assumes that any two corresponding features are normally distributed about their true value. This leads to a local matching cost that is the weighted squared error between the features. This is the only local cost associated with the matching of two features, i.e. there is no *local* smoothness cost. This is interesting since many previous stereo algorithms include a cost based on the disparity [6, 15, 23] or disparity gradient (the difference in disparity between two pixels divided by their distance apart) [31, 32] of neighboring pixels. Our work suggests that a *local* smoothness cost may not (always) be necessary.

The global cost function that is eventually minimized is the sum of the local costs of matching pixels plus the sum of occlusion costs for unmatched pixels. The global optimization is efficiently performed in 1D along each epipolar line assuming monotonic ordering using dynamic programming. Dynamic programming has been used in many stereo algorithms, e.g. Baker and Binford [3], Ohta and Kanade [27], Geiger *et al* [15] and Belhumeur [5]. These algorithms can be characterized by the global cost function that is minimized. Baker and Binford first determine correspondences between edges using dynamic programming and then perform another stage of dynamic programming on the intensity values between consecutive pairs of corresponding edges in order to "fill in the gaps". Intensity *variances* provide a metric for comparing intensity values. Ohta and Kanade match intensity segments based on the variance of each segment. The cost of an occlusion is not fixed but a function of the variance of the occluded region. Their edge-based method is particularly significant in their effort to extend the global optimization across epipolar lines to find consistency across scanlines. We believe that the use of a variance measure to compare features is not appropriate. First, the variance measure ignores the actual intensity values, yet two regions of equal variance might have significantly different (mean) intensity values. Secondly, since the cost of matching is proportionally to the variance, there is an inherent bias against matching corresponding textured regions.

The experiments described in Section (3) demonstrate that good correspondences can be found using only ordering and uniqueness constraints, i.e. without *local* smoothness constraints. This is an interesting result. Correspondences errors are, however, clearly

visible, and an investigation of their source revealed that multiple *global* minimum exist. This gives rise to (minor) artifacts in the disparity map. Similar multiple global minima may exist for other stereo algorithms. The existence of multiple global minima suggest the need for additional constraints. Traditionally, such constraints have been imposed by modifing the original cost function with one or more regularization terms. A fourth motivation for this work was to investigate alternative procedures to classical regularization. In Section (4), an alternative procedure is developed in which the *globally smoothest* solution, i.e. the solution with the least number of discontinuities, from amongst the many possible solutions is recovered. We believe this approach, described briefly in [30] but apparently not utilized, is of significant interest. The spirit of the approach is to give precedence to the data and only apply constraints in circumstances where the data can be ambiguously (multiply) interpreted, i.e. prior knowledge (in the form of constraints) is used as seldom as possible in order to reduce any bias. The additional smoothness or cohesivity constraints can be incorporated directly into the dynamic programming algorithm with relatively little additional computational cost. The cohesivity constraints are based on minimizing the total number of horizontal and/or vertical discontinuties between epipolar lines. Experiments reveal that minimizing the sum of the horizontal and vertical discontinuities provides the most accurate results.

Geiger *et al* [15] and Belhumeur and Mumford [5] have developed Bayesian formulations of the stereo problem with strong similarities to the work described here [4]. While Geiger *et al* match intensity windows rather than individual pixels, the major distinction of the ML approach described here is the simplicity of the local cost function and the novel manner in which *global* cohesivity constraints are imposed.

Matthies [24] has also derived ML and maximum a posteriori (MAP) stereo algorithms for the cases of (i) a statistically uncorrelated disparity field and (ii) a 1D correlated disparity field within the epipolar scanline. The former case reduces to a sum of squared differences over independent windows while the latter case is formulated as a dynamic programming algorithm to minimizes the sum of squared differences in the individual pixel intensities plus a regularization term consisting of the sum of squared differences between neighbouring disparities. While the underlying statistical framework is similar there are significant differences between the two approaches. Most significantly, Matthies algorithms do not model the occlusion process. Further, though Matthies 1D correlated model can be considered to match individual pixels a regularization term is also present while the uniqueness and monotonic ordering constraints are absent.

Section (5) presents experimental results for a selection of both synthetic and natural stereograms. These results demonstrate the significant improvement in performance provided by the intensity normalization procedure and the global cohesivity constraints. Also of interest is the result of applying the algorithm to a synthetic ellipsoid pair. While the surface is completely textureless, the ellipsoidal shape is recovered. This shape from disparate intensity also suggests that the normal assumption is approximately correct for

[4] All three approaches were developed contemporaneously but independently.

quite curved surfaces.

While very good correspondence accuracy is achieved, some errors still persist. Several authors have observed that more reliable correspondences can be found by using more than two image frames. For example, Ayache [1, 2] describe a trinocular stereo system in which hypothesized matches between images features in Cameras 1 and 2 are directly tested by a simple verification step that examines a specific point[5] in the third image. Ayache notes that a "third camera reinforces the geometric constraints. This allows simplification of the matching algorithms, and a reduction of the importance of heuristic constraints. This makes the procedure faster and the results more robust." Nevertheless, three cameras do not eliminate all errors of correspondence and further improvements can be obtained by using more cameras.

Several researchers have described N-frame stereo algorithms. Two basic strategies exist. However, the two approaches are *not* concerned with the same problem. The first approach attempts to *reduce the noise in the* $3D$ *position estimates* by filtering over N stereo *pairs*. For example, early work by Moravec [26] described a common geometric configuration in which all of the N images frames lie on a horizontal line. Moravec's algorithm takes the set of features in the central image and then estimates the feature correspondences for each of the $N-1$ stereo pairs. These (noisy) results are then combined through an additive weighting that attempts to model the relative uncertainty in the features position due to variations in the length of the baseline[6]. More recently, Matthies *et al* [25] have proposed a Kalman filter based approach to reducing the positional error by a weighted averaging of the position estimates from many independent stereo pairs.

Note that in the above approaches, each stereo pair is matched independently, without exploiting the geometric constraints afforded by the N-camera configuration. Second, the $N-1$ independent estimates of a feature's position are summed to form a weighted estimate of position. Implicit in such an approach, is the assumption that the errors in the positional estimates are normally distributed about a nominal value. However, while random errors most certainly exist, our experience suggests that many correspondence errors are *not* random. Instead, such errors are due to inherent ambiguities in the matching process due to a combination of the peculiarities of the particular cost function to be minimized, any heuristic constraints that are applied and/or the data itself. Thus, it appears to be quite common for systematic errors to arise which cannot be removed by a weighted averaging. If incorrect correspondences were entirely random, then averaging over a stationary temporal sequence of stereo pairs would converge to a perfect solution. However, this does not happen in practice.

The second strategy is not concerned with reducing noise but with reducing the number of correspondence errors. These approaches generalize and exploit the geometric constraints of trinocular stereo to a N-camera configuration. For example, Kanade *et al* [20] describe a multi-baseline stereo algorithm in which the similarity between two fea-

[5]This point is the projection of the 3D point hypothesized by the assumed correspondence of the two features in Frames 1 and 2 onto the image plane of camera 3

[6]The standard deviation in the estimated position of a feature is inversely proportional to the baseline.

tures is measured by the sum of squared differences (SSD) of the intensities over a window. Given an image P_0, the SSD is calculated for each of the $N-1$ possible image pairs. The geometric constraints are then invoked in order to calculate the sum of the SSD's (SSSD). Kanade *et al* show well localized minima in the SSSD cost function, indicating that much of the ambiguity present in a single SSD estimate has been eliminated. The reader is also directed to earlier work by Tsai [34] which describes a similar algorithm that uses a different similarity measure. However, the experimental results of Tsai are for simulated images, making comparison with Kanade *et al* difficult.

Both Kanade *et al* and Tsai's methods are window-based correlation methods. These techniques potentially suffer from (1) the blurring of depth boundaries when a window overlaps two different surfaces, and (2) the effects due to a lack of an explicit occlusion model. Recently, Kanade has addressed the issue of occlusion and spurious features [21] with a method that analyses the shape of the SSSD curve in the vicinity of minima. However, explicit modelling of the occlusion process is more desirable.

Very recently, Roy and Meuniers' [33] described a multi-frame stereo algorithm that has many similarities with the algorithm described here. Once again a particular image P_0 is selected and the correspondences are found for each of $N-1$ stereo pairs using a dynamic programming algorithm. However, for each stereo pair, Roy and Meunier apply the geometric constraints to the remaining $N-2$ images to verify the proposed matches. The cost of matching two features becomes the sum of the squared differences between features $\mathbf{z}_{i_1}$ in the left image and $\mathbf{z}_{i_N}$ in the right image and the $(N-2)\mathbf{z}_{i_s}$ where $\mathbf{z}_{i_s}$ is the projection of the 3D point onto the image plane of camera s. After determining the $N-1$ stereo correspondences, the $(N-1)$ depth maps are summed to produce the final depth map[7]. We believe that such a summation is incorrect, for reasons discussed earlier. Another problem not addressed by this algorithm is the fact that the proposed cost of matching two features does not account for the possibility that the feature may be occluded in an intermediate view, i.e. it is assumed that corresponding points are visible in *all* views.

In Section (6), we derive a maximum likelihood cost function for an N camera arbitrary stereo configuration. The algorithm determines the "best" set of correspondences between a given pair of images, which we refer to as the *principal* pair[8]. The remaining $N-2$ images are used to "verify" hypothesized matches between the principal pair of images. The algorithm models occlusion not only in the principal stereo pair but also in the $N-2$ intermediate views. Like other multi-frame stereo algorithms, the computational and memory costs of this approach increase linearly with each additional view.

Experimental results are then presented for two outdoor scenes. The results clearly demonstrate significant reductions in correspondence errors as more cameras are used to verify matches. A small amount of artifacting is present in the solutions that is the

[7]A final (composite) occlusion map is determined by calculating the logical OR of the individual occlusion maps.

[8]Thus, a point visible in the left image but not in the right image will be labelled occluded even if corresponding points are visible in the intermediate views to determine its depth.

result of the cohesivity constraints and the implicit bias of the algorithm for frontal planar surfaces. Comparison with a disparity map generated *without* the intermediate occlusion model is also presented.

Finally, Section (8) concludes with a discussion of the advantages and disadvantages of the ML algorithm and possible future work.

2 The Maximum Likelihood Cost Function

In this section, the cost of matching two features, or declaring a feature occluded is first derived, then a global cost function that must be minimized is derived. To begin, we introduce some terminology as developed by Pattipati *et al* [29]. Let the two cameras be denoted by $s = \{1, 2\}$ and let $\mathbf{Z}_s$ represent the set of measurements obtained by each camera along corresponding epipolar lines: $\mathbf{Z}_s = \{\mathbf{z}_{i_s}\}_{i_s=0}^{m_s}$ where m_s is the number of measurements from camera s and $\mathbf{z}_{0_s}$ is a dummy measurement, the matching to which indicates no corresponding point. For epipolar alignment of the scanlines, $\mathbf{Z}_s$ is the set of measurements along a scanline of camera s. The measurements $\mathbf{z}_{i_s}$ might be simple scalar intensity values or higher level features. Each measurement $\mathbf{z}_{i_s}$ is assumed to be corrupted by additive, white noise.

The condition that measurement $\mathbf{z}_{i_1}$ from camera 1, and measurement $\mathbf{z}_{i_2}$ from camera 2 originate from the same location, $\mathbf{X}$, in space, i.e. that $\mathbf{z}_{i_1}$ and $\mathbf{z}_{i_2}$ correspond to each other is denoted by Z_{i_1,i_2}. The condition in which measurement $\mathbf{z}_{i_1}$ from camera 1 has no corresponding measurement in camera 2 is denoted by $Z_{i_1,0}$ and similarly for measurements in camera 2. Thus, $Z_{i_1,0}$ denotes occlusion of feature z_{i_1} in camera 2.

Next, we need to calculate the *local* cost of matching two points $\mathbf{z}_{i_1}$ and $\mathbf{z}_{i_2}$. The likelihood that the measurement pair Z_{i_1,i_2} originated from the same point $\mathbf{X}$ is denoted by $\Lambda(Z_{i_1,i_2} \mid \mathbf{X})$ and is given by

$$\Lambda(Z_{i_1,i_2} \mid \mathbf{X}) \;=\; \left(\frac{1-P_D}{\phi}\right)^{\delta_{i_1 i_2}} \left[P_D p(\mathbf{z}_{i_1} \mid \mathbf{X}_k) \times P_D p(\mathbf{z}_{i_2} \mid \mathbf{X}_k)\right]^{1-\delta_{i_1,i_2}} \tag{1}$$

where δ_{i_1,i_2} is an indicator variable that is unity if a measurement is not assigned a corresponding point, i.e. is occluded, and zero otherwise and ϕ is the field of view of the camera The term $p(\mathbf{z} \mid \mathbf{X})$ is a probability density distribution that represents the likelihood of measurement $\mathbf{z}$ assuming it originated from a point $\mathbf{X} = (x, y, z)$ in the scene. The parameter P_D represents the probability of detecting a measurement originating from $\mathbf{X}$ at sensor s. This parameter is a function of the number of occlusions, noise etc. Conversely, $(1 - P_D)$ may be viewed as the probability of occlusion. If it is assumed that the measurements vectors $\mathbf{z}_{i_s}$ are Normally distributed about their ideal value $\mathbf{z}$, then

$$p(\mathbf{z}_{i_s} \mid \mathbf{X}) = \mid (2\pi)^d \mathbf{S}_s \mid^{-\frac{1}{2}} exp\left\{-\frac{1}{2}(\mathbf{z}-\mathbf{z}_{i_s})' \mathbf{S}_s^{-1} (\mathbf{z}-\mathbf{z}_{i_s})\right\} \tag{2}$$

where d is the dimension of the measurement vectors $\mathbf{z}_{i_s}$ and $\mathbf{S}_s$ is the covariance martix associated with the error $(\mathbf{z} - \mathbf{z}_{i_s})$. Since the true value, $\mathbf{z}$, is unknown we approximate it

by maximum likelihood estimate $\hat{\mathbf{z}}$ obtained from the measurement pair Z_{i_1,i_2} and given by

$$\mathbf{z} \approx \hat{\mathbf{z}} = \frac{\mathbf{S}_{i_1}^{-1}\mathbf{z}_{i_1} + \mathbf{S}_{i_2}^{-1}\mathbf{z}_{i_2}}{(\mathbf{S}_{i_1}^{-1} + \mathbf{S}_{i_2}^{-1})} \tag{3}$$

where $\mathbf{S}_{i_s}$ is the covariance associated with measurement $\mathbf{z}_{i_s}$.

Now that we have established the cost of the individual pairings Z_{i_1,i_2}, it is necessary to determine the total cost of all pairs. Denote by γ a feasible pairing of all measurements and let , be the set of all feasible partitions, i.e. , $= \{\gamma\}$. Then we wish to find the pairings or partition γ that maximizes $L(\gamma)/L(\gamma_0)$ where the likelihood $L(\gamma)$ of a partition is defined as

$$L(\gamma) = p(\mathbf{Z}_1, \mathbf{Z}_2 \mid \gamma) = \prod_{Z_{i_1,i_2} \epsilon \gamma} \Lambda(Z_{i_1,i_2} \mid \mathbf{X}) \tag{4}$$

The maximization of $L(\gamma)/L(\gamma_0)$ is eqivalent to

$$\min_{\gamma \epsilon \Gamma} J(\gamma) = \min_{\gamma \epsilon \Gamma} [-\ln(L(\gamma))] \tag{5}$$

which leads to

$$\begin{aligned} \min_{\gamma \epsilon \Gamma} J(\gamma) = \min_{\gamma \epsilon \Gamma} \quad & \textstyle\sum_{Z_{i_1,i_2} \epsilon \gamma} \Big\{ \quad \delta_{i_1,i_2} \ln \left(\frac{P_D^2 \phi}{(1 - P_D) \mid (2\pi)^d \mathbf{S} \mid^{\frac{1}{2}}} \right) + \\ & (1 - \delta_{i_1,i_2}) \quad \left[\frac{1}{4} (\mathbf{z}_{i_1} - \mathbf{z}_{i_2})' \mathbf{S}^{-1} (\mathbf{z}_{i_1} - \mathbf{z}_{i_2}) \right] \Big\} \end{aligned} \tag{6}$$

assuming that the covariances S_{i_s} are equal.

The first term of the summation represents the cost of an occlusion in the left or right views, while the latter term of Equation (6) is the cost of matching two features. Clearly, as the probability of occlusion $(1 - P_D)$ becomes small the cost of not matching a feature increases, as expected.

2.1 Dynamic Programming Solution

The minimization of Equation (6) is a classical weighted matching or assignment problem [28]. There exist well known algorithms for solving this with polynomial complexity $O(N^3)$ [28]. If the assignment problem is applied to the stereo matching problem directly, non-physical solutions are obtained. This is because Equation (6) does not constrain a match at $\mathbf{z}_{i_s}$ to be close to the match for $\mathbf{z}_{(i-1)_s}$, yet surfaces are usually smooth, except at depth discontinuities. In order to impose this smoothness condition, previous researchers have included a disparity penalty to their cost function [6, 23, 31, 32, 35]. The problem with this approach is that it tends to blur the depth discontinuities as well as introduce additional free parameters that must be adjusted.

Instead, we make the common assumptions [27] of:

1. *uniqueness*, i.e. a feature in the left image can match to no more than one feature in the right image and vice versa and

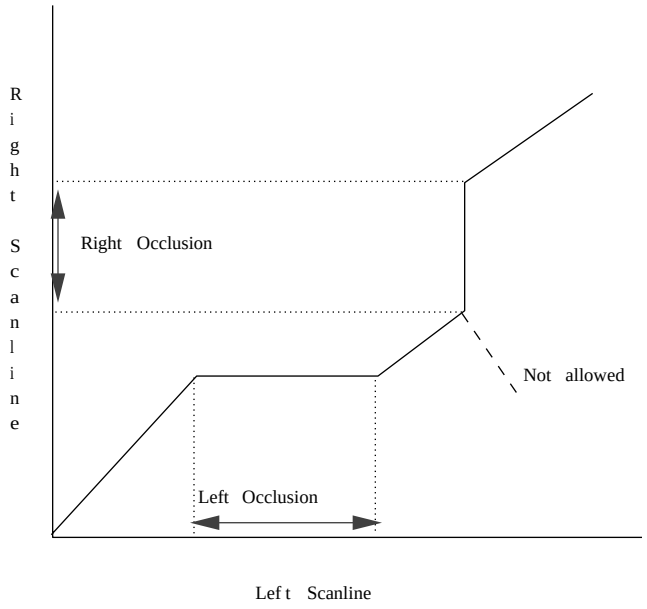

Figure 1: A path representing a matching of points in the left and right images. The solid line represents a legal set of matches. The dashed path violates the uniqueness and ordering constraints.

2. *monotonic ordering*, i.e. if $\mathbf{z}_{i_1}$ is matched to $\mathbf{z}_{i_2}$ then the subsequent measurement $\mathbf{z}_{i_1+1}$ may only match measurements $\mathbf{z}_{i_2+j}$ for which $j > 0$.

These constraints are illustrated in Figure (1) where a path representing a matching of points in the left and right images. The solid line represents a legal set of matches while the dashed path is an illegal set of matches since they violate the uniqueness and ordering constraints. A horizontal line segment denotes occlusion in the left image; a vertical line segment denotes occlusion in the right image; a diagonal line segment to a point (i, j) denotes the matching of the left feature i with the right feature j.

The minimization of Equation (6) subject to these constraints can be solved by dynamic programming in $O(NM)$, where N and M are the number of measurements in each of the two epipolar lines, as outlined in Figure (2). Reconstruction of the optimum path then proceeds as outlined in Figure (3) where $C(i, j)$ represents the cost of matching the first i features in the left image with the first j features in the right image and $\mathrm{c}(\mathbf{z}_{1,i}, \mathbf{z}_{2,j})$ is the cost of matching the two features $\mathbf{z}_{1,i}, \mathbf{z}_{2,j}$ as shown in Equation (6).

Of course, this general solution can be further improved by realizing that there is a practical limit to the disparity between two measurements. This is also true for human stereo, the region of allowable disparity being referred to as Panum's fusional area [22]. If a measurement $\mathbf{z}_{i_1}$ is constrained to match only measurements $\mathbf{z}_{i_2}$ for which[9] $(i_1 - \Delta x) \leq$

[9]This assumes that the left image is i_2 and therefore shifted to the right relative to the right image i_1.

```
Occlusion = [ln( (P_D/(1-P_D)) * φ/|(2π)^d S_s^{-1}|^{1/2} )]
for (i=1;i≤ N;i++){ C(i,0) = i*Occlusion }
for (i=1;i≤ M;i++) { C(0,i) = i*Occlusion}
for(i=1;i≤ N;i++){
   for(j=1;j≤ M;j++){
      min1 = C(i-1,j-1)+c(z_{1,i},z_{2,j});
      min2 = C(i-1,j)+Occlusion;
      min3 = C(i,j-1)+Occlusion;
      C(i,j) = cmin = min(min1,min2,min3);
      if(min1==cmin) M(i,j) = 1;
      if(min2==cmin) M(i,j) = 2;
      if(min3==cmin) M(i,j) = 3;
   }}
```

Figure 2: Pseudo-code describing how to calculate the optimum match.

```
p=N;
q=M;
while(p!=0 && q!=0){
   switch(M(p,q)){
      case 1:
         p matches q
         p--;q--;
         break;
      case 2:
         p is unmatched
         p--;
         break;
      case 3:
         q is unmatched
         q--;
         break;
   }}
```

Figure 3: Pseudo-code describing how to reconstruct the optimum match.

$i_2 \leq i_1$ then the time required by dynamic programming algorithm can be reduced to linear complexity $O(N\Delta x)$.

3 Experimental Results - synthetic data

The preceding theoretical development is independent of the actual matching primitives used in a particular implementation. For the experiments described below, the feature primitives were the scalar intensity values of the individual pixels. There are several advantages with working directly on the pixel intensities. First, problems associated with feature extraction and/or with adaptively sized windows common to area-based correlation methods are completely avoided. Second, the intensity based method provides a *dense* disparity map, in contrast to the sparse maps of feature based approaches. This also eliminates the need for sophisticated interpolation techniques.

Unless otherwise stated, all experiments described here were performed with scalar measurement vectors representing the intensity values of the individual pixels, i.e. $\mathbf{z}_{i_s} = I_{i_s}$. The field of view of each camera, ϕ_s, is assumed to be π and the measurements are assumed to be corrupted with white noise of variance $\sigma^2 = 4$. Finally, the probability of detection P_D is assumed to be 0.99.

3.1 Random Dot Stereograms

Figure (5) shows the disparity map (with reference to the left image) obtained for a random dot stereogram consisting of three rectangular regions one above the other. The left random dot stereogram is shown in Figure (4) where approximately 50% of the points are black and the rest white. The black pixel values in the disparity map indicate points in the left image that were considered to be occluded in the right image. While the number of correct matches is 95.4%, it is interesting to examine why the correct depth estimates have not been found at every point on every line. In particular, since the RDS pair is noise free, a perfect match is expected, so the right side of each rectangle should exhibit a depth discontinuity that is aligned with neighboring scanlines. This is not the case in practice. Close examination of this phenomenon revealed there are multiple *global* minima! Dynamic programming is guaranteed to find a global minima but not necessarily the same one for each scanline. Hence, the misalignment of the vertical depth discontinuities.

Figure (6) illustrates how these global minima arise. Investigation of this phenomenon revealed that there are many such alternative matchings. This problem is addressed next.

4 Cohesivity constraints

When more than one global minimum exists, i.e. multiple solutions paths exist, the algorithm arbitrarily chooses one of these paths, resulting in small variations between lines. The arbitrariness arises within the dynamic programming algorithm during the

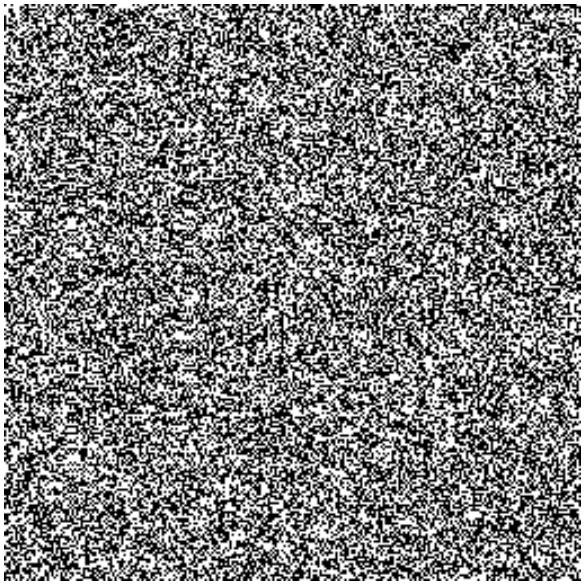

Figure 4: Left view of a random dot stereogram.

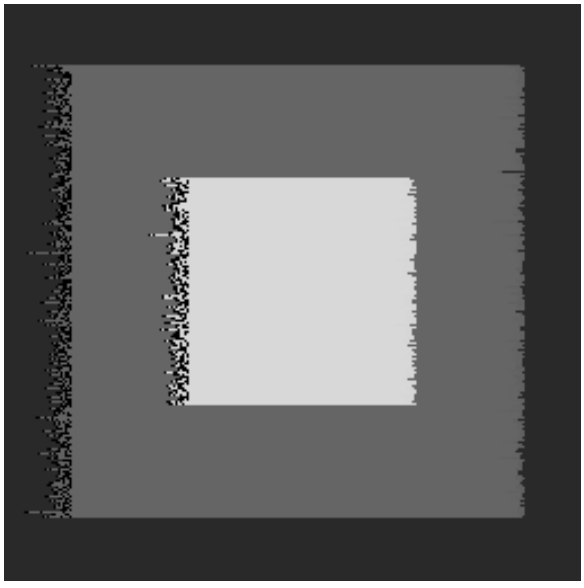

Figure 5: Disparity map obtained with $P_D = 0.99$.

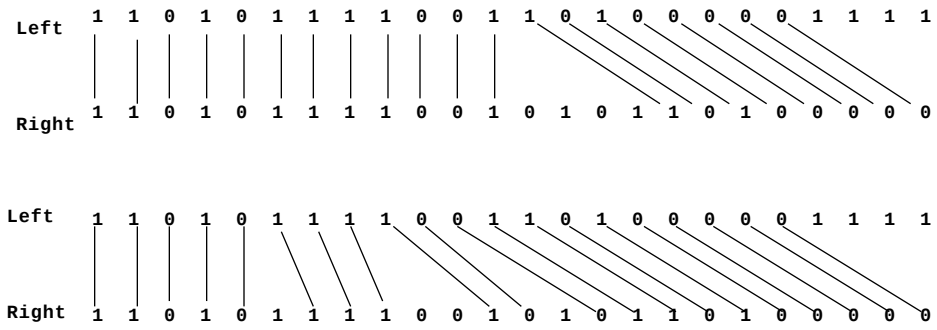

Figure 6: Two alternative matches both of equal cost found in the neigborhood of a discontinuity

test for the minimum cost path,
`C(i,j) = min(C(i-1,j-1)+c(i,j),C(i-1,j)+Occlusion,C(i,j-1)+Occlusion)`,
i.e., in deciding whether to take the horizontal, vertical or diagonal path to (i, j). This decision becomes arbitrary if more than one path is a minimum.

A more rational decision is to choose the "smoothest" path that minimizes the cost function $C(i, j)$. There are many ways to define "smoothest". Previous stereo algorithms have incorporated a smoothing term into the cost function $C(i, j)$ based on the difference in disparity between neighboring pixels. This regularization term [30] penalizes large disparity changes arguing that most surfaces are smooth. However, surfaces are not smooth at depth discontinuities which are the most important features of depth maps.[10] Yuille *et al* [35], Geiger *et al* [15] and Belhumeur and Mumford [5] have addressed the problem of smoothing with discontinuities within the framework of regularization. Here, we propose an alternative method.

Regularization methods are usually employed to restrict the class of admissible solutions. There are two approaches to finding $\mathbf{z}$ given data $\mathbf{y}$, and $\mathbf{Az} = \mathbf{y}$:

1. The first is to find the $\mathbf{z}$, from among the $\mathbf{z}$ that satisfy

$$\| \mathbf{Az} - \mathbf{y} \|^2 \leq \epsilon \tag{7}$$

[10] Although not regularization in the mathematical sense, Ohta and Kanade [27] also incorporate an inter-scanline continuity cost into their cost function.

that minimizes $\| \mathbf{Pz} \|^2$, a stabilizing functional.

2. The second method is to minimize

$$\| \mathbf{Az - y} \|^2 + \lambda \| \mathbf{Pz} \|^2 \tag{8}$$

where λ is a regularization term.

The second method has been widely used within the computer vision community. However, determining the "optimum" value for λ can be difficult. Since λ controls the degree of smoothing, then if λ is too large the resulting disparity map is too smooth while if λ is too small the disparity map is too noisy. The introduction of line processes have significantly improved quality of results obtained from regularization. Nevertheless, there is no guarantee that the solution to Equation (8) will minimize the original cost function of Equation (7).

While Poggio *at al* [30] outlined method 1, the authors are unaware of any application of this approach, perhaps because it is unclear how to efficiently compute such a minimization. Solutions, via this method are guaranteed to be within ϵ of the original cost function. We now describe how solutions can be found via method 1 within the framework of dynamic programming.

Instead of incorporating a smoothing term into the cost function $C(i, j)$, a second optimization can be performed that selects from the set of solutions that minimize $C(N, M)$, that solution which contains the least number of discontinuities. Performing this minimization *after* first finding all maximum likelihood solutions is very different from incorporating the discontinuity penalty into the original cost.

Smoothness can be defined in a number of ways. This paper considers definitions that minimize

1. the number of horizontal discontinuities along a scanline or

2. the total number of horizontal and vertical discontinuities along and across scanlines respectively.

Other definitions were examined, including

1. the number of vertical discontinuities across scanlines or

2. the number of horizontal discontinuities *then* the number of vertical discontinuities.

3. the number of vertical discontinuities *then* the number of horizontal discontinuities.

but were found to be inferior to the first two.

Minimizing the total number of horizontal discontinuities can be accomplished as part of the dynamic programming algorithm without having to enumerate all the maximum likelihood solutions, and is outlined below.

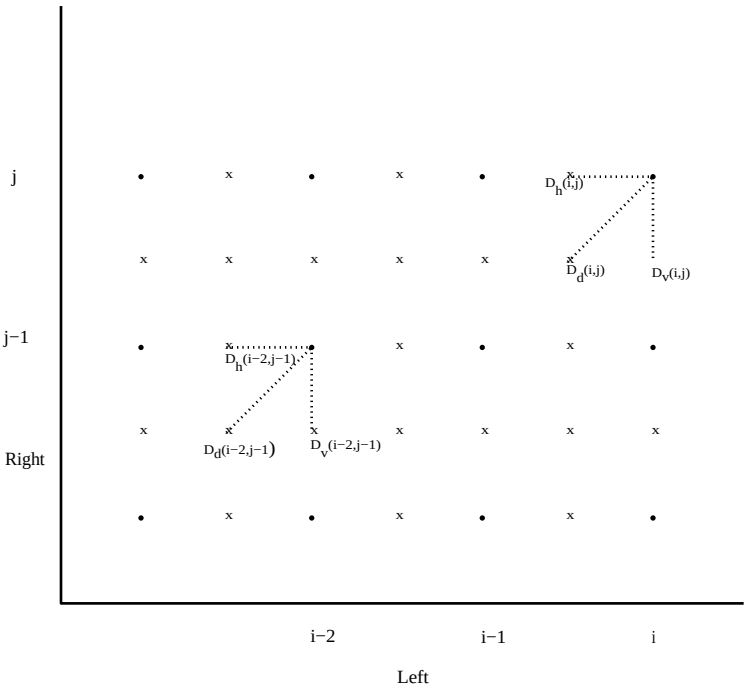

Figure 7: Illustration representing the storage of the additional information $D_v(i,j)$, $D_h(i,j)$, $D_d(i,j)$.

4.1 Maximum likelihood, minimum horizontal discontinuities

In order to minimize the number of horizontal discontinuities, it is necessary to record how the algorithm arrived at (i,j). This is done using three matrices labelled D_v`(i,j)`, D_h`(i,j)`, D_d`(i,j)`, denoting whether (i,j) was arrived at from a horizontal, vertical or diagonal path. The $D(i,j)$ matrices record the total number of horizontal discontinuities in the matching of the first i points in the left image with the first j points in the right image. This is depicted in Figure (7). The information stored in the $D(i,j)$ matrices can then be used to break any ties that occur in the calculation of $C(i,j)$. Algorithmically, this is accomplished as outlined in Figure (8). Notice that the `M(i,j)` matrix used for reconstruction is no longer used. Instead, the optimum path can be reconstructed directly from the `D(i,j)` matrices as outlined in Figure (9). Minimizing the number of horizontal discontinuities has the advantage that each scanline of the image can be solved independently and therefore in parallel.

The result of applying the maximum likelihood minimum horizontal (MLMH) discontinuity algorithm to the random dot stereogram is shown in Figure (10). A significant improvement is evident, with the percentage of correct matches increasing to 98.7%. Once again, imperfect matching indicates the existence of multiple global minima, but their number is far fewer.

```
Occlusion = [ln(P_Ds/(1-P_Ds) * 1/|(2π)^d S_s^-1|^1/2)]
for (i=1;i≤ N;i++){ C(i,0) = i*Occlusion }
for (i=1;i≤ M;i++) { C(0,i) = i*Occlusion}
for(i=1;i≤ N;i++){
   for(j=1;j≤ M;j++){
      min1 = C(i-1,j-1)+c(z_1,i ,z_2,j);
      min2 = C(i-1,j)+Occlusion;
      min3 = C(i,j-1)+Occlusion;
      C(i,j) = cmin = min (min1,min2,min3);
      if(min1==cmin)
         D_d(i,j) = imin(D_d(i-1,j-1), D_h(i-1,j-1)+1,D_v(i-1,j-1)+1);
      else
         D_d(i,j) = HUGE;
      if(min2==cmin)
         D_h(i,j) = imin(D_d(i-1,j)+1, D_h(i-1,j),D_v(i-1,j)+1);
      else
         D_h(i,j) = HUGE;
      if(min3==cmin)
         D_v(i,j) = imin(D_d(i,j-1)+1, D_h(i,j-1)+1,D_v(i,j-1));
      else
         D_v(i,j) = HUGE;
   }}
```

Figure 8: Maximum likelihood, minimum horizontal discontinuities (MLMH) algorithm. Note than `min()` returns the minimum value of its arguments while `imin()` returns the index, (1, 2, or 3) to the minimum value. The matrices D_d, D_h and D_v record whether the current position (i, j) was arrived at via a diagonal, horizontal or vertical move - see text for details.

4.2 Maximum likelihood, minimum horizontal plus vertical discontinuities

Clearly, minimizing the total number of horizontal and vertical discontinuties will result in a perfect solution to the random dot stereogram of Figure (4). Unfortunately, minimizing vertical discontinuities between epipolar lines cannot be performed by dynamic programming. Though it is conceptually straightforward to perform the vertical minimization, i.e. find all global maxiumum likelihood solutions for each epipolar line and then choose one solution per line to minimize the sum of the vertical dicontinutites between lines, in practice it is relatively costly. Consequently, we have implemented an approximation to this, still based on dynamic programming, that minimizes the *local* discontinuties between adjacent epipolar lines.

The MLMH+V algorithm can be computed in one of two ways. Either one can compute the solution to the previous line and then minimize the number of vertical discon-

```
/* Find start point and initialize costs d1, d2 and d3 together with the index counters e and f.
 These are different for each of the three cases. */
p = NL;
q = NR;
switch(imin(D_d[p,q],D_v[p,q],D_h[p,q])){
   case 1:
      p matches q;
      d1=0; d2=1; d3=1;
      e=1; f=1;
      break;
   case 2:
      p is unmatched;
      d1=1; d2=0; d3=1;
      e=1; f=0;
      break;
   case 3:
      q is unmatched;
      d1=1; d2=1; d3=0;
      e=0; f=1;
      break;
   }
/*
 * Now begin reconstruction.
 */
while(p> 0 && q> 0){
   switch(imin(D_d[(p-e),q-f]+d1,D_v[(p-e),q-f]+d2,D_h[(p-e),q-f]+d3)){
      case 1:
         (p-e) matches (q-f);
         d1=0; d2=1; d3=1;
         p=p-e; q=q-f;
         e=1; f=1;
         break;
      case 2:
         (p-e) is unmatched;
         d1=1; d2=0; d3=1;
         p=p-e; q=q-f;
         e=1; f=0;
         break;
      case 3:
         (q-f) is unmatched;
         d1=1; d2=1; d3=0;
         p=p-e; q=q-f;
         e=0; f=1;
         break;
}}
```

Figure 9: Reconstructing the MLMH solution.

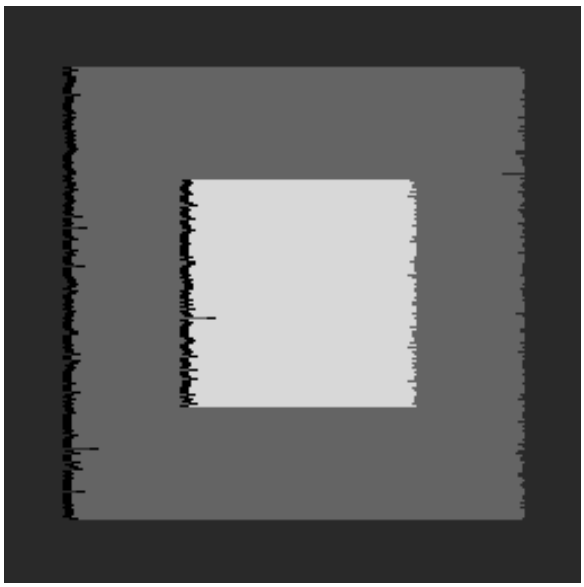

Figure 10: Disparity map for maximum likelihood minimum discontinuity

tinuities between the current line with the previous line. Or a two pass scheme can be employed, the first pass calculating the MLMH solution for initialization purposes. During the second pass, the each line is compared with the MLMH line above and below it to determine the number of vertical discontinuities. This method can then be iterated until convergence, though in practice, little if any improvement was noticable over just two passes.

On a sequential machine, the first method is faster, taking approximately 20% more time than the MLMH solution, while the second method takes twice as long. On synthetic data such as the ellipsoid of Figure (32), the sequential bottom-to-top processing also introduced some artifacts, e.g. extending vertical edges beyond their termination points, although this did not appear to be the case for natural scenes. A parallel implementation should avoid the first method since this is very sequential. However, the second method is highly parallelizable.

Figure (11) outlines the MLMH+V algorithm. The function `IsMatched(i)` checks to see if there is also a match in the previous line (or in the line above and line below in the two pass method). Similarly, the functions `IsLeftOcclusion(i)` and `IsRightOcclusion(i)` check whether there is an occlusion in the left or right image at the point i.

Figure (12) shows the MLMH+V solution. It is clear that the solution is suboptimal, since the vertical depth discontinuity is not perfectly straight. Nevertheless, there is still a further improvement in the solution, the percentage accuracy increasing to 99.1%.

```
for(i=1;i≤NL;i++){
   for(j=1;j≤ML;j++){
/*
 * Find minumum cost match
 * (i) C(i-1,j-1) + c(i,j) or (ii) C(i-1,j) + Occlusion or (iii) C(i,j-1) + Oclussion
 */
      cost1 = C(i-1,j-1) + c(z_{1,i},z_{2,j});
      cost2 = C(i-1,j) + Occlusion;
      cost3 = C(i,j-1) + Occlusion;
      cmin = min(cost1, cost2, cost3);
/*
 * In case of tie, e.g. cost1==cost2, then choice of match is chosen
 * to minimize the number of horizontal and vertical discontinuities
 * The variable sum indicates whether a match occurred at (i-1,j-1)
 * (i,j) or (i+1,j+1).
 * The DV matrices count the number of horizontal and vertical
 * discontinuities
 */
      if(cmin==cost1){
         vcost = IsMatched(i);
         DV_d(i,j)=min(DV_d(i-1,j-1),DV_v(i-1,j-1)+1, DV_h(i-1,j-1)+1)+vcost;
      }
      else{
/* if not minimum then can't use this path so set infinite  */
         DV_d(i,j)=HUGE;
      }
      if(cmin==cost2){
         vcost = IsLeftOcclusion(i);
         DV_v(i,j) = min(DV_d(i-1,j)+1, DV_v(i-1,j), DV_h(i-1,j))+vcost;
      }
      else{
         DV_v(i,j)=HUGE;
      }
      if(cmin==cost3){
         vcost = IsRightOcclusion(i);
         DV_h(i,j) = min(DV_d(i,j-1)+1, DV_v(i,j-1), DV_h(i,j-1))+vcost;
      }
      else{
         DV_h(i,j)=HUGE;
      }
/* Finally record minimum cost and increment pointers */
      C(i,j)=cmin;
   }}
```

Figure 11: Maximum likelihood, minimum horizontal plus vertical discontinuities, (MLMH+V), algorithm

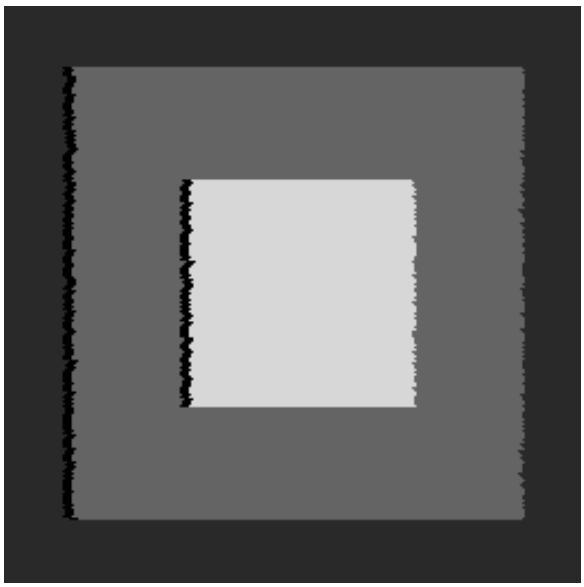

Figure 12: RDS Solution using MLMH+V

5 Experiments with cohesivity constraint

In this section we demonstrate the performance of the ML, MLMH and MLMH+V algorithms on a variety of natural and synthetic images. The disparity maps show the disparities for the left image where, for display purposes, unmatched pixels, i.e. those that are occluded in the right image, are assigned the disparity value of whichever of the left or right neigboring pixel is furthest away.

Application of the MLMH and MLMH+V algorithms to natural scenes requires a slight modification to the algorithms of Section (4). The problem is that the noise present in real images may be sufficient to affect the maximum likelihood solutions so that only a single global minimum exists. Other minima exist nearby that have costs close to the minimum, but the noise has randomly biased one of the solutions to be a minimum. This problem can be alleviated by altering the dynamic programming algorithms to test for "approximate equality" rather than exact equality. Unfortunately, this introduces an additional free parameter, ϵ. In practice, we have found that setting ϵ to $(0.5\ \texttt{Occlusion}) \leq \epsilon \leq (0.9\ \texttt{Occlusion})$ works well and that the solutions are stable to variations in ϵ. The cost of solutions found using this modification are typically within 5% of the global minimum value. Since no ground truth information is known, only a qualitative evaluation can be made.

5.1 The "Parking meter"

Figures (14 - 16) show the results of applying the algorithms to a stereo pair, the left image of which is shown in Figure (13). The ML solution, Figure (14) has considerable

streaking especially around the border of the car and around the narrow sign pole in the middle right of the image. The MLMH solution of Figure (15) significantly reduces this streaking and the MHMH+V solution almost eliminates it, much of the remaining streaking being due to quantization error which is unavoidable unless subpixel features are used. Especially noteworthy is the narrow sign pole in the middle right of Figure (16) which illustrates the sharp depth discontinuities that can be obtained with the algorithm.

Figure 13: Left image of the "Parking meter" stereo pair, courtesy of T. Kanade and T. Nakahara of CMU.

5.2 The Pentagon

Figure (17) is the left image of the "Pentagon" stereogram. Figures (18 - 20) shows the resulting disparity maps for the ML, MLMH and MLMH+V algorithms. Surprisingly, there is little or no improvement between the three algorithms. However, significant detail is obtained, as is evident from the overpasses and freeways in the upper right corner of the disparity maps. These disparity maps are comparable to results of Geiger *et al* [15] and Cochran and Medioni [9].

5.3 The "Shrub" and image normalization

Figures (22 - 24) show the results of applying the algorithms to a stereo pair called "Shrub", the left image of which is shown in Figure (21). Although coarse depth information is obtained, the disparity maps are poor with many artifacts present. Investigation of this problem revealed that the left and right image pair violated the Normal

Figure 14: ML disparity map for the "Parking meter".

Figure 15: MLMH disparity map for the "Parking meter".

Figure 16: MLMH+V disparity map for the “Parking meter”.

Figure 17: The Pentagon

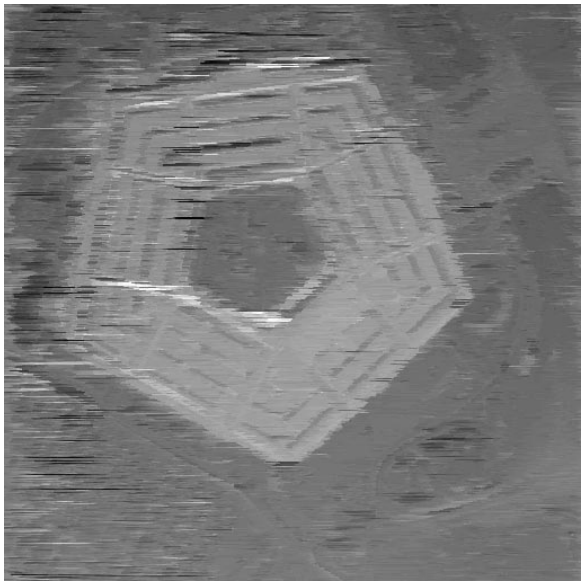

Figure 18: Maximum likelihood disparity map for the Pentagon.

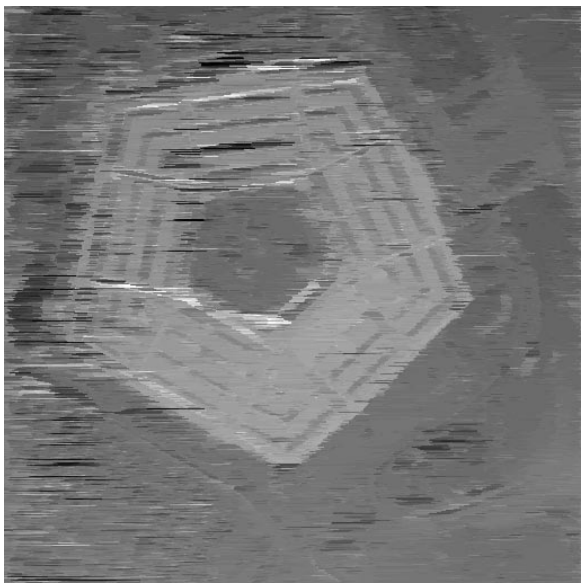

Figure 19: MLMH disparity map for the Pentagon.

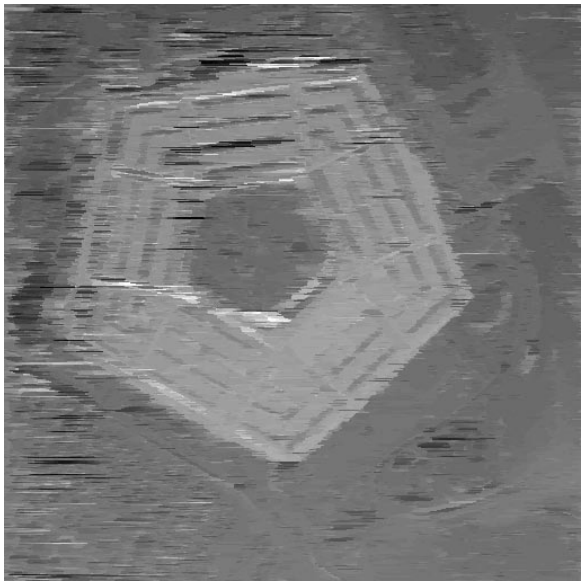

Figure 20: The MLMH+V disparity map for the Pentagon.

Figure 21: Left image of the "Shrub" stereo pair, courtesy of T. Kanade and T. Nakahara of CMU.

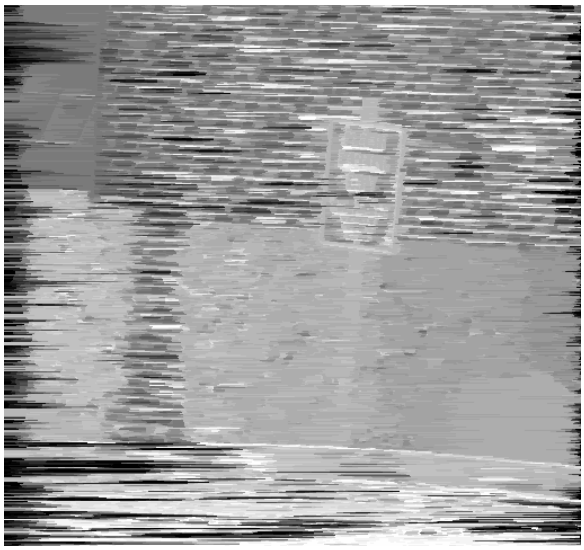

Figure 22: ML disparity map for the “Shrub”.

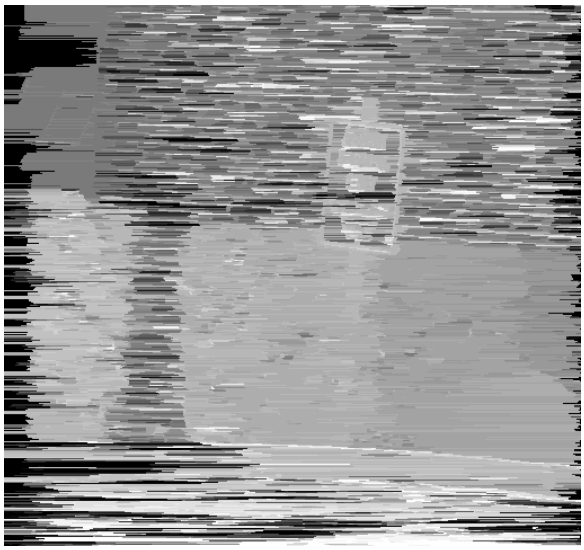

Figure 23: MLMH disparity map for the “Shrub”..

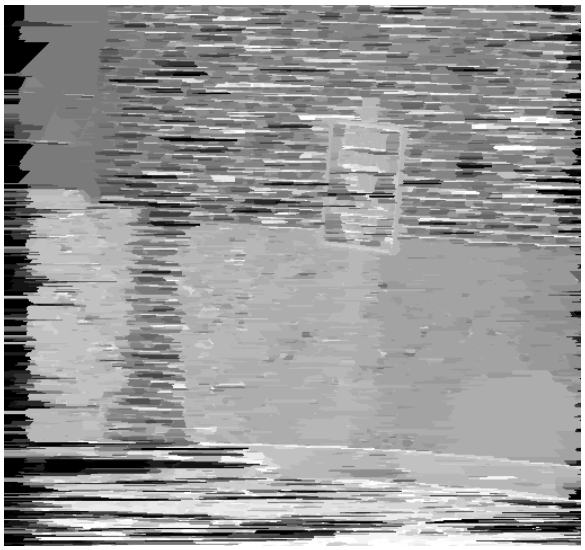

Figure 24: MLMH+V disparity map for the "Shrub".

distribution assumption used to compare corresponding intensity values. In particular, careful examination of the intensity values at corresponding points revealed significant non-zero biases. It is suspected that between the time of taking the left and right images, illumination conditions changed; perhaps a cloud obscured the sun. It was decided to model this variation in illumination by constant additive and multiplicative factors,[11] i.e.

$$I_l(x, y) = AI_r(x, y) + B \tag{9}$$

This relationship includes a constant multiplicative term, A, together with the more common additive term, B. The commonly used Laplacian operator would remove the additive bias, B, but not the multiplicative term. In a seperate study of the validity of the constant image brightness assumption for the JISCT stereo database [7, 12], we found that in almost one in three images the assumption was invalid. Moreover, a simple additive bias did not adequately model the relationship between corresponding left and right intensities. The model of Equation (9) was found to be sufficient though a nonlinear model of the form

$$I_l(x, y) = AI_r^C(x, y) + B$$

is most accurate.

[11] Gennert [16] showed that the intensities of two corresponding points are approximately related by a spatially varying multiplicative factor. Gennert found that an additive term was unnecessary. However, Gennert's multiplicative relationship models how the intensities of two corresponding points varies due to surface orientation and reflectance models. We have assumed this relationship to be a Normal distribution. Rather, our constant additive and multiplicative constants attempt to model changes in illumination conditions and differences in camera responses.

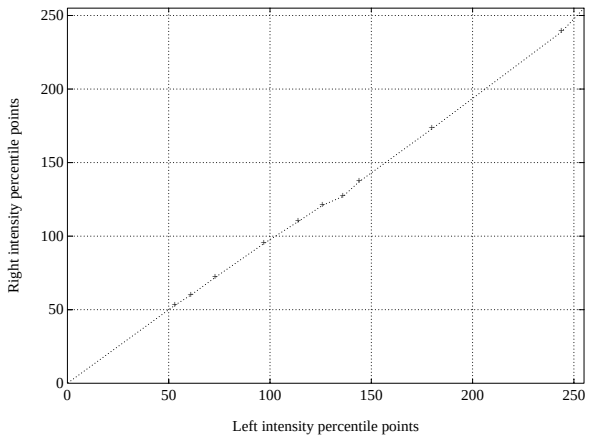

Figure 25: Left versus Right plot of ten percentile points from intensity histograms.

If the number of occluded points is small compared to the overall number of pixels, then the intensity histograms for the left and right images are approximately the same except for the fixed offset B and the scaling term A. Estimation of the constants A and B was performed by first calculating the intensity histograms for both the left and right image. Then plotting the ten percentile points as depicted in Figure (25). A linear regression can then be performed on these points, the slope and intercept providing estimates for A and B respectively. In practice, we performed piecewise linear approximation between the ten percentile points.

If the ML, MLMH and MLMH+V algorithms are applied to the normalized images significantly better results are obtained, as is evident from Figures (26-28). Again, streaking is evident, especially on the surface of the brick wall. The MLMH algorithm reduces this streaking and a further reduction is obtained from the MLMH+V algorithm. However, a small number of artifacts is still present on the brick wall and on the horizontal edges of the sign.

5.4 Face

Figures (29) and (30) show the left and right views of the bust of a face. This pair of images was created by rotating the bust through 10°. Figure (31) shows the disparity map obtained using the MLMH+V algorithm. Qualitatively good results are obtained.

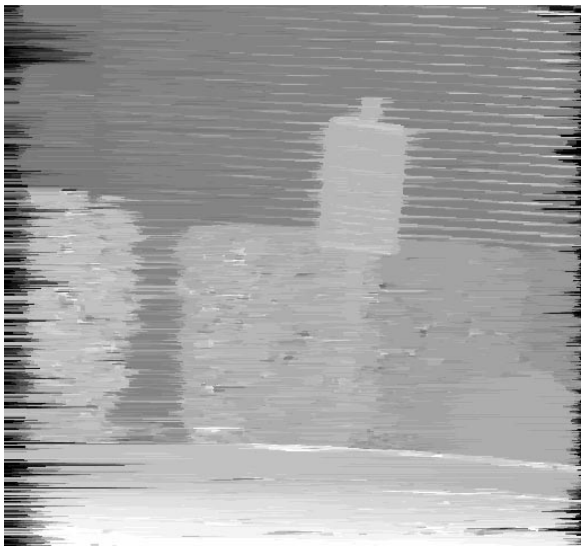

Figure 26: ML disparity map for the "Shrub" after normalization.

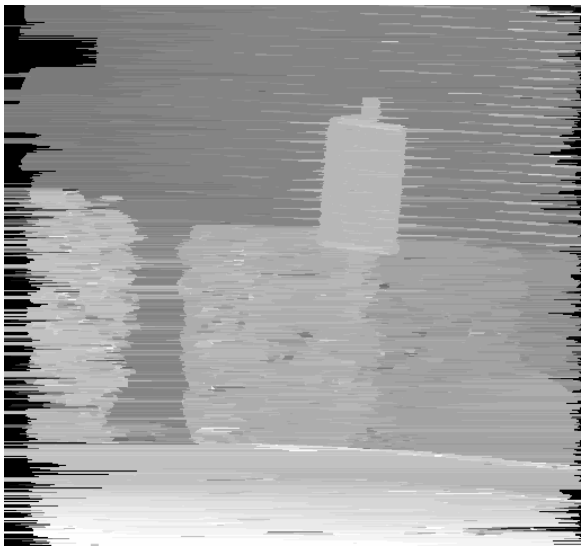

Figure 27: MLMH disparity map for the "Shrub" after normalization.

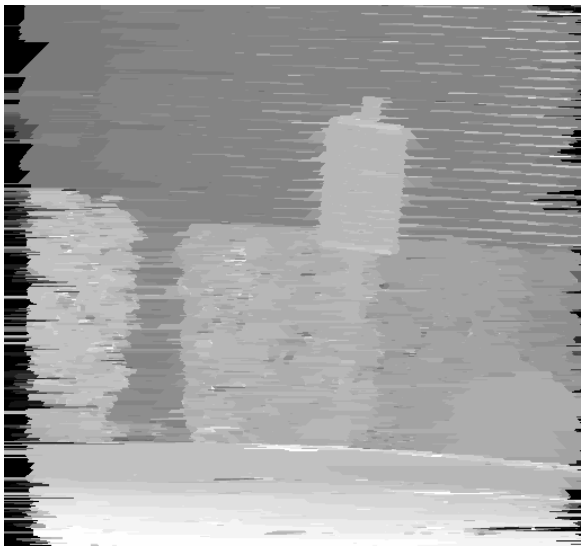

Figure 28: MLMH+V disparity map for the "Shrub" after normalization.

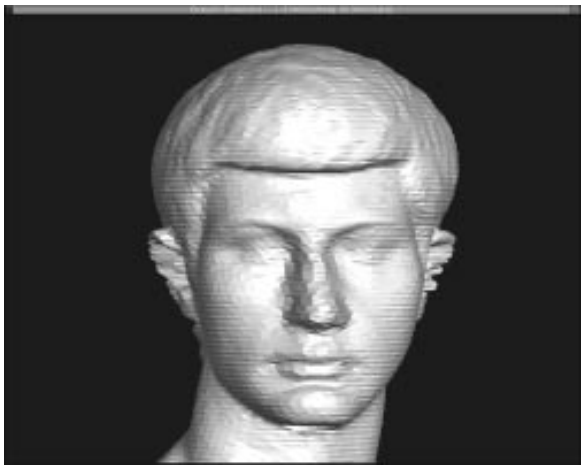

Figure 29: Left image of the "Face", courtesy of J. Tajima and S. Sakamoto of NEC.

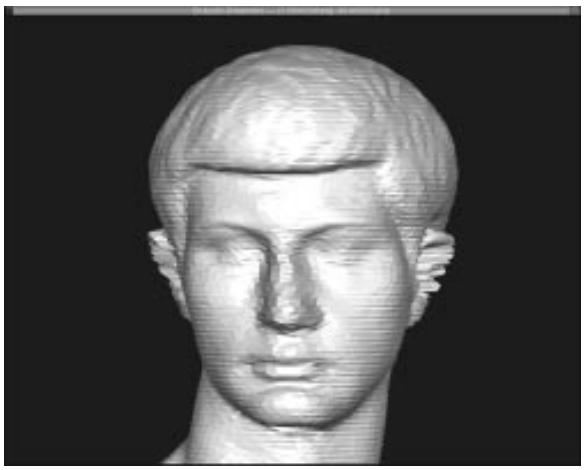

Figure 30: Right image of the "Face", courtesy of J. Tajima and S. Sakamoto of NEC.

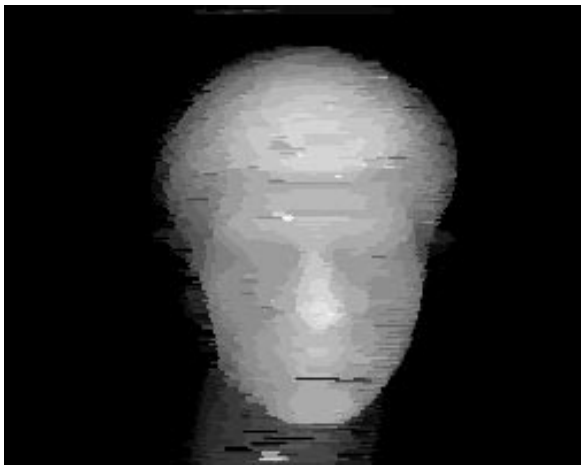

Figure 31: MLMH+V disparity map for the "Face".

5.5 Ellipsoid - Shape from disparate shading

Figure (32) and Figure (33) show the left and right images of an ellipsoid. The ellipsoid has been synthetically generated such that no zero crossings occur on its surface[8]. As such, edge based stereo algorithms are incapable of estimating depth variations over the ellipsoid's surface. Bulthoff [8] has called this class of images "intensity-based stereo". Figure (34) shows the result of applying the MLMH+V algorithm to the stereo pair. Note that a significant amount of 3D structure is recovered despite the absence of edges.

Bulthoff [8] has shown that humans are able to determine depth in the absence of edges and has therefore conjectured that there may be a seperate mechanism for intensity-based stereo. The fact that humans are better able to estimate depth when edges are present is taken as evidence that an edge-based stereo mechanism must also be present. However, the performance of intensity-based stereo algorithm described here is also improved when texture is added to the ellipsoid, even though edges are not explicitly extracted. Instead, the edges or texture simply help to reduce the ambiguity present when matching the two intensity signals. Could a single intensity-based mechanism account for both edge-based and intensity-based human performance?

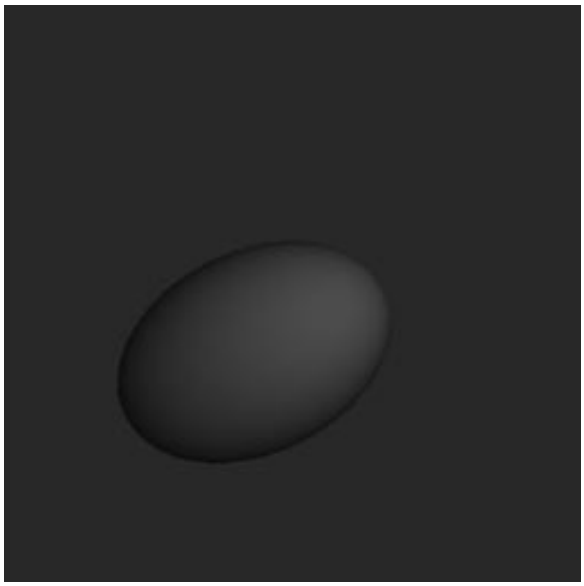

Figure 32: Left image of the "Ellipsoid".

6 The Modified Cost Function for N-Cameras

The previous section demonstrated how the cohesivity constraints defined in Section (4) can be applied to resolve ambiguous correspondences. An alternative to imposing cohesivity constraints to resolve ambiguous correspondences is to use more than two cameras.

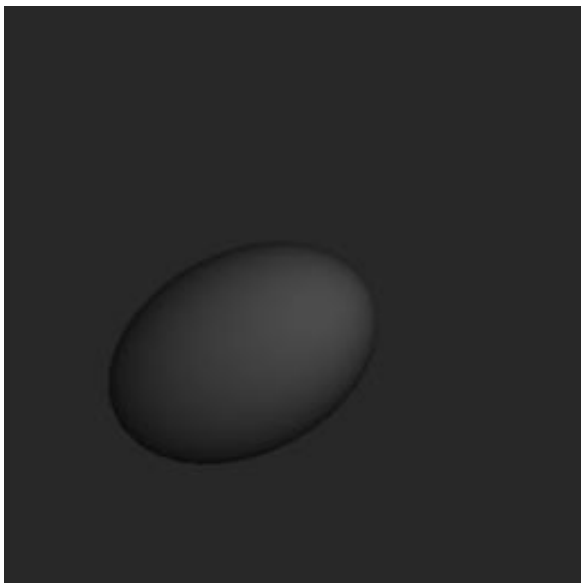

Figure 33: Right image of the "Ellipsoid".

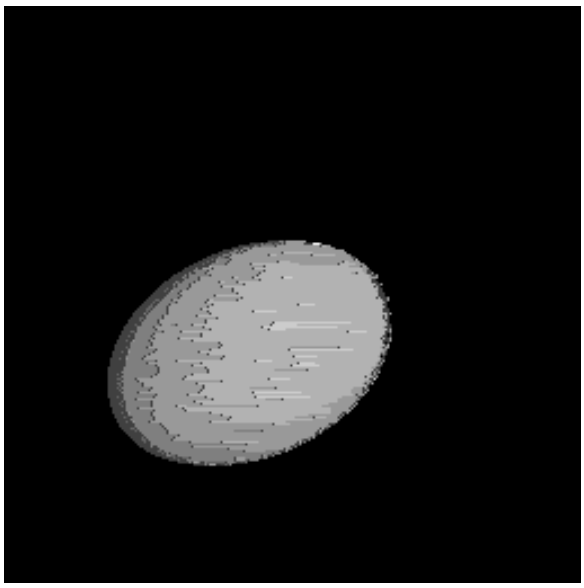

Figure 34: MLMH+V disparity map for the "Ellipsoid".

Here, we extend the two camera maximum likelihood to N-cameras. The N-camera stereo algorithm determines the "best" set of correspondences between a given pair of cameras, referred to as the *principal* cameras. Knowledge of the relative positions of the cameras allows the 3D point hypothesized by an assumed correspondence of two features in the principal pair to be projected onto the image plane of the remaining $N-2$ cameras. These $N-2$ points are then used to verify proposed matches.

Let the two principal cameras be denoted by $s = \{1, N\}$ and the set of intermediate cameras by $s = \{2, \ldots, N-1\}$. The two principal cameras may also be referred to as the left and right cameras though, in practice, the pair may have an arbitrary relative geometry. The condition that measurement $\mathbf{z}_{i_1}$ from camera 1, (the left camera) and measurement $\mathbf{z}_{i_N}$ from camera N, (the right camera), originate from the same location, $\mathbf{X}_k$, in space, i.e. that $\mathbf{z}_{i_1}$ and $\mathbf{z}_{i_N}$ correspond to each other is denoted by Z_{i_1,i_N}.

The likelihood that the measurement pair Z_{i_1,i_N} originated from the same point $\mathbf{X}_k$ is denoted by $\Lambda(Z_{i_1,\ldots,i_N} \mid \mathbf{X}_k)$ and is defined as

$$\begin{aligned}\Lambda(Z_{i_1,\ldots,i_N} \mid \mathbf{X}_k) &= \left(\frac{(N-1)(1-P_D)}{\phi}\right)^{\delta_{i_1 i_N}} \times \\ &\quad [P_D p(\mathbf{z}_{i_1} \mid \mathbf{X}_k) \times P_D p(\mathbf{z}_{i_N} \mid \mathbf{x}_k) \times \\ &\quad \prod_{s=2}^{N-1} (P_D p(z_{i_s} | \mathbf{X}_k))^{1-\delta_s} \left(\frac{1-P_D}{\phi}\right)^{\delta_s}\Bigg]^{1-\delta_{i_1 i_N}} \end{aligned} \tag{10}$$

where $\delta_{i_1 i_N}$ is an indicator variable that is unity if measurement $\mathbf{z}_{i_1}$ in the left image or $\mathbf{z}_{i_N}$ in the right image, is not assigned a corresponding point, i.e. is occluded. Otherwise $\delta_{i_1 i_N} = 0$ indicates that measurement $\mathbf{z}_{i_1}$ corresponds to $\mathbf{z}_{i_N}$. The indicator variable δ_s is unity if the 3D point hypothesized by the match $\mathbf{Z}_{i_1,i_N}$ is occluded in camera s, and zero otherwise. Note that the features $\{\mathbf{z}_{i_2}, \ldots, \mathbf{z}_{i_{N-1}}\}$ are determined by projecting the 3D point hypothesized by the correspondence of $\mathbf{z}_{i_1}$ and $\mathbf{z}_{i_N}$ onto the image planes of each of the $N-2$ cameras. This is straightforward provided the relative position of the cameras is known (see [33] for details).

Now that we have established the cost of the individual pairings Z_{i_1,i_N}, it is necessary to determine the total cost of all pairs. Let γ denote a feasible pairing of all measurements and let , be the set of all feasible partitions, i.e. , $= \{\gamma\}$. Then we wish to find the pairings (or partition), γ, that maximizes $L(\gamma)$ where the likelihood $L(\gamma)$ of a partition is defined as

$$L(\gamma) = p(\mathbf{Z}_1, \mathbf{Z}_N \mid \gamma) = \prod_{Z_{i_1,\ldots,i_N} \epsilon \gamma} \Lambda(Z_{i_1,\ldots,i_N} \mid \mathbf{X}) \tag{11}$$

The maximization of $L(\gamma)$ is equivalent to

$$\min_{\gamma \epsilon \Gamma} J(\gamma) = \min_{\gamma \epsilon \Gamma} [-\ln(L(\gamma))] \tag{12}$$

which leads to

$$\min_{\gamma \epsilon \Gamma} J(\gamma) = \min_{\gamma \epsilon \Gamma} \sum_{Z_{i_1,\ldots,i_N} \epsilon \gamma} \delta_{i_1,i_N} \ln \left(\frac{P_D^2 \phi}{(N-1)(1-P_D) \mid (2\pi)^d \mathbf{S} \mid^{\frac{1}{2}}}\right) +$$

$$(1-\delta_{i_1,i_N})\left\{\frac{1}{4}(\mathbf{z}_{i_1}-\mathbf{z}_{i_N})'\mathbf{S}^{-1}(\mathbf{z}i_1-\mathbf{z}_{i_N})+\right.$$
$$\left.\sum_{s=2}^{N-1}(1-\delta_s)((\hat{\mathbf{z}}-\mathbf{z}_{i_s})'\mathbf{S}^{-1}(\hat{\mathbf{z}}-\mathbf{z}_{i_s})+\delta_s\ln\left(\frac{P_D\phi}{(1-P_D)\mid(2\pi)^d\mathbf{S}\mid^{\frac{1}{2}}}\right)\right\} \quad (13)$$

assuming that the covariances, $\mathbf{S}_{i_s}$, are equal.

The first term of the outer summation of Equation (13) is the cost associated with declaring a feature occluded. The other term is the cost of matching two features. This cost is the sum of the weighted squared errors of the left and right features $\mathbf{z}_{i_1}$ and $\mathbf{z}_{i_N}$ plus the sum over the intermediate views of either the squared error cost or the occlusion cost depending on whether the feature is visible or not in an intermediate view.

Equation (13) must be minimized subject to the common constraints of uniqueness and ordering (monotonicity). While straightforward to perform using dynamic programming, satisfying the uniqueness constraints for all of the features in all N views would be prohibitive. Instead, we choose to apply the uniqueness constraint only to the left and right image features, i.e. it is allowable for a feature in an intermediate view to support (match) more than one pair of correspondences between features in the left and right images. Equation (13) can then be written as

$$\begin{aligned}\min_{\gamma\epsilon\Gamma}J(\gamma) &= \min_{\gamma\epsilon\Gamma}\sum_{Z_{i_1,i_2}\epsilon\gamma}\delta_{i_1,i_N}\ln\left(\frac{P_D^2\phi}{(N-1)(1-P_D)\mid(2\pi)^d\mathbf{S}\mid^{\frac{1}{2}}}\right)+\\ &\quad(1-\delta i_1,i_N)\left\{\frac{1}{4}(\mathbf{z}i_1-\mathbf{z}_{i_N})'\mathbf{S}^{-1}(\mathbf{z}_{i_1}-\mathbf{z}_{i_N})+\right.\\ &\quad\left.\sum_{s=2}^{N-1}\min\left((\hat{\mathbf{z}}-\mathbf{z}_{i_s})'\mathbf{S}^{-1}(\hat{\mathbf{z}}-\mathbf{z}_{i_s}),\ln\left(\frac{P_D\phi}{(1-P_D)\mid(2\pi)^d\mathbf{S}\mid^{\frac{1}{2}}}\right)\right)\right\} \quad (14)\end{aligned}$$

Comparing Equation (14) with (6), we see that for $N=2$, the costs are identical. For, $N>2$, i.e. there are intermediate views, the cost of matching is appended by the last term in Equation (14). This `min` term provides support for the match, based on the first argument or returns the cost of occlusion if the feature is assumed to be occluded in the intermediate view.

7 N-Camera Experimental Results

The experimental results described here determine the correspondence between the two principal images by minimization of Equation (14) together with the cohesivity constraint of minimizing the sum of the horizontal and vertical discontinuities in the resulting disparity map, as described in Section (4). Once again, the measurements are individual pixel intensities with an assumed standard deviation of $\sigma=2.0$. A value of $P_D=0.9$ is used throughout. Two image sequences are examined; a horizontal sequence and a horizontal plus vertical sequence. In order to provide subpixel estimates of the disparity map, all images were interpolated by 5 times prior to stereo matching. The resulting desparity map was then decimated by 5 times to produce the results described next. Finally, all disparity maps have been histogram equalized for visualization.

7.1 Horizontal Sequence

Figure (35) shows the first and last frames of a horizontal motion sequence. Figure (36)

Figure 35: Leftmost and rightmost views of the shrub sequence. Images are courtesy of T. Kanade and E. Kawamura of CMU.

shows the disparity map obtained from the original maximum likelihood algorithm. Notice that the disparity map contains many spurious occluded points (represented by black pixels) and there is significant systematic error on the rear wall in the vicinity of the (approximately uniform intensity) mortar stripes. Nevertheless, significant detail is apparent; notice, for example, the small shrub in the bottom right.

Figure (37) shows the disparity map obtained with one additional image located at the midpoint between the stereo pair. The number of spurious "occlusions" is significantly reduced and the stripe errors markedly reduced.

Figure (38) shows the resulting disparity map for 7 images each equally spaced between camera 1 and N. It is clear that there is a significant improvement in the quality of the disparity map of Figure (38). Many of the spurious "occlusions" and much of the striping error have been eliminated. However, some small "block" artifacting is apparent due to the horizontal and vertical continuity constraints imposed by the algorithm. Nevertheless, Figure (38) has far fewer correspondence errors than the original two frame stereo solution of Figure (36).

For comparison puposes, Figure (39) shows the results of applying the algorithm assuming *no* occlusions in the intermediate views, i.e. the min term of Equation (14) is replaced by its first argument. A significant increase in the number of spurious occlusions is apparent. This illustrates the importance of the modelling the occlusion process in all views, not just the principal pair.

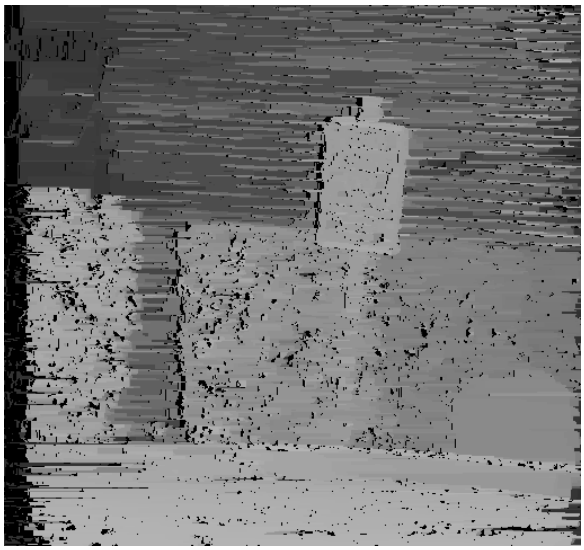

Figure 36: Disparity map obtained from leftmost and rightmost views.

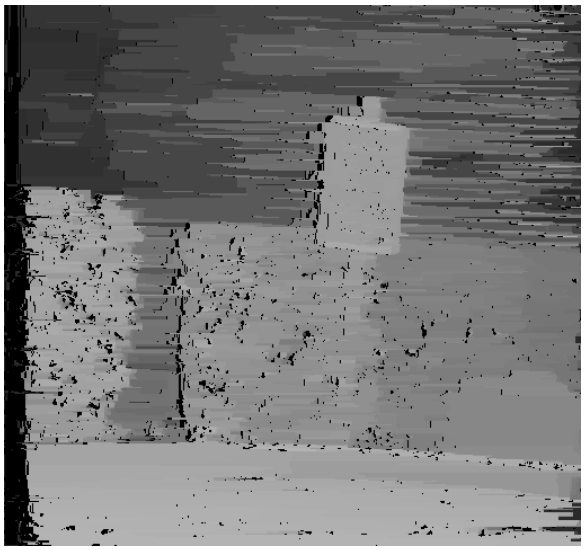

Figure 37: Disparity map obtained with three images.

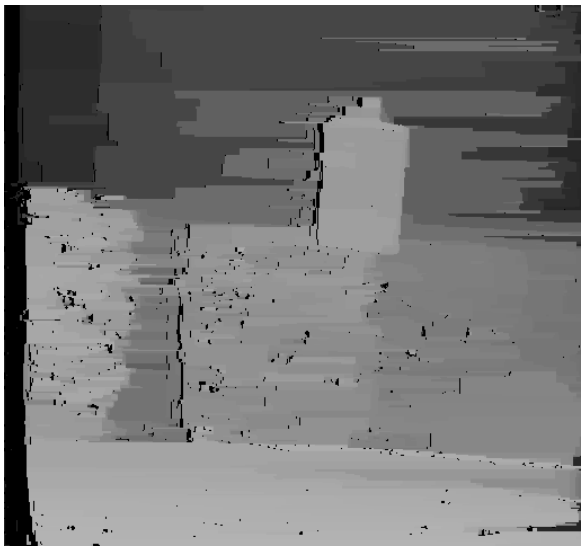

Figure 38: Disparity map obtained with seven images.

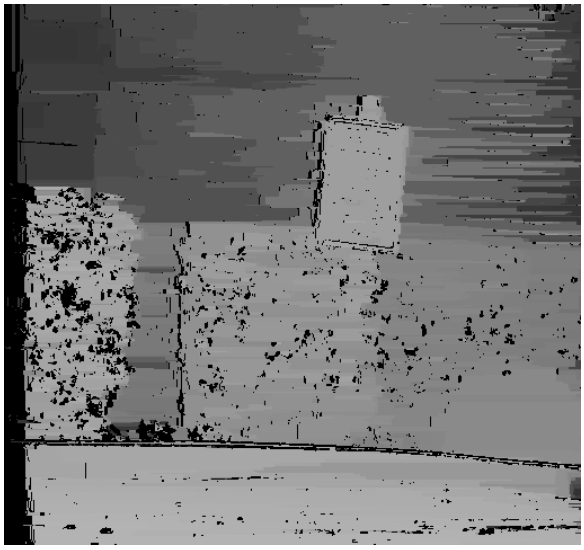

Figure 39: Disparity map obtained with six images with no modelling of the intermediate occlusion process.

7.2 Horizontal and Vertical Sequence

Figure (40) show the left, right and uppermost images of a multiframe sequence. Figure (41)

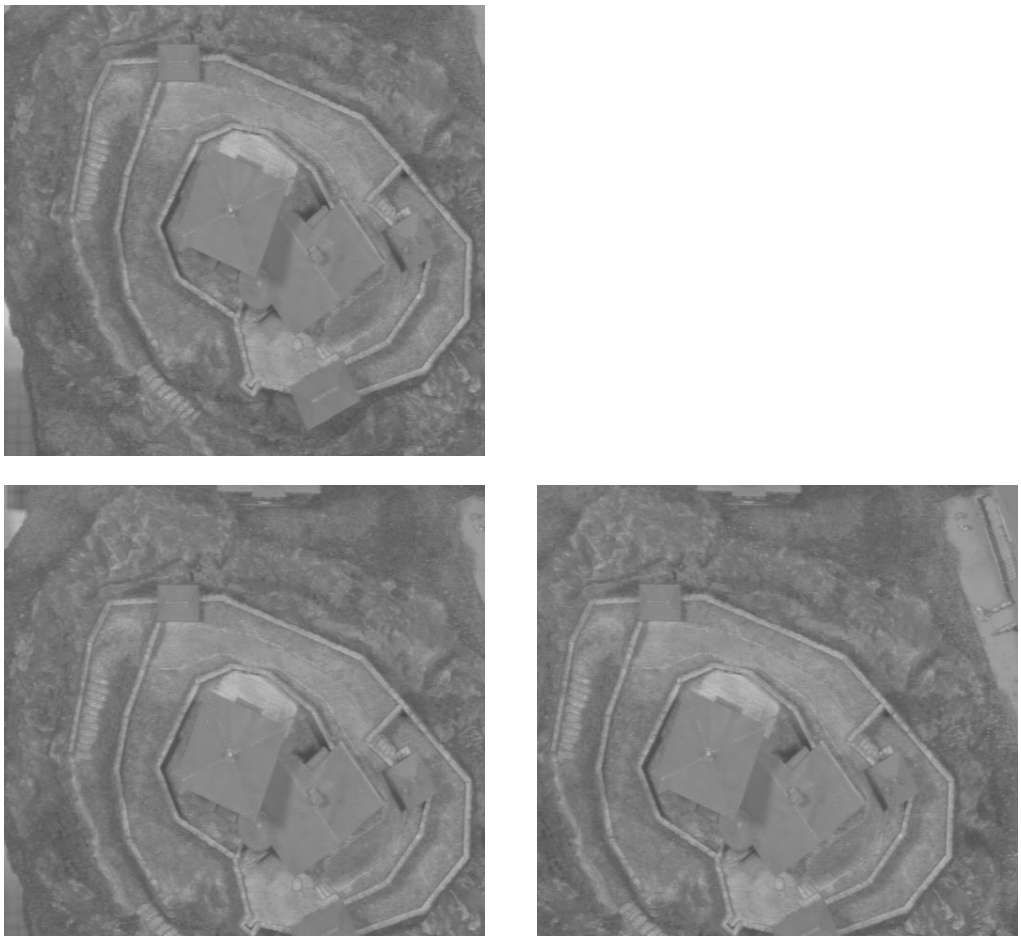

Figure 40: Left, right and uppermost images of the "Castle" sequence. (Images courtesy of T. Kanade and E. Kawamura of CMU.)

shows the disparity map obtained using only the center and rightmost images. Good depth detail is apparent but once again, there are spurious occlusion points. Although the fine perimeter wall structures appear to have been resolved close inspection revealed that this sometimes corresponded to the shadow region beside the wall and not to the wall itself, i.e. false correspondences were made. The structure at the very top of the image is poorly resolved and false correspondences are present around the small roof structure on the right. The small roof structure in the upper left of the image is barely resolved. The major roof structures are correctly discriminated although the sloping surfaces have been given a frontal planar orientation because of the implicit bias associated with the algorithm. This bias could be removed in a similar manner to Belhemeur [4].

Figure (42) shows the disparity map obtained using a trinocular configuration consisting of the three images of Figure (40). There is a noticable reduction in the number of spurious occlusion points together with a reduction in correspondence errors as expected.

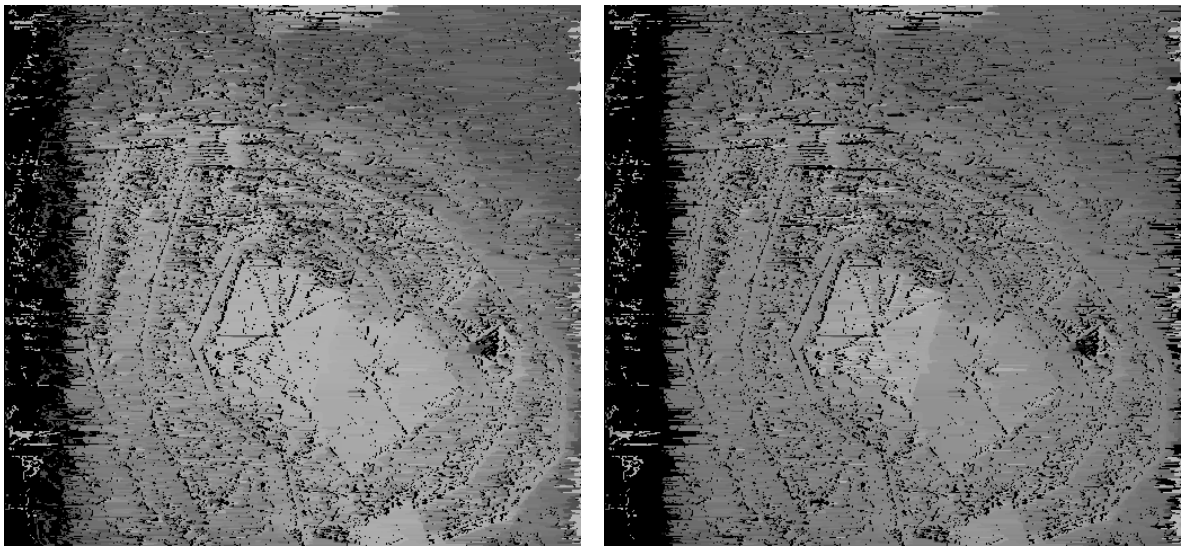

Figure 41: Disparity map for center and rightmost stereo pair. (The left image is histogram equalized while the right image has been linearly stretched from 110 to 170, the remaining intensity values being clipped to either 0 of 255).

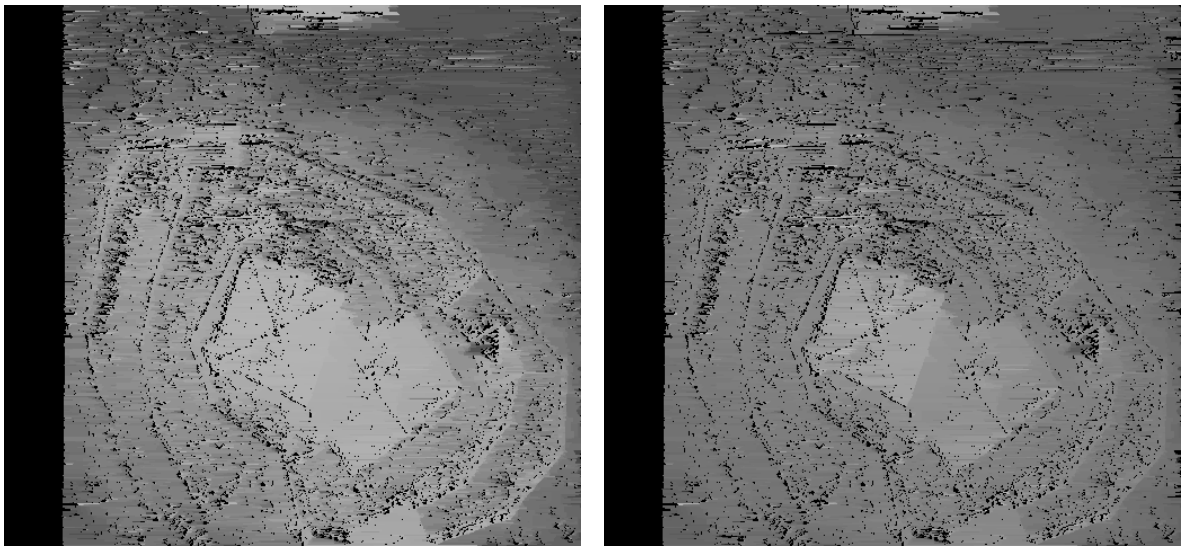

Figure 42: Disparity map for center, upper and rightmost image triple. (The left image is histogram equalized while the right image has been linearly stretched from 110 to 170, the remaining intensity values being clipped to either 0 of 255).

The small roof structure in the upper left of the image is beginning to be resolved but there are correspondence errors still in the small structure located in the middle right portion of the image.

Figure (43) shows the disparity map obtained using all 13 frames. Occlusions are now

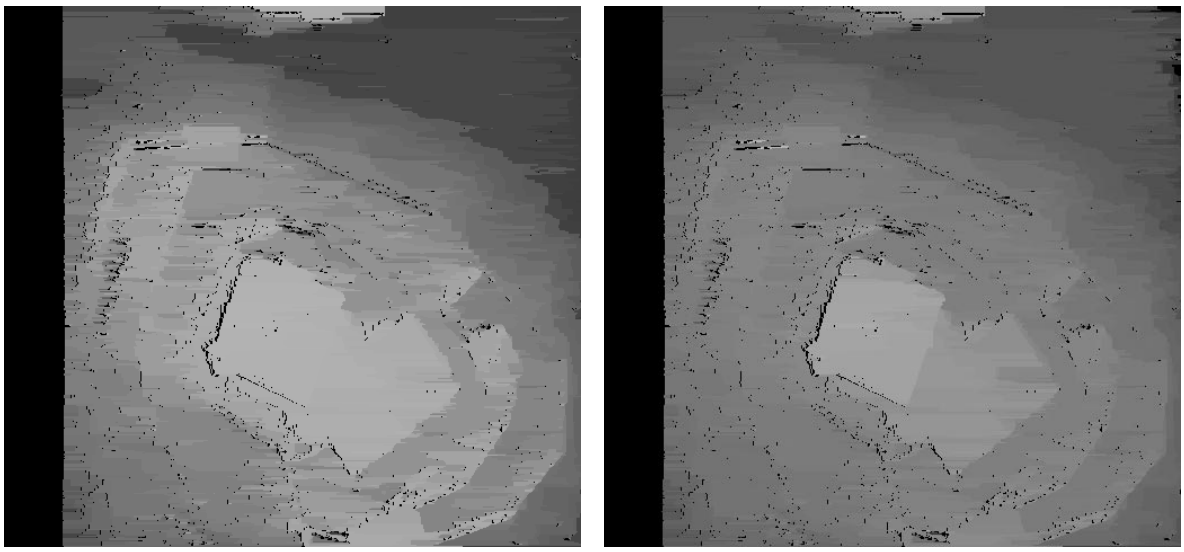

Figure 43: Disparity map obtained using all 13 images. (The left image is histogram equalized while the right image has been linearly stretched from 110 to 170, the remaining intensity values being clipped to either 0 of 255).

almost all clustered at significant changes in surface height; most spurious occlusions have been removed. The structure at the top of the image is now sharply resolved. The small building to the right has now been correctly resolved and the stairwell immediately beside it is very well defined. The small roof structure in the upper left is also correctly detected. There are some height variations on the two primary roof structures; these artifacts are mostly likely due to the algorithms implicit bias for frontal planar surfaces. However, the main depth discontinuities are very sharply defined.

It is currently unclear whether the remaining correspondence errors are due to violations of the normal assumption in the vicinity of these errors which are not corrected for by the normalization procedure or whether perhaps small errors in the camera positions are causing the verification points in the $N-2$ intermediate frames to not be true corresponding points. This needs further investigation.

8 Conclusion

Determining the correspondence between two stereo images was formulated as a Bayesian sensor fusion problem. A local cost function was derived that consists of (1) a normalized squared error term that represents the cost of matching two features and (2) a (fixed) penalty for an unmatched measurement that is a function of the probability of occlusion.

These two terms are common to other stereo algorithms, but the additional smoothing term based on disparity differences between neighboring pixels is avoided. Instead, uniqueness and ordering constraints, imposed via a dynamic programming algorithm constrain the solution to be physically sensible.

The dynamic programming algorithm has complexity $O(NM)$ which reduces to $O(N)$ if a disparity limit is set. The algorithm is fast, especially since no feature extraction or adaptive windowing ir required. Typical times for a 512x512 images with a disparity limit of 25 pixels running on a MIPS R3000 processor at 35MHz are 30s, 40s and 50s for the ML, MLMH and MLMH+V algorithms respectively. This is between 30 [15] and 1500 [9] times faster than comparable algorithms.

Experimental results were first presented for random dot stereograms. These revealed that multiple *global* minima exist. The multiple global minima cause small local differences between neighboring scanlines. In order to choose from among these global minima, two cohesivity constraints were investigated based on minimizing the total number of horizontal (MLMH) or horizontal-plus-vertical discontinuities (MLMH+V). These constraints were imposed by modifications to the dynamic programming algorithm rather than by classical regularization methods. This alternative approach is interesting and probably warrants further work. Experimental results indicate that the maximum likelihood minimum horizontal discontinuities (MLMH) also suffers from multiple global minima, though far fewer than the maximum likelihood algorithm alone. The MLMH+V algorithm improves on the performance of the MLMH algorithm even though an approximate suboptimal algorithm is employed for computational reasons. Clearly, design of an efficient, optimal algorithm for determining the MLMH+V solution is needed. Qualitatively, best disparity maps were obtained using the MLMH+V algorithm, though very acceptable results are provided by the MLMH algorithm.

A variety of stereo images were examined. The pair denoted "Shrub" revealed that the algorithm was sensitive to additive and multiplicative intensity offsets. A normalization procedure based on comparing the ten percentile points of the histograms of the two images provided a straighforward way of automatically detecting and eliminating this condition.

Using the scalar intensity values of the individual pixels as a measurement vector has the advantages of eliminating any feature extraction stage and/or complex adaptive windowing. The algorithm can be easily extended to incorporate multiple attributes into the measurement vector, e.g. intensity, color, edge, texture, provided all the elements of the vector satisfy the Normal assumption.

The maximum likelihood stereo algorithm was then generalized to N cameras. The cost function explicitly models occlusion in the principal (left and right) images together with possible occlusions in any of the intermediate views. Once again, the global cost function is efficiently minimized through dynamic programming which enforces ordering (monotonicity) and uniqueness constraints. However, while it is guaranteed that a feature in the left image will match no more than a single feature in the right image, the implementation does not prevent features in intermediate views from matching multiple features

in the left and right cameras. We do not believe this is a significant problem. However, if necessary, it could be corrected by a more elaborate and consequently computationally more expensive dynamic programming algorithm.

The experimental results clearly show a significant improvement in matching accuracy as more intermediate views are used to verify hypothesized matches. Very good detail is extracted from the images and the number of spurious occlusion points is considerably reduced. Comparison with solutions obtained with *no* modelling of intermediate occlusions clearly demonstrate the benefits and importance of the occlusion model.

There are several avenues for future work, including (1) the need to find an efficient *and optimum* solution to the MLMH+V method, (2) investigation of the validity of the constraints and assumptions. In particular, while it is common to assume uniqueness, there is evidence that human stereopsis does not impose such a condition, epscially in the perception of transparent surfaces[18, 19]. It would be challenging to eliminate this constraints. For the N-camera case, we assumed that the position of the N cameras is precisely known in order to determine the points in the $N-2$ intermediate frames. This is a very significant assumption, since if violated, a correctly hypothesized correspondence between two feature in the left and right images could project to *incorrect* positions on the image planes of the intermediate cameras. Since very significant errors might result, a sensitivity analysis should be performed. This may explain some of the correspondence errors present in the 13 frame solution.

Since the computation and memory costs scale linearly with the number of views, it is desirable to minimize the correspondence error with the fewest number of additional views. An obvious question then is where successive views should be taken. This is a difficult question with interesting connections to sensing strategies in path planning, robot map making and vision [13]. A corollary to this is whether there is a minimum distance separation. Kanade *et al* have rightly pointed out that small separations reduce the search space over which it is necessary to look for a match. However, it is unclear whether closely spaced intermediate views have less disambiguation than a more separated sequence. Moreover, in the dynamic programming approach presented here, the search space is determined only by the two principal views so the location of intermediate views should not be constrained by search space considerations.

Acknowledgements

Thanks to Y. Bar-Shalom and K. R. Pattipati of the University of Connecticut, D. Geiger of NYU, D. W. Jacobs and L. Williams for valuable discussion on issues related to this paper. Thanks to S. Roy of the University of Montreal for discussions relating to his work. Special thanks to P. Yianilos for suggesting the histogram normalization algorithm and B. Bialek for suggesting the constant multiplicative model. Also thanks to T. Kanade, T. Nakahara and E. Kawamura of CMU and J. Tajima and S. Sakamoto of NEC for supplying many of the stereo images. Special thanks to K.G. Lim of Cambridge University for the

interest shown in this algorithm. Finally, thanks to H. Bulthoff and B. Julesz for helpful comments, especially regarding human stereopsis.

References

[1] N. Ayache. *Artificial Vision for Mobile Robots: Stereo Vision and Multisensory Perception*. The MIT Press, 1990.

[2] N. Ayache and O. Faugeras. Building, registrating, and fusing noisy visual maps. In *International Conference on Computer Vision*, June 1987.

[3] H. H. Baker and T. O. Binford. Depth from edge and intensity based stereo. In *Int. Joint Conf. on Artificial Intelligence*, pages 631–636, 1981.

[4] P. N. Belhumeur. A binocular stereo algorithm for reconstructing sloping, creased, and broken surfaces in the presence of half-occlusion. In *Proc. Int. Conf. on Computer Vision*. IEEE, 1993.

[5] P. N. Belhumeur and D. Mumford. A bayesian treatment of the stereo correspondence problem using half-occluded regions. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*. IEEE, 1992.

[6] A. Blake and A. Zisserman. *Visual Reconstruction*. MIT Press, 1987.

[7] R. C. Bolles, H. H. Baker, and M. J. Hannah. The JISCT stereo evaluation. In *Proc. of DARPA Image Understanding Workshop*, pages 263–274, 1993.

[8] H. H. Bulthoff. Shape from X: Psychophysics and computation. In M. S. Landy and J. A. Movshon, editors, *Computational Models of Visual Processing*, pages 305–330. The MIT Press, 1991.

[9] S. D. Cochran and G. Medioni. 3-d surface description from binocular stereo. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 14(10):981–994, 1992.

[10] I. J. Cox. A maximum likelihood N-camera stereo algorithm. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 733–739, 1994.

[11] I. J. Cox, S. Hingorani, B. M. Maggs, and S. B. Rao. Stereo without disparity gradient smoothing: a Bayesian sensor fusion solution. In D. Hogg and R. Boyle, editors, *British Machine Vision Conference*, pages 337–346. Springer-Verlag, 1992.

[12] I. J. Cox, S. Roy, and S. L. Hingorani. Dynamic histogram warping of images pairs for constant image brightness. In *IEEE Int. Conf. on Image Processing*, (submitted) 1995.

[13] I. J. Cox and G. T. Wilfong. *Autonomous Robot Vehicles*. Springer-Verlag, 1990.

[14] J. P. Frisby and S. B. Pollard. Computational issuses in solving the stereo correspondence problem. In M. S. Landy and J. A. Movshon, editors, *Computational Models of Visual Processing*, pages 331–357. MIT Press, 1991.

[15] D. Geiger, B. Ladendorf, and A. L. Yuille. Binocular stereo with occlusions. In *Second European Conference on Computer Vision*, pages 425–433, 1992.

[16] M. A. Gennert. *A Computational Framework for Understanding Problems in Stereo Vision*. PhD thesis, MIT, 1987.

[17] B. K. P. Horn. *Robot Vision*. MIT Press, 1986.

[18] B. Julesz. *Foundations of Cyclopean Perception*. University of Chicago Press, 1971.

[19] B. Julesz. Personal communication, 1992.

[20] T. Kanade, M. Okutomi, and T. Nakahara. A multiple-baseline stereo method. In *Proceeding of DARPA Image Understanding Workshop*, pages 409–426, 1992.

[21] T. Kanade, M. Okutomi, and T. Nakahara. Unpublished preprint. 1993.

[22] D. Marr. *Vision*. W. H. Freeman & Company, 1982.

[23] D. Marr and T. Poggio. A cooperative stereo algorithm. *Science*, 194, 1976.

[24] L. Matthies. Stereo vision for planetary rovers: Stochastic modeling to near real-time implementation. *Int. J. of Computer Vision*, 8(1):71–91, 1992.

[25] L. Matthies, T. Kanade, and S. Shafer. Kalman filter-based algorithms for estimating depth from image sequences. *Int. J. Computer Vision*, 3:209–236, 1989.

[26] H. P. Moravec. Visual mapping by a robot rover. In *Proc. Int. Joint Conf. on A.I.*, pages 598–600, 1979.

[27] Y. Ohta and T. Kanade. Stereo by intra- and inter- scanline search using dynamic programming. *IEEE Trans. Pattern Analysis and Machine Intelligence*, PAMI-7(2):139–154, 1985.

[28] C. H. Papadimitriou and K. Steiglitz. *Combinatorial Optimization*. Prentice Hall, 1982.

[29] K. R. Pattipati, S. Deb, and Y. Bar-Shalom. Passive multisensor data association using a new relaxation algorithm. In *Multitarget-Multisensor Tracking: Advanced Applications*, pages 219–246. Artech House, 1990.

[30] T. Poggio, V. Torre, and C. Koch. Computational vision and regularization theory. *Nature*, 317:638–643, 1985.

[31] S. B. Pollard, J. E. W. Mayhew, and J. P. Frisby. Pmf: A stereo correspondence algorithm using disparity gradient limit. *Perception*, 14:449–470, 1985.

[32] K. Prazdny. Detection of binocular disparities. *Biological Cybernetics*, 52, 1985.

[33] S. Roy and J. Meunier. Stereoscopic analysis of multiple images. *International J. of Computer Vision*, (submitted).

[34] R. Y. Tsai. Multiframe image point matching and 3-d surface reconstruction. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 5(2):159–174, 1983.

[35] A. L. Yuille, D. Geiger, and H. Bulthoff. Stereo integration, mean field theory and psychophysics. In *First European Conference on Computer Vision*, pages 73–82, 1990.