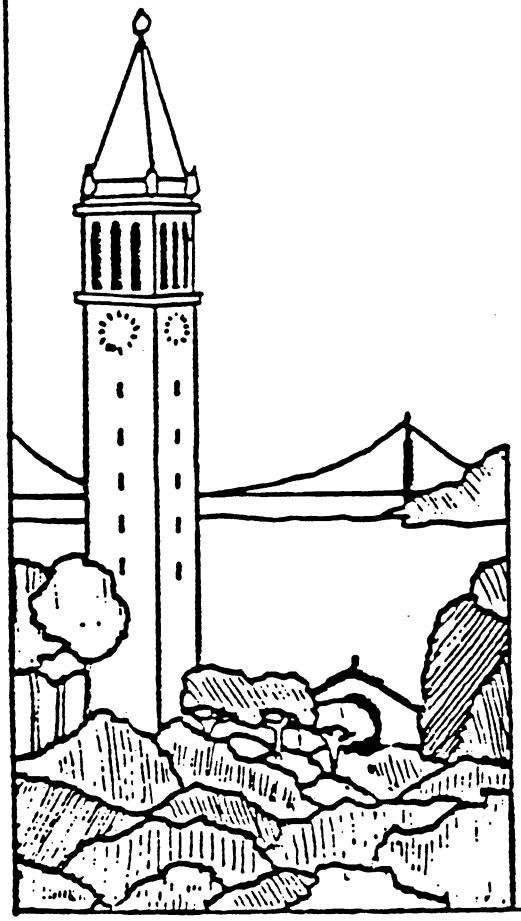
A Computational Framework for Determining Stereo Correspondence from a Set of Linear Spatial Filters

David G. Jones and *Jitendra Malik*



Report No. UCB/CSD 91/655

October 1991

Computer Science Division (EECS)
University of California
Berkeley, California 94720

A Computational Framework for Determining Stereo Correspondence from a Set of Linear Spatial Filters

David G. Jones[†] Jitendra Malik[‡]

Technical Report UCB–CSD 91–655

October, 1991

Abstract

We present a computational framework for stereopsis based on the outputs of linear spatial filters tuned to a range of orientations and scales. This approach goes beyond edge-based and area-based approaches by using a richer image description and incorporating several stereo cues that previously have been neglected in the computer vision literature.

A technique based on using the pseudo-inverse is presented for characterizing the information present in a vector of filter responses. We show how in our framework viewing geometry can be recovered to determine the locations of epipolar lines. An assumption that visible surfaces in the scene are piecewise smooth leads to differential treatment of image regions corresponding to binocularly visible surfaces, surface boundaries, and occluded regions that are only monocularly visible. The constraints imposed by viewing geometry and piecewise smoothness are incorporated into an iterative algorithm that gives good results on random-dot stereograms, artificially generated scenes, and natural grey-level images.

[†]McGill University
Department of Electrical Engineering
Montréal, Québec, Canada H3A 2A7
phone: (514) 398-8348 fax: 398-7348
email: djones@lightning.mcrcim.mcgill.edu

[‡]University of California, Berkeley
Computer Science Division
Berkeley, California, USA 94720
phone: (510) 642-7597 fax: 642-5775
email: jmalik@robotics.berkeley.edu

[‡]This work has been supported in part by a NSF PYI award (IRI-8957274) to JM.

1 Introduction

Binocular stereopsis is based on the cue of *disparity* — the two eyes (or cameras) receive slightly different views of the three-dimensional world. This disparity cue includes differences in position, horizontal and vertical, as well as differences in orientation or spacing of corresponding features in the two images and can be used to extract the three-dimensional structure in the scene.

This depends, however, upon first obtaining a solution to the correspondence problem. The principal constraints that make this feasible are:

1. Similarity of corresponding features in the two views.
2. Viewing geometry which constrains corresponding features to lie on epipolar lines.
3. Piecewise continuity of surfaces in the scene because of which nearby points in the scene have nearby values of disparity. The disparity gradient constraint (Burt and Julesz, 1980; Pollard et al., 1985) and the ordering constraint (Baker and Binford, 1982) are closely related.

Different approaches to the correspondence problem exploit these constraints in different ways. The two best studied approaches are area correlation (Hannah, 1974; Gennery, 1977; Moravec, 1977; Barnard and Thompson, 1980) and edge matching (Marr and Poggio, 1979; Grimson, 1981; Baker and Binford, 1982; Pollard et al., 1985).

The difficulties with approaches based on area correlation are well known. Because of the different viewpoints, the effects of shading can give rise to differences in brightness for non-lambertian surfaces. A more serious difficulty arises from the effects of differing amounts of foreshortening in the two views whenever a surface is not strictly fronto-parallel. Still another difficulty arises at surface boundaries, where a depth discontinuity may run through the region of the image being used for correlation. It is not even guaranteed in this case that the computed disparity will lie within the range of disparities present within the region.

In typical edge-based stereo algorithms edges are deemed compatible if they are near enough in orientation and have the same sign of contrast across the edge. To cope with the enormous number of false matches, a coarse-to-fine strategy may be adopted (e.g., Marr and Poggio, 1979; Grimson, 1981). In some instances, additional limits can be imposed, such as a limit on the rate at which disparity is allowed to change across the image (Mayhew, 1983; Pollard et al., 1985). Although not always true, assuming that corresponding edges must obey a left-to-right ordering in both images can also be used to restrict the number of possible matches and lends itself to efficient dynamic programming methods (Baker and Binford, 1982). With any edge-based approach, however, the resulting depth information is sparse, available only at edge locations. Thus a further step is needed to interpolate depth across surfaces in the scene.

A third approach is based on the idea of first convolving the left and right images with a bank of linear filters tuned to a number of different orientations and scales. The responses of these filters at a given point constitute a vector that characterizes the local structure of the image patch. The correspondence problem can be solved by seeking points in the other view where this vector is maximally similar. One approach that makes use of this representation has been previously been investigated by Kass (1983).

Our contribution in this paper is to develop this filter-based framework. We present techniques that exploit the constraints arising from viewing geometry and the assumption that the scene is composed of piecewise smooth surfaces. A general viewing geometry is assumed, with the optical axes converged at a fixation point, instead of the simpler case of parallel optical axes frequently assumed in machine vision. Exploiting piecewise smoothness raises a number of issues — the correct treatment of depth discontinuities, and associated occlusions, where unpaired points lie in regions seen only in one view. We develop an iterative framework which exploits all these constraints to obtain a dense disparity map.

This paper is organized as follows. Section 2 gives an introduction to the use of filtering as a first stage of visual processing. A technique based on using the pseudo-inverse is presented for characterizing the information present in a vector of filter responses. Section 3 demonstrates the performance of a simple-minded matching strategy based on just comparing filter responses. This helps to motivate the need for exploiting the additional constraints imposed by the viewing geometry and piecewise smoothness. These constraints are developed further in Sections 4, 5, and 6. The complete algorithm is presented in section 7.

2 Local Analysis of Image Patches by Filtering

In order to solve the correspondence problem, stereo algorithms attempt to match *features* in one image with corresponding features in the other. Central to the design of these algorithms are two choices: *What* are the image features to be matched? *How* are these features compared to determine corresponding pairs.

It is important to recall that stereo is just one of many aspects of early visual processing: stereo, motion, color, form, texture, etc. It would be impractical for each of these to have its own specialized representation different from the others. The choice of a "feature" to be used as the basis for stereopsis must thus be be constrained as a choice of the input representation for many early visual processing tasks, not just stereo. For the human visual system, a simple feature such as a "pixel" is not even available in the visual signals carried out of the eye. Already the pattern of light projected on the retina has been sampled and spatially filtered. At the level of visual inputs to the cortex, visual receptive fields are well approximated as linear spatial filters, with impulse response functions that are the Laplacian of a two-dimensional Gaussian, or simply a difference of Gaussians. Very early in cortical visual processing, receptive fields become oriented and are well approximated by linear spatial filters, with impulse response functions that are similar to partial derivatives of a Gaussian (Young, 1985).

Since "edges" are derived from spatial filter outputs, the detection and localization of edges may be regarded as an unnecessary step in solving the correspondence problem. A representation based on edges actually discards information useful in finding unambiguous matches between image features in a stereo pair. An alternative approach, explored here is to treat the the spatial filter responses at each image location, collectively called the *filter response vector*, as the feature to be used for computing stereo correspondence.

Although this approach is loosely inspired by the current understanding of processing in the early stages of the primate visual system (for a recent survey, DeValois and DeValois, 1988), the use of spatial filters may also be viewed analytically. The filter response vector characterizes a local image region by a set of values at a point. This is similar to characterizing an analytic function by

its derivatives at a point. From such a representation, one can use a Taylor series approximation to determine the values of the function at neighboring points. Because of the commutativity of differentiation and convolution, the spatial filters used are in fact computing "blurred derivatives" at each point. The advantages of such a representation have been described in some detail (Koenderink and van Doorn, 1987; Koenderink, 1988). Such a representation provides an efficient basis for various aspects of early visual processing, making available at each location of the computational lattice, information about a whole neighborhood around the point.

The primary goal in using a large number of spatial filters, at various orientations, phases, and scales is to obtain rich and highly specific image features suitable for stereo matching, with little chance of encountering false matches.

At this point, one might be tempted to formulate more precise, mathematical criteria and to attempt to determine an *optimal* set of filters. The alternative viewpoint taken here is that a variety of filter sets would each be adequate and any good stereo algorithm should not depend critically upon the precise form of the spatial filters chosen.

2.1 The Filter Set

The implementation and testing of these ideas requires some particular set of filters to be chosen, though at various times, alternative filters to those described below have been used, always giving more or less similar results.

The set of filters used consisted of rotated copies of filters with impulse responses $F(x,y) = G_n(x) \times G_0(y)$, where $n = 1, 2, 3$ and G_n is the n^{th} derivative of a Gaussian. The scale, σ, was chosen to be the same in both the x and y directions.

$$
\begin{aligned}
G_0(x) &= \frac{1}{\sigma\sqrt{2\pi}} \exp(\frac{-z^2}{2}) \quad ; \quad z = \frac{x}{\sigma} \\
G_1(x) &= -\frac{1}{\sigma} z\, G_0 \\
G_2(x) &= \frac{1}{\sigma^2}\left(z^2 - 1\right) G_0 \\
G_3(x) &= -\frac{1}{\sigma^3}\left(z^3 - 3z\right) G_0
\end{aligned}
$$

Filters at seven scales were used, with the area of the filters increasing by a factor of two at each scale. In terms of pixels, the filters are $w \times w$, with $w \in \{3, 5, 7, 10, 14, 20, 28\}$, and $\sigma = w/8$.

The filters at the largest scale are shown in Fig. 1. Smaller versions of the same filters are used at finer scales. Nine filters at seven scales would give 63 filters, except at the finest scale the higher derivatives are useless because of quantization errors, and so were discarded.

2.2 Singular Value Decomposition

Regardless of why a particular set of filters may be chosen, it is useful to know if there is an automatic procedure that can be used to evaluate the degree to which the chosen filters are independent. Any filter that can be expressed as the weighted sum of others in the set is redundant. Even filters

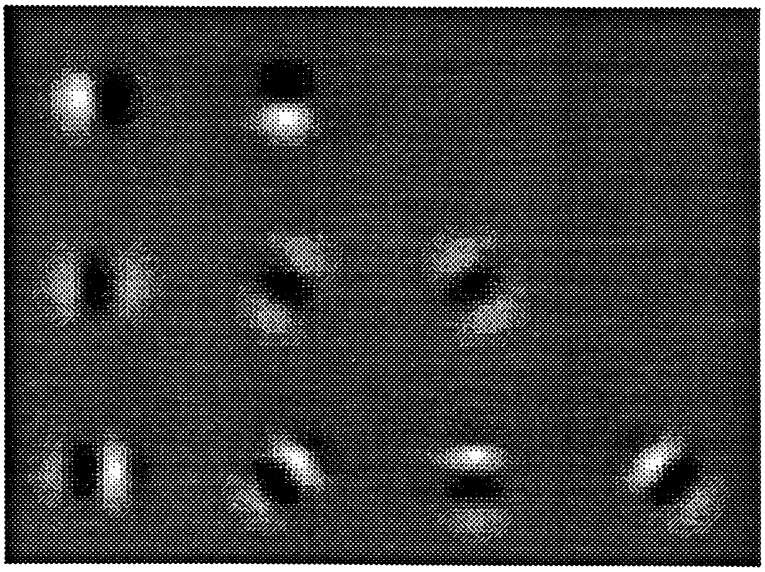


Figure 1: Linear spatial filter set.

for which this is not strictly true, but *almost* true may be a poor choice, especially where this may lead to numerical instability in some computations involving filter responses. The singular value decomposition provides just this information.

Any $m \times n$ matrix A, may be expressed as the product of an $m \times k$ matrix U, a $k \times k$ diagonal matrix Σ, and a $k \times n$ matrix V^T, where $k = \min(m, n)$, the entries in Σ are positive or zero, and the columns of U and V are orthonormal. This decomposition is known as the *singular value decomposition*. While this is the form that will be preferred by anyone implementing this on a computer (especially when $m = 900$ and $n = 60$), it is sometimes useful to write $A = U\Sigma V^T$ where U and V are both square, orthogonal matrices, and Σ is $m \times n$, with diagonal singular values $\sigma_1 \ldots \sigma_k$, sorted in decreasing order. More details may be found in a standard linear algebra or numerical analysis text (e.g., Golub and Van Loan, 1983).

A spatial filter may be represented as a column vector, F_i, by writing out its entries row by row. Taken together, a set of spatial filters forms a matrix F. This is a convenient representation of the linear transformation that maps image patches to a vector of filter responses. For an image patch represented as a vector I, the filter response vector is simply $v = F^T I$. Applying the singular value decomposition yields

$$\begin{bmatrix} & F^T & \end{bmatrix} = \begin{bmatrix} & U & \end{bmatrix} \begin{bmatrix} & \Sigma & \end{bmatrix} \begin{bmatrix} & V^T & \end{bmatrix}$$

The number of non-zero entries in Σ is the rank r, or the dimension of the vector space spanned by the filters. The first r columns of V form an orthonormal basis set for this vector space, ranked in order of the visual patterns to which this particular set of filters is most sensitive. The corresponding singular values indicate how sensitive. In the form depicted above, where V is square, the remaining

rows form an orthonormal basis for the null space of F — those spatial patterns to which F is entirely insensitive. The matrix U may be thought of as an orthonormal basis set for the space of possible filter responses vectors, or merely as a change of basis matrix.

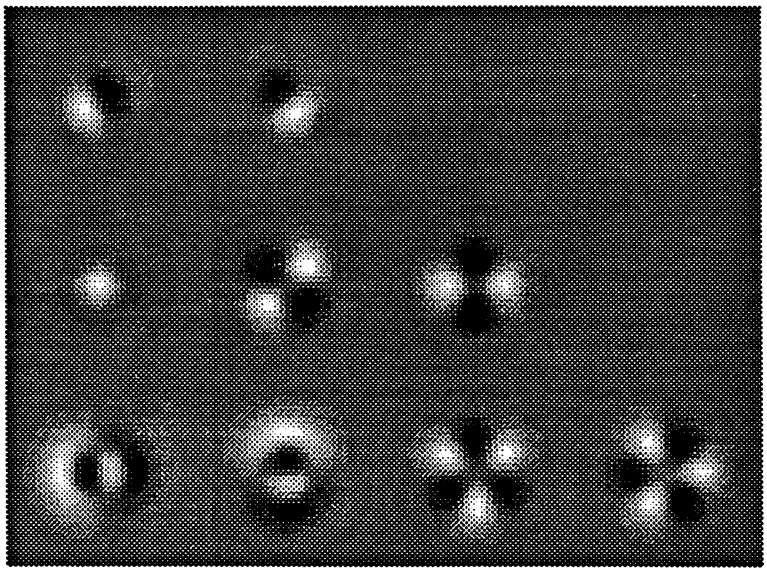


Figure 2: Orthonormal basis set.

As an example of this decomposition, the orthonormal basis for the set of filters in Fig. 1 is shown in Fig. 2.

One telltale sign of a poorly chosen set of filters is the presence of singular values that are zero, or very close to zero. Consider, for example, a filter set consisting of the first derivative of a Gaussian at four different orientations, θ.

$$G^{\theta}_{1,0}(x,y) \;=\; G_1(u) \times G_0(v) \quad ; \quad u = x\cos\theta - y\sin\theta, \quad v = x\sin\theta + y\cos\theta$$

The vector space spanned by these four filters is only two dimensional. Only two filters are needed, since the other two may be expressed as the weighted sum of these, and thus carry no additional information. If one did not already know this analytically, this procedure quickly makes it apparent. Such filters for which a small number of orientation allows the easy computation of filter responses for other orientations have been termed *steerable filters* (Koenderink, 1988; Freeman and Adelson, 1990; Perona, 1991). For Gaussian derivatives in particular, it turns out that $n + 1$ different orientations are required for the n^{th} Gaussian derivative.

As a further example, the reader who notes the absence of unoriented filters and is tempted to enrich the filter set by adding a $\nabla^2 G$, Laplacian of Gaussian filter, should think twice. This filter is already contained in the filter set in the sense that it may be expressed as the weighted sum of the oriented filters $G^{\theta}_{2,0}(x,y)$. Similar filters, such as a difference of Gaussians, may not be entirely redundant, but they result in singular values close to zero, indicating that they add little to the filter set.

At the coarsest scales, filter responses vary quite smoothly as one moves across an image. For this reason, the filter response at one position in the image can quite accurately be computed from

filter responses at neighboring locations. This means it is not strictly necessary to have an equal number of filters at the coarser scales, and any practical implementation of this approach would take advantage of this by using progressively lower resolution sampling for the larger filter scales. Regardless of such an implementation decision, it may be assumed that the output of every filter in the set is available at every location in the image, whether it is in fact available directly or may be easily computed from the outputs of a lower resolution set of filters.

2.3 Image Encoding and Reconstruction

What information is actually carried by the filter response vector at any given position in an image? This important question is surprisingly easy to answer. The singular value decomposition described earlier provides all that is necessary for the best least-squares reconstruction of an image patch from its filter response vector. Since $v = F^T I$, and $F^T = U\Sigma V^T$, the reconstructed image patch can be computed using the *generalized inverse* (or the Moore-Penrose pseudo-inverse) of the matrix F^T.

$$I' = V\, 1/\Sigma\, U^T\, v$$

The matrix $1/\Sigma$ is a diagonal matrix obtained from Σ by replacing each non-zero diagonal entry σ_i by its reciprocal, $1/\sigma_i$.

An example of such a reconstruction is given in Fig. 3. The finest detail is preserved in the center of the patch where the smallest filters are used. The reconstruction is progressively less accurate as one moves away from from the center.

Because there are fewer filters than pixels in the image patch to be reconstructed, the reconstruction is necessarily incomplete. The high quality of the the reconstructed image, however, confirms the fact that most of the visually salient features have been preserved. The reduction in the number of values needed to represent an image patch means this is an efficient encoding — not just for stereo, but for other aspects of early visual processing in general. Since this same encoding is used throughout the image, this notion of efficiency should be used with caution. In terms of merely representing the input images, storing a number of filter responses for each position in the image is clearly less efficient than simply storing the individual pixels. In terms of carrying out computations on the image, however, there is a considerable savings for even simple operations such as comparing image patches. Encoded simply as pixels, comparing 30 × 30 image regions requires 900 comparisons. Encoded as 60 filter responses, the same computation requires an order of magnitude less effort.

3 Using Filter Outputs for Matching

How should filter response vectors be compared? Although corresponding filter response vectors in the two views should be very similar, differences in foreshortening and shading mean that they will rarely be identical. A variety of measures can be used to compare two vectors, including the angle θ between them, and the length of their vector difference, that is the sum of the squared differences of individual spatial filters. These and similar measures are zero when the filter response vectors are identical and otherwise their magnitude is proportional to some aspect of the difference between potentially corresponding image patches. It turns out that any number of such measures

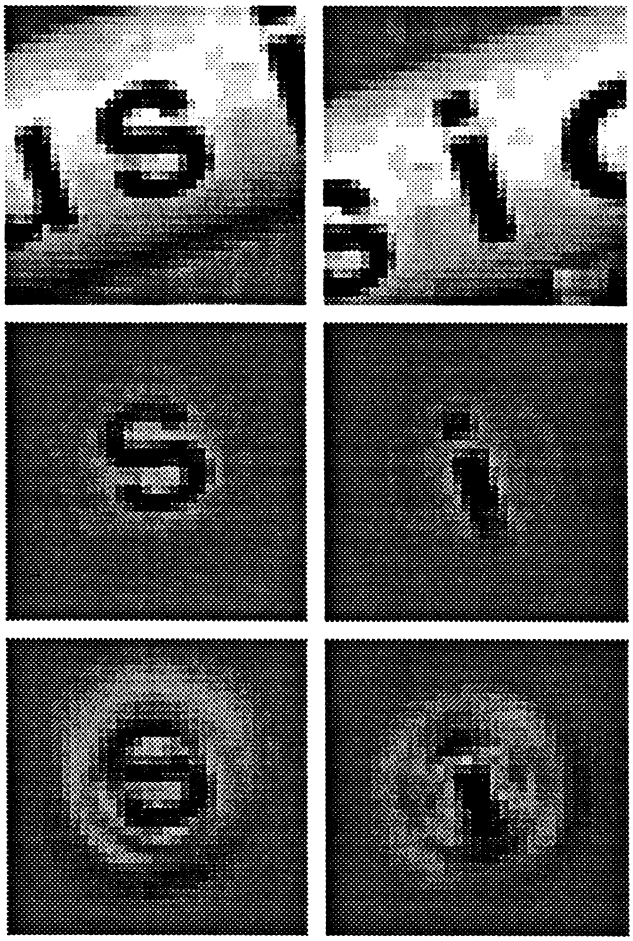


Figure 3: Image reconstruction. Two example image patches (*top*), were reconstructed (*bottom*) from spatial filter responses at their center. Original image patches masked by a Gaussian (*middle*) are shown for comparison.

do indistinguishably well at identifying corresponding points in a pair of stereo images, except at depth discontinuities.

Near an object boundary, the larger spatial filters lie across an image patch containing the projection of more than one surface. Because these surfaces lie at different depths, and thus have different horizontal disparities, the filter responses can differ considerably in the two views, even when they are centered on points that correspond. This difference is much more appropriately treated as an outlier, making least-squares approaches inapplicable. We used the L_1 norm, the sum of absolute differences of corresponding filter responses, which is a better choice than the L_2 norm under these conditions.

$$\theta = \sum_i |F_i \cdot I_r(i,j) - F_i \cdot I_l(i+h_r, j+v_r)|$$

The choice of (h_r, v_r) that minimizes this expression is taken as the best initial estimate of positional disparity.

With a basic image feature defined and rule for comparing these features to determine correspon-

dence, the core of a stereo algorithm is established. For each point in both views of a stereo pair, it is initially necessary to search a small two-dimensional region in the other view to estimate positional disparities. Though these initial disparity estimates can be quite accurate, they can be substantially improved using several techniques described in the next section.

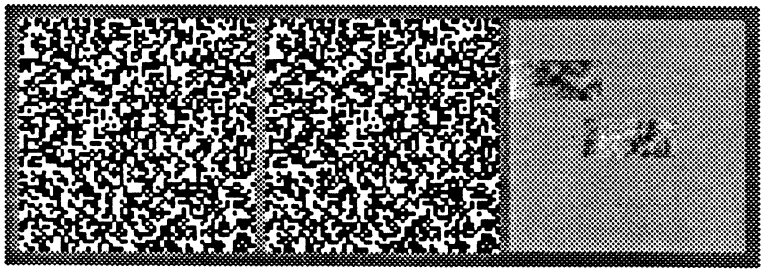


Figure 4: Determining correspondence by comparing filter response vectors.

Fig. 4 depicts an example of the how correspondence is established. For two example points in the left image, a small region of the right image is searched. For each candidate corresponding point, the difference in filter response vectors, θ, is computed. The positional offset to the point that minimizes this difference is recorded as the positional disparity. In this case, the horizontal disparity for the central point correctly indicates that the central square is standing out in depth.

When approached as a problem of determining which black dot in one view corresponds with which black dot in the other, the correspondence problem seems quite difficult. In fact, Julesz random-dot stereograms are among the richest stimuli — containing information at all orientations and scales. When the present approach using spatial filters is used, the filter response vector at each point proves to be quite distinctive, making stereo-matching quite straightforward and unambiguous.

3.1 Experimental Results

An implementation of this approach using the outputs of a number of spatial filters at a variety of orientations and scales as the basis for establishing correspondence has proven to give quite good results, for random-dot stereograms, artificial grey-level images, as well as natural images. Some typical examples are presented here.

The recovered disparity map for a Julesz random-dot stereogram is presented in Figure 5. The central square standing out in depth is clearly detected. Disparity values at each image location are presented as grey for zero horizontal disparity, and brighter or darker shades for positive or negative disparities. Because these are offsets in terms of image coordinates, the disparity values for corresponding points in the left and right images have equal magnitudes, but opposite signs. Whenever the support of the filter set lies almost entirely on a single surface, the disparity estimates are correct. Even close to depth discontinuities, the recovered disparity is quite accurate, despite the responses from some of the larger filters being contaminated by lying across surfaces at different depths.

In each view, there is a narrow region of the background just to one side of the near central square that is visible only in one eye. In this region, there is no corresponding point in the other view and the recovered disparity estimates appear as noise. Methods for coping with these initial difficulties are discussed in later sections. In the lower panels of the same figure, the measure of dissimilarity

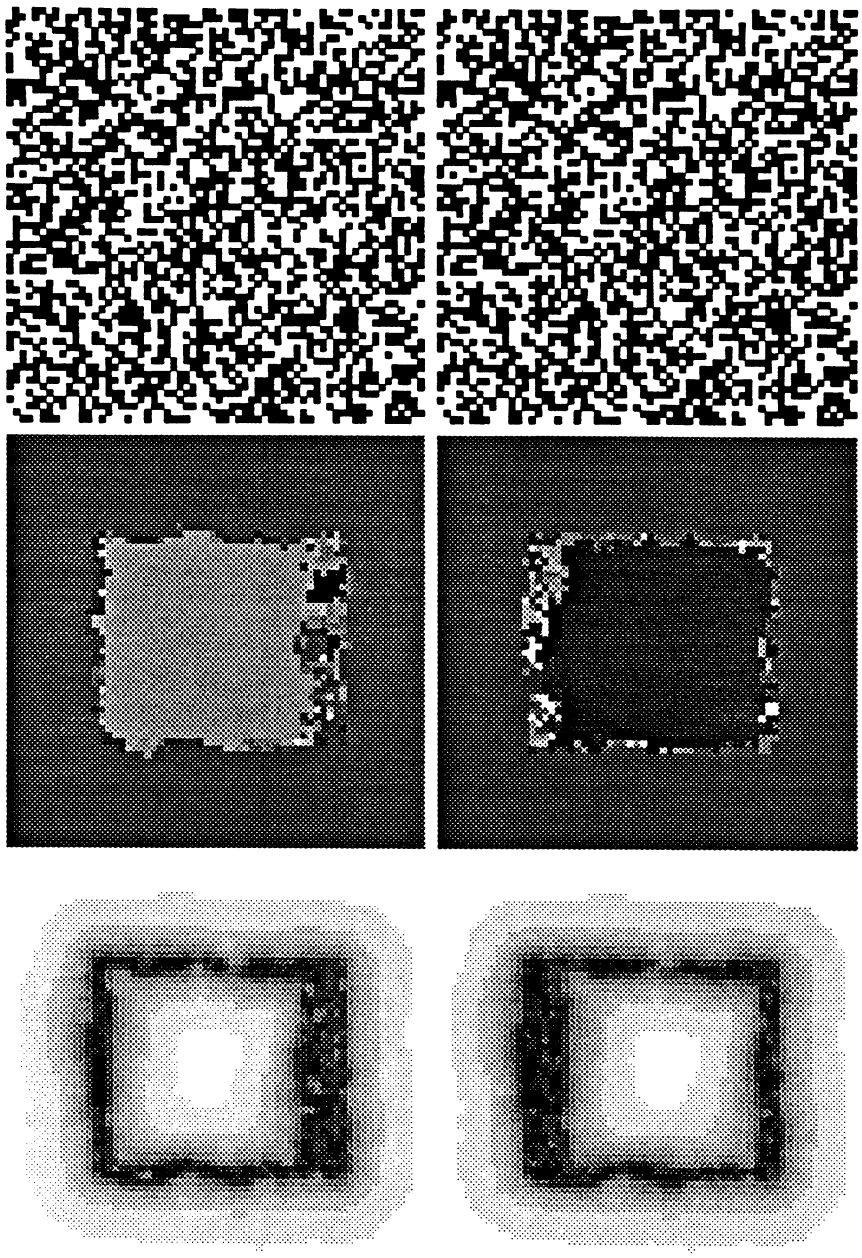


Figure 5: Disparity map: random-dot stereogram. For a random-dot stereogram (*top*), the recovered disparity map (*middle*) and dissimilarity or error map (*bottom*) are shown.

θ between corresponding filter response vectors is shown, with darker shades indicating larger differences. Larger differences are clearly associated with depth discontinuities.

As an example of a natural grey-level image, a stereo pair of fruit lying on a table cloth is shown in Figure 6. The recovered disparity values clearly matches the shapes of the familiar fruit quite well. Once again, some inaccuracies are present right at object boundaries. The measure of dissimilarity, or error shown at the bottom of the figure provides a blurry outline of the fruit in the scene. A mark on the film, present in one view and not the other (on the canteloupe) is also clearly identified

in this error image.

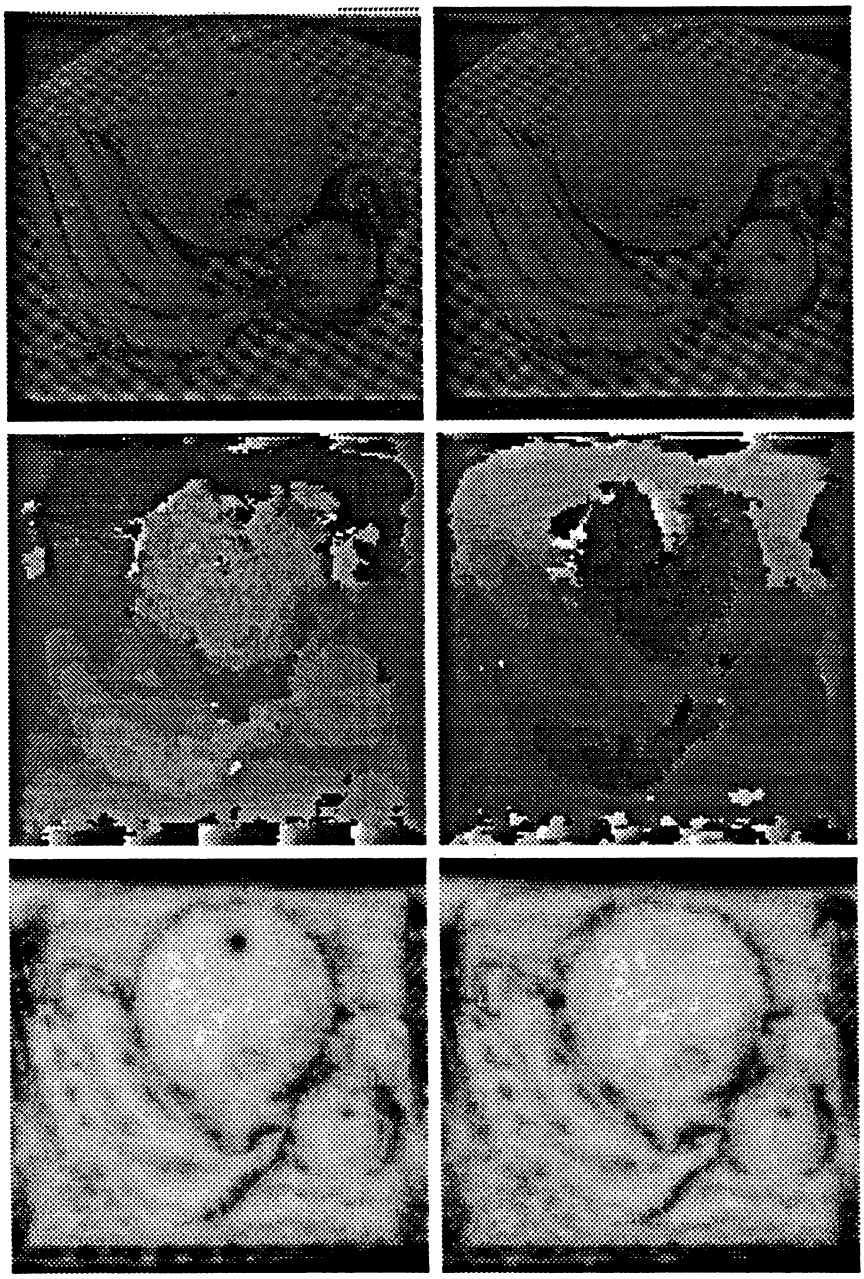


Figure 6: Disparity map: fruit. For the stereo pair shown (*top*), the recovered disparity map (*middle*) and dissimilarity or error map (*bottom*) are shown.

As a final example, a ray-traced image of various geometric shapes in a three-sided room is depicted in Figure 7. For this stereo pair, the optical axes are not parallel, but converged to fall on a focal point in the scene. This introduces vertical disparities between corresponding points. Estimated values for both the horizontal and vertical disparities are shown. Within surfaces, recovered disparities values are quite accurate and there are some inaccuracies right at object boundaries. Just to the right of the polyhedron in this scene is a region of the background visible only in one view. The

recovered disparity values are nonsense, since even though there is no correct disparity, this method will always choose one candidate as the "best". Another region in this scene where there are some significant errors is along the room's steeply slanted left wall. In this case, the large differences in foreshortening between the two views poses a problem, since the filter responses at corresponding points on this wall will be considerably different. A method for handling slanted surfaces such as this has been discussed in detail elsewhere (Jones, 1991).

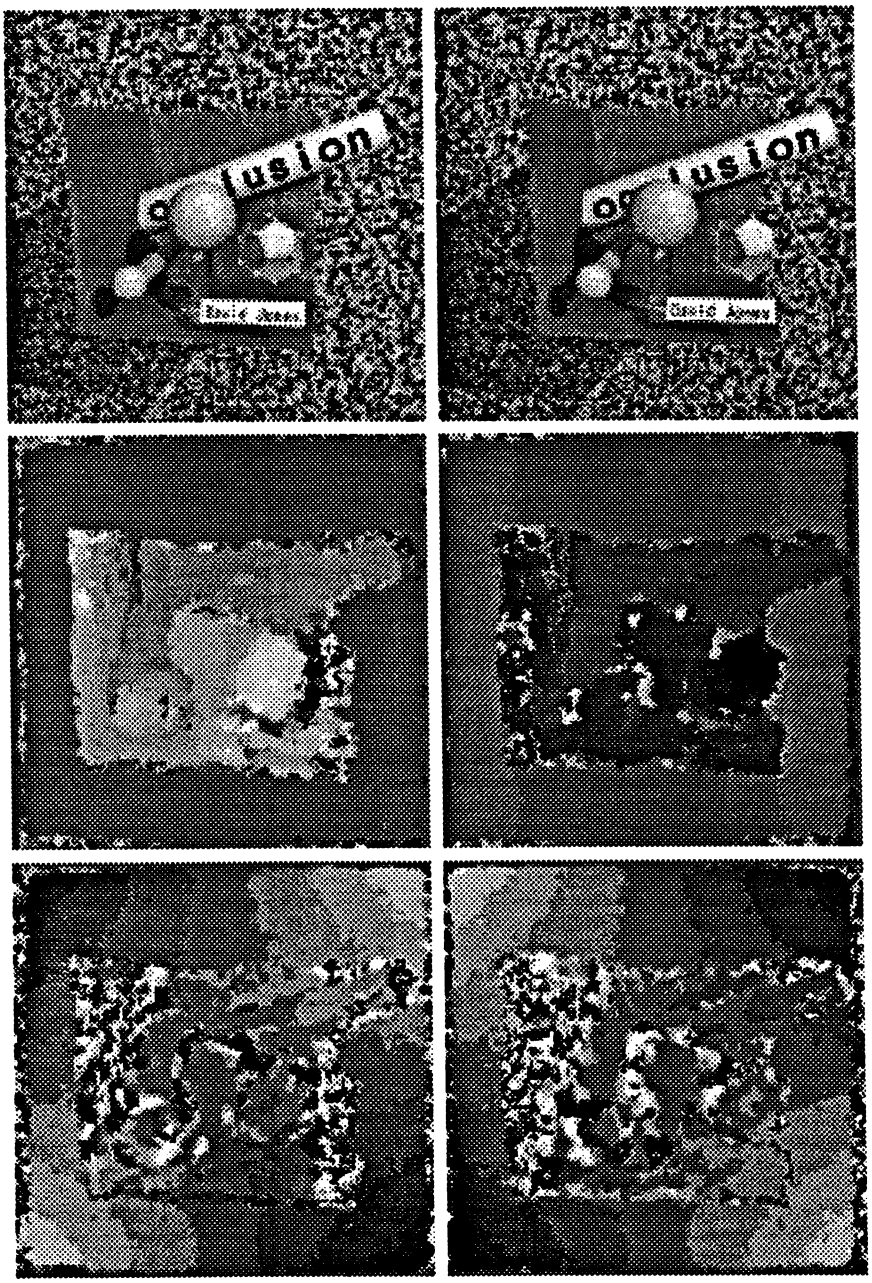


Figure 7: Disparity map: a simple raytraced room. For the stereo pair (*top*), the recovered estimates of the horizontal (*middle*) and vertical (*bottom*) components of positional disparity are shown.

4 Overview

The previous section describes a method for establishing correspondence between points in a stereo pair of images. This approach only makes use of the fact that the stereo projections of a point give rise to *similar* corresponding image patches in the two views. Here a number of additional constraints that can be applied to restrict the number of candidate correspondences and lead to more accurate disparity values are explored. These constraints fall into two distinct categories.

First, by virtue of the basic geometry involved in a pair of eyes (or cameras) viewing a three-dimensional scene, corresponding points must always lie along epipolar lines in the images. Exploiting this epipolar constraint reduces an initially two-dimensional search to a much faster and more accurate one-dimensional one. This kind of constraint applies everywhere an image, regardless of the make-up of the scene being viewed.

A second class of constraints derives from the assumption that the world is made up of piecewise smooth surfaces. Within the boundaries of a binocularly visible surface, disparity values change slowly across the image. Object boundaries, however, are often marked by sudden depth changes. Also associated with object boundaries is the presence of occluded regions visible only in one view. The identification of these three different cases in the various parts of an image allows different strategies to be applied to improve the accuracy of the recovered disparity values.

By making use of these additional constraints, an improved solution to the correspondence problem can be found. Iteratively repeating this procedure with the new disparity estimates converges towards a solution that accurately reflects the three-dimensional structure of the scene.

5 Viewing Geometry

The offsets or positional disparities between corresponding points in the left and right images of a stereo pair, may in general have horizontal and vertical components. Although this may seem to suggest that for each point in one image, a two-dimensional window in the other image must be searched in order to find the corresponding point, this amount of work is in fact unnecessary if the parameters describing the viewing situation are known. This is because for any point in one image, the corresponding point must always lie on a *epipolar line* in the other image — exactly where on this line depends on the distance of the point from the viewer.

To understand why this is true, let P be an arbitrary point visible in two views. Consider the plane determined by P and the centers of the two lenses. The intersection of this plane with each image is an line. The image of P in the left image is a point, P_l, and would be the same point for any P' on the same *line of sight*. Since this line of sight through P is also contained in the plane, the image of any P' must on a line in the right image. When a pair of cameras point straight ahead, with their optical axes parallel, these epipolar lines are horizontal. That is, there is no vertical component to positional disparity. This significantly reduces the effort needed to establish correspondence which can now be limited to a search along corresponding image scan lines. Because of this, many stereo algorithms assume either that the images come from cameras so configured, or else assume that an initial process has *rectified* the images, by transforming the images so epipolar lines become horizontal. This is straightforward if the camera parameters are known precisely, but often this is not the case.

Assuming the cameras are in a fixed configuration is unrealistic. Simply changing the direction of gaze by rotating the cameras in place invalidates the "corresponding scan line" assumption. The ability to converge the optical axes such that a major part of the scene is in the neighborhood of the fixation plane is essential for maximizing depth sensitivity.

When the viewing parameters are not known in advance, there is still information present in the images (Mayhew, 1982; Gillam, 1983). that can be used to determine where the cameras are pointing. Unfortunately, using this information would require knowing which points correspond, which is largely what the stereo algorithm itself is designed to determine! This suggests that placing image rectification strictly before stereo correspondence is unwise.

The following describes how the process of recovering the viewing parameters can be combined with the process of establishing stereo correspondence. Recovered viewing parameters provide two benefits. First, the search for corresponding points is reduced from a two-dimensional search to a one-dimensional search. Second, horizontal and vertical disparities that are inconsistent with the recovered viewing parameters can be eliminated and replaced with more accurate estimates.

5.1 Relation between viewing parameters and disparity

Each camera has a number of degrees of freedom. For translation, there are (i_0, j_0), the image coordinates of the point where the optical axis pierces the image plane, and f, the distance from the lens to the image plane. Assuming the optical axes are converged to intersect at a focal point, then a coordinate frame can be constructed so the two lens centers lie on the x-axis, separated by a baseline distance b, and the optical axes both lie in the xz-plane. Two gaze angle, γ_l, γ_r, describe the directions of the optical axes. These parameters are depicted in Fig. 8.

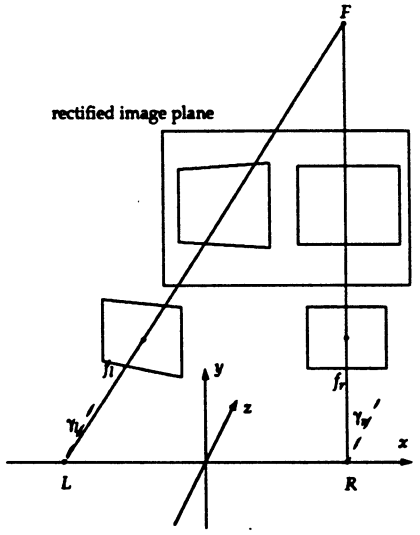


Figure 8: Coordinates for vertical disparity calculations.

Consider a point P on the image plane of a camera, with image coordinates (i, j). If the center of this lens lies at $(x_c, 0, 0)$ on the x-axis of a three-dimensional coordinate system, and it is assumed that $(i_0, j_0) = (0, 0)$, then the 3-D coordinates for P are given by:

$$x = x_c - f \sin\gamma + i \cos\gamma \qquad y = j \qquad z = f \cos\gamma + i \sin\gamma$$

If this point is projected onto a new image plane, $z = z'$, then the projected 3-D coordinates of P' become:

$$x' = x_c + \frac{-f\sin\gamma + i\cos\gamma}{f\cos\gamma + i\sin\gamma} \qquad y' = \frac{j}{f\cos\gamma + i\sin\gamma} \qquad z' = z'$$

The rectified image coordinates for the left and right camera are found by replacing x_c and γ with parameters specific to each camera.

If $P_l = (i_l, j_l)$ and $P_r = (i_r, j_r)$ are corresponding image points, then $i_l = i_r + h_r$ and $j_l = j_r + v_r$ define the horizontal and vertical disparities h_r, v_r. When these points are projected onto the rectified image plane, giving P'_l and P'_r, then there is no longer any vertical disparity, so $y'_l = y'_r$. This allows the relationship between vertical disparity on the image plane and the camera parameters to be determined:

$$\begin{aligned} y'_l &= y'_r \\ \frac{j_r + v_r}{f\cos\gamma_l + (i_r + h_r)\sin\gamma_l} &= \frac{j_r}{f\cos\gamma_r + i_r\sin\gamma_r} \\ v_r &= j_r\left(\frac{f\cos\gamma_l + (i_r + h_r)\sin\gamma_l}{f\cos\gamma_r + i_r\sin\gamma_r} - 1\right) \end{aligned}$$

This expression can be simplified by defining $C = \cos\gamma_l/\cos\gamma_r$, $T_l = \tan\gamma_l/f$, and $T_r = \tan\gamma_r/f$.

$$v_r = j_r\left(C\left(\frac{1 + (i_r + h_r)T_l}{1 + i_r T_r}\right) - 1\right)$$

Notice that when the cameras are both pointed nearly straight ahead, with γ_l and γ_r near zero, then C is near 1, while T_l and T_r are both near 0. Clearly, under these circumstances, the expression for vertical disparity, v_r, is very small. Also notice that near the center of the image, where (i_r, j_r) is near $(0,0)$, vertical disparity is very small, for a very wide range of horizontal disparities. Vertical disparities can only be large away from the center of the image, in the periphery. Lastly, notice that if a particular point, P_r, in the right image is selected and its image coordinates (i_r, j_r) treated as constants, then the expression for v_r a linear function of h_r. This determines the (epipolar) line in the left image on which P_l, the point corresponding to P_r, must lie.

If the focal lengths of the cameras are not known to be identical, and the image origins (i.e., where the optical axis pierces the image plane) are not known, then the expression for v_r becomes a bit more complicated. Using f_l, f_r for the distance from the lens to the image plane of each camera and (i_{0r}, j_{0r}), (i_{0l}, j_{0l}) for the coordinates of the image origins,

$$v_r = (j_r - j_{0r})\left(C'\left(\frac{1 + (i_r + h_r)T'_l}{1 + i_r T'_r}\right) - 1\right) + (j_{0l} - j_{0r})$$

where C', T'_l, and T'_r are defined as follows:

$$\begin{aligned} C' &= \frac{\cos\gamma_l(1 - \frac{i_{0l}\tan\gamma_l}{f_l})}{\cos\gamma_r(1 - \frac{i_{0r}\tan\gamma_r}{f_r})} \\ T'_l &= \frac{\tan\gamma_l}{f_l - i_{0l}\tan\gamma_l} \\ T'_r &= \frac{\tan\gamma_r}{f_r - i_{0r}\tan\gamma_r} \end{aligned}$$

Substituting $(0,0)$ for the image origins gives the previous, simpler expression for v_r.

Additional parameters may be added to allow for more general viewing situations. For example a parameter, α. If the optical axes of the two cameras are converged, but do not intersect at a focal point, then it may be assumed that they make an angle $\pm\alpha$ with the xz-plane and come closest to intersecting at some "intended" focal point, $F = (x_f, \pm y_f, z_f)$. If this is not true initially, then the coordinate frame can always be rotated to make it true. The expression for v_r in this more general viewing situation becomes more complicated, but can be derived nonetheless using the same method as above.

5.2 Recovering Viewing Parameters

The importance of deriving an expression for vertical disparity, v_r, in terms of image coordinates, (i_r, j_r), horizontal disparity, h_r, and viewing parameters is that this could be used to reduce the a two-dimensional search for corresponding points to a one-dimensional search, if the viewing parameters were known, or could somehow be recovered.

If reasonable initial estimates of horizontal and vertical disparities can be determined at each position in the image, then the viewing parameters can be estimated quite accurately. Since this relationship involving vertical disparity must be true throughout the image, a 256 x 256 image, for example, gives 65,536 expressions involving the unknown viewing parameters! Furthermore, since reasonable initial guesses for the parameters are often known, (e.g., $C' = 1$, $T'_l, T'_r = 0$, $j_{0l}, j_{0r} = 128$) any of a number of methods could be used to recover the viewing parameters quite robustly. A current estimate of the viewing parameters can be used to calculate the difference between the observed and expected values of vertical disparity at each point in the image. This sum of these *errors* can be used to drive a gradient descent algorithm, which quickly converges to a solution. For the results presented here, an implementation of the simplex algorithm was used (Numerical Recipes).

Once good estimates of the viewing parameters are available, positional disparity estimates that were initially incorrect may be discarded if they are inconsistent with the recovered viewing parameters. Re-estimating the positional disparities can be done efficiently by restricting search to be near to epipolar lines. These more accurate positional disparity estimates give rise to improved estimates of the viewing parameters, and so on. In practice, this process converges quickly.

6 Piecewise Continuity

The assumption that the scenes under consideration consist of smooth, piecewise continuous, opaque surfaces allows several techniques to be applied to improve positional disparity estimates. This requires that a distinction be made between regions of the images that corresponding to projections of binocularly visible surfaces, object boundaries, and monocularly visible surface regions.

6.1 Smoothness and Consistency

A common way of dealing with initially noisy or inaccurate disparity estimates is to use some amount of smoothing or regularization (e.g., Poggio, et al., 1985). With these methods, disparity values

are constrained to be *similar* to neighboring disparity values except when they lie across surface boundaries. While earlier approaches ignored the distinction between pixel neighbors which lie on the same surface and pixel neighbors which lie across discontinuity curves (Grimson, 1981), more recent approaches to the problem of surface interpolation and reconstruction have developed the basic methodology for smoothing in the presence of discontinuities (Geman and Geman, 1984; Blake and Zisserman, 1987), (Mumford and Shah, Perona and Malik). The locations of discontinuities, represented either explicitly or implicitly, must be to influence the local operations performed.

Because we perform the matching process independently starting from the left and right views, we have an additional constraint that must be satisfied in binocularly visible regions. The positional disparity values for corresponding points should have equal magnitudes, but opposite signs.

This constraint of local surface smoothness, and consistency between the two views can be applied both as a post-processing step to clean up recovered disparity values as well as providing a restricted range of acceptable disparities for a subsequent iteration of solving the correspondence problem.

A simple approach that may be applied throughout image regions corresponding to binocularly surfaces is to use median filtering to discard erroneous disparity values within smooth surfaces, while leaving sharp depth discontinuities at object boundaries. How these binocularly visible regions are distinguished from those visible only in one view is described in the next section.

6.2 Detecting and Localizing Occlusion

Whenever there is a surface depth discontinuity which has a vertically oriented edge, distant surfaces are occluded to different extents in the two eyes, leading to the existence of unpaired image points which are seen in one eye only. The realization of this goes back to Leonardo Da Vinci who as a sculptor and painter was very aware that there was a certain aspect of real three-dimensional objects and scenes that he could never capture in a painting, namely the quality of *relief*.

> It is impossible that a picture copying outlines, shade, light, and color with the highest perfection can appear to possess the same relief as that which appears in an object in nature.

He observed that to the right of a near object is a region the right eye can see, but the left eye cannot. Contrary to the now common notion of disparity, he reasoned that the critical cue for the perception of relief was the presence of these monocularly visible regions — something he could never reproduce in a painting (translation in, Kemp, 1989). This situation is depicted in Fig. 9.

Recent psychophysical work by Nakayama and Shimojo(Nakayama and Shimojo, 1990) has convincingly established that the human visual system can exploit this cue for depth in a manner consistent with the geometry of the situation.

For a point inside a monocularly visible image region, there is no correct disparity value that can be assigned because there is no corresponding point in the other view. Any method that systematically attempts to assign disparity values to each location in an image based on the local intensity structure in the image (e.g., the method of Section 3), will likely end up with estimates that are random noise, or worse, systematically wrong. Examples of this can be found by inspecting the occluded regions in Fig. 7. On the other hand, since occluded regions necessarily comprise a

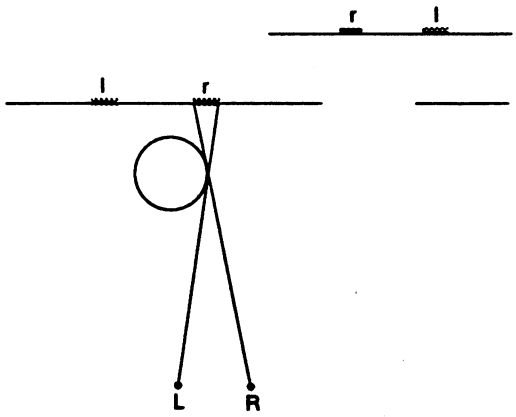


Figure 9: Occlusion. In this view from above, it is clear that at depth discontinuities there are often regions visible to one eye, but not the other. To the right of each near surface is a region r that is visible only to the right eye, R. Similarly, to the left of a near surface is monocular region, l, visible only to the left eye, L.

relatively small fraction of the overall image area, the effect of these inaccurate disparity estimates on the recovery of viewing parameters described in the previous section is small, and would be even less if a method that discards outliers were used.

The identification of occluded regions is important for several reasons. First, nonsense disparity estimates within occluded regions can be eliminated entirely from the recovery of viewing parameters. Second, the knowledge of which regions are binocularly visible and which are not can be used when refining disparity estimates, especially to prevent nonsense disparity estimates within occluded regions from being propagated to neighboring binocularly visible regions. Lastly, identified occluded regions make certain aspects of the three-dimensional structure of a scene explicit, including the delineation of the occluding contours.

A common way of dealing with these occluded regions is to treat them as sources of noise and use some amount of smoothing, based upon the assumption that the scene consists of a single smooth continuous surface. The presence of occluded regions violates this assumption. When disparity estimates inside occluded regions are not random noise, but instead have some consistent value, smoothing can have the effect of propagating errors to neighboring image regions. If these two kinds of regions, binocularly and monocularly visible, could be identified, even roughly, then applying different iterative update rules to each could potentially do much better than the indiscriminate smoothing of disparity values.

As a first try, it would seem reasonable that regions where disparity estimates are noisy, or where the *errors* associated with the best disparity values are still high should be good candidates for being labelled as occluded regions. In practice, these are rather poor cues, because disparity values inside occluded regions can sometimes be systematically wrong instead of just noisy, and it is also often possible for a search for a corresponding image patch to fortuitously find a nearby image region that is similar enough to yield a reasonably small matching error. Still, inspection of Fig. 7 suggests that it should be possible to determine roughly where the occluded regions are.

Another possibility is suggested by the fact that an occluded region in the right image can only occur to the right of a nearer surface, but never on the left, and vice versa for the left image. A right image region could be labelled as occluded if it was flanked on the left by near surface and

on the right by a far surface, with the width of the occluded region proportional to the difference in disparity. In practice, however, inspection of disparity values in the neighborhood surrounding a point in question is quite problematic, especially when they are either noisy initial estimates, or simply incorrect because they are themselves in an occluded region.

The problem of detecting and localizing occluded regions in a pair of stereo the images is made much easier when one recalls that there are indeed a *pair* of images. The occluded regions in one image include exactly those points for which there is no corresponding point in the other image. This suggests that the best cue for finding occluded regions in one image lies in the disparity estimates for the *other* image!

Define a *visibility map* for one view as being 1 at each image position that is visible in the other view, and 0 otherwise (i.e., an occluded region). The horizontal and vertical disparity values for each point in, say, the left image are signed offsets to give the coordinates of the corresponding point in the right image. If the visibility map for the right image is initially all zero, it can be filled in systematically as follows. For each position in the left image, set the corresponding position in the right visibility map to 1. Those positions that remain zero had no corresponding point in the other view and are quite likely occluded. An example of a visibility map computed in this manner is shown in Fig. 11.

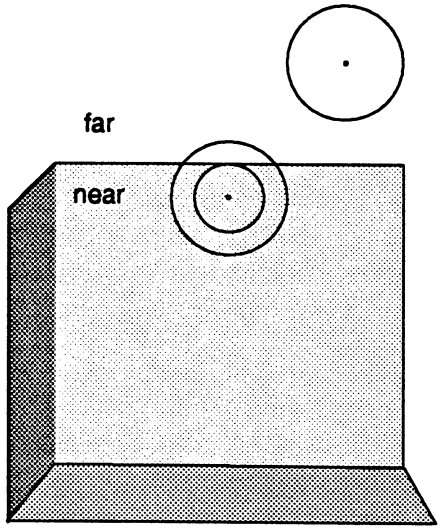


Figure 10: Scale selection. Schematic diagram depicting a three-sided room similar to the one in Fig. 7. When attempting to determining correspondence for a point on a near surface, larger filters that cross depth boundaries can result in errors. If depth discontinuities could be detected, such large scale filters could be selectively ignored in these situations.

Small errors in disparity estimates or unavoidable quantization errors may leave small cracks in the regions labelled as binocularly visible. To alleviate this, a variety of methods could be used such as applying some small amount of smoothing or median filtering to the the visibility map, slightly thinning the regions labelled as occluded, as was done here. Since real occluded regions tend to occur as long cohesive regions along object boundaries, these methods generally apply. On the other hand, any use of the visibility map must be able to tolerate errors.

Having established a means for finding regions visible only from a one viewpoint, what has been achieved? If the disparity values are accurate, then the visibility map, besides simply identifying

binocularly visible points, also explicitly delimits occluding contours.

If disparity values are only initial estimates, then knowledge of whether a point is likely to be binocularly or monocularly visible can be used to decide how to refine the disparity estimates. In binocularly visible regions median filtering enforces a smoothness constraint within surface patches while retaining the sharp depth discontinuities at depth boundaries. On the other hand, within monocularly visible regions, initial disparity estimates are very likely meaningless and should be ignored. If a disparity value must be assigned, then it might seem reasonable to replace the disparity values inside an occluded region by the disparity values of the further of the two adjacent binocularly visible regions. In the right image, this should lie to the right of the occluded region; in the left image, to the left. In practice, this is very unstable when applied iteratively. If a position is mistakenly labelled as occluded, then its disparity value may be drastically changed. Such large changes may lead to increased errors in the visibility map, and so on. A particularly serious example of this occurs for steeply slanted surfaces such as the left wall in Fig. 7. Quantization errors unavoidably leave small errors in the right visibility map. Replacing disparity estimates by adjacent further values quickly causes the disparity estimates for such a surface to collapse and become equal to an adjacent far surface.

Given that the visibility map already identifies which regions are occluded, the above step of assigning occluded regions the same disparity as the more distant neighboring visible surface can be saved until last, after the disparity values have been iteratively refined.

6.3 Depth Discontinuities and Scale Selection

The output of a set of spatial filters at a range of orientations and scales provides a rich description of an image patch. For corresponding image patches in a stereo pair of images, it is expected that these filter outputs should be quite similar. This expectation is reasonable when all of the spatial filters are applied to image patches which are the projections of single surfaces. When larger spatial filters straddle depth discontinuities, possibly including occluded regions, the response of filters centered on corresponding image points may differ quite significantly. This situation is depicted in Fig. 10. Whenever a substantial area of a filter is applied to a region of significant depth variation, this difficulty occurs (Fig. 7).

From an initial disparity map, it is possible to estimate where such *inappropriately large scale* filters are being used by applying the following procedure. At each position in the image, the median disparity is determined over a neighborhood equal to the support of the largest spatial filter used for stereo matching. Over this same neighborhood, the difference between each disparity estimate and this median disparity is determined. These differences are weighted by a Gaussian at the same scale as the filter, since the center of the image patch has a greater effect on the filter response. The sum of these weighted disparity differences provides a measure of the amount of depth variation across the image patch affecting the response of this spatial filter. When this sum exceeds an appropriately chosen threshold, it may be concluded that the filter is too large for its response to be useful in computing correspondence. Otherwise, continuing to make use of the outputs of large spatial filters provides stability in the presence of noise.

Taking the difference between the disparity estimate at each location and the median disparity value of the neighborhood makes the implicit assumption that surfaces are nearly fronto-parallel. Instead, it will generally be better to use the difference from the best-fitting *plane* of disparity over the neighborhood.

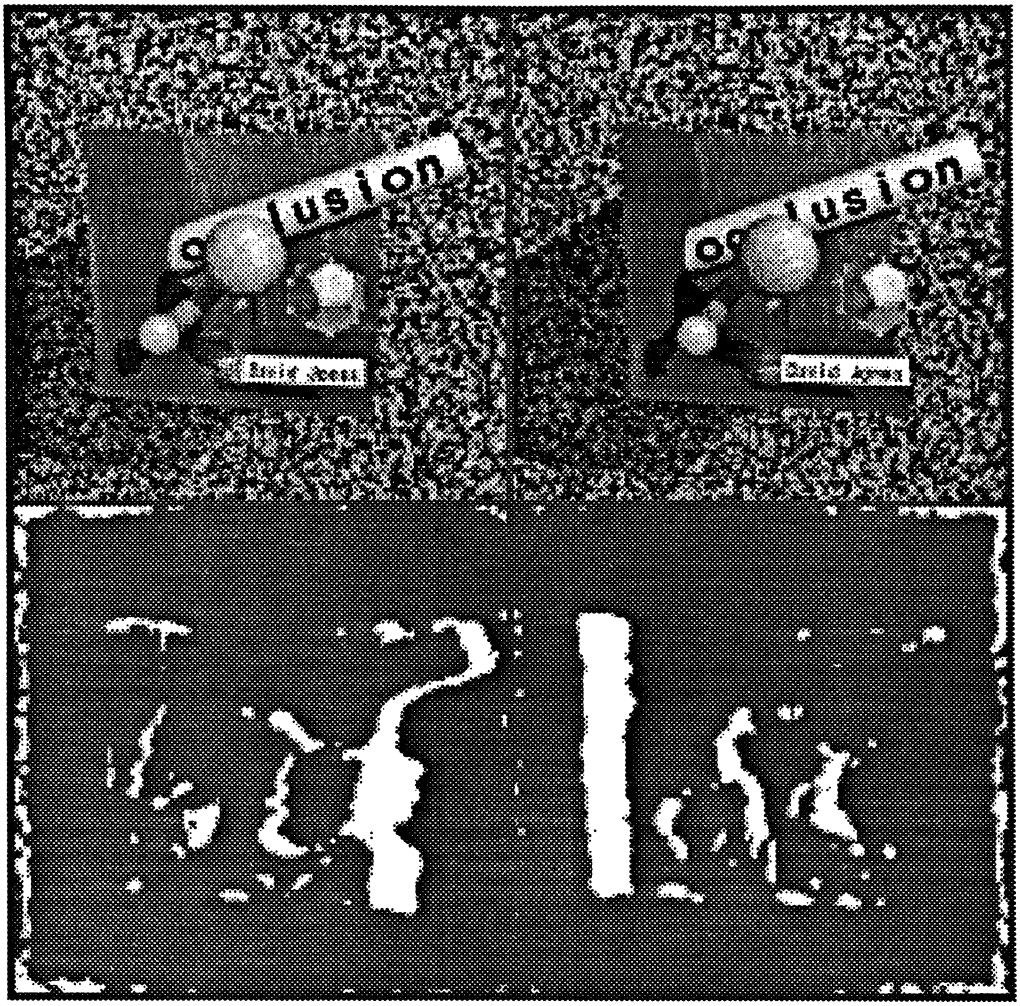


Figure 11: Visibility map. The white areas in the lower panels mark the regions determined to be visible only from one of the two viewpoints.

To record the results of applying the previous procedure, the notion of a *scale map* is introduced. At each position in an image, the scale map records the scale of the largest filter used in computing stereo correspondence. For the computation of initial disparity estimates, all the scales of spatial filters are used. From initial disparity estimates, it is possible to modify the scale map as described above. For each position in the scale map, if it is determined that an inappropriately large scale filter was used, then the scale value at that position is decremented. Otherwise, the test is redone at the next larger scale, if there is one, to see if the scale can be incremented. It is important that this process of adjusting the scale map is done in small steps, with the disparity values being recalculated between each step. This prevents an initially noisy disparity map, which seems to have a great deal of depth variation, from causing the largest scale filters to be incorrectly ignored. An example scale map is shown in Fig. 12.

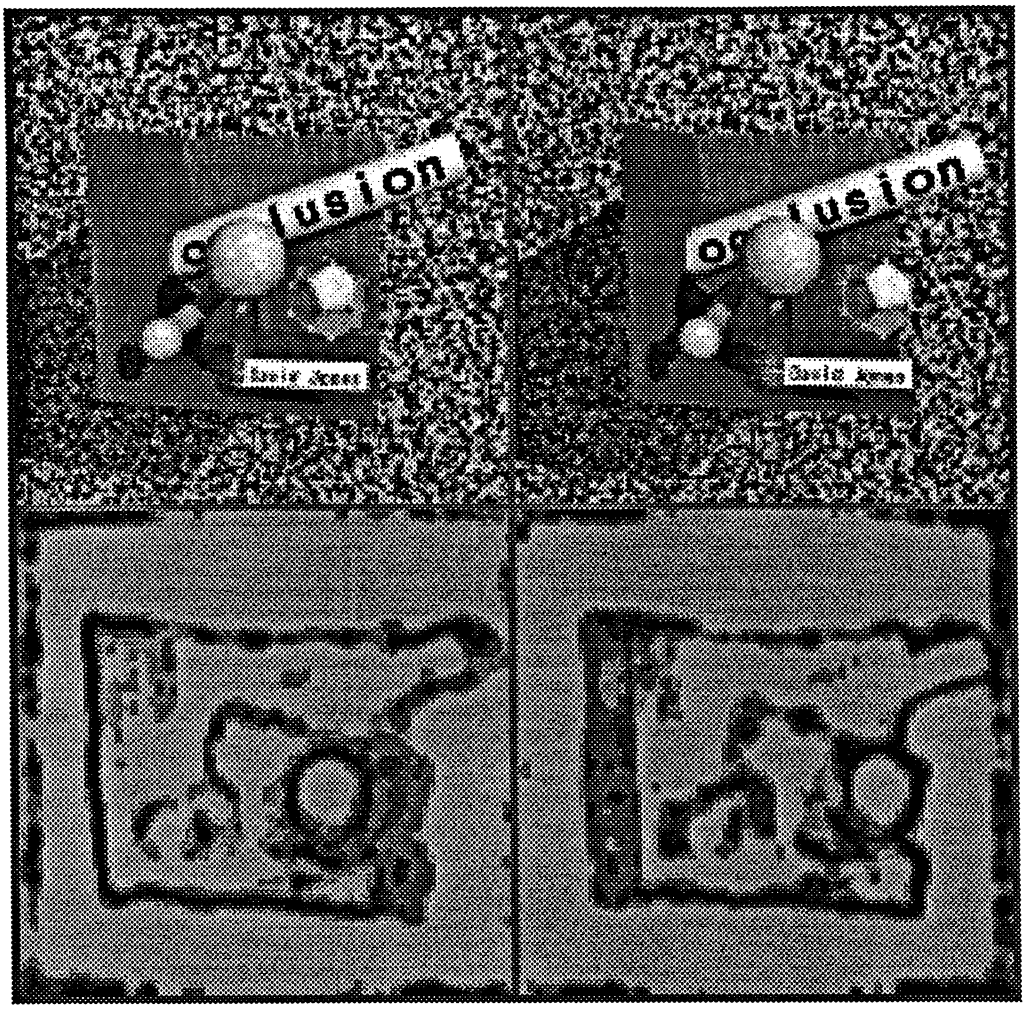


Figure 12: Scale map. The darker areas in the lower panels mark the regions where larger scale filters are being discarded because they lie across depth discontinuities.

7 A Refined Iterative Stereo Algorithm

Once initial estimates of horizontal and vertical disparity have been made, additional information becomes available which can be used to improve the quality of the disparity estimates. This additional information includes estimates of the viewing parameters, the location of occluded regions, and the appropriate scale of filters to be used for matching.

7.1 Iterative Framework

Besides cleaning up some obvious inconsistencies in the disparity map by enforcing piecewise smoothness, it is possible to improve upon the initial disparity estimates by re-solving the correspondence problem, but this time with added knowledge about the viewing geometry, and estimates of the locations of binocularly, versus monocularly visible surfaces. Instead of simply minimizing the dissimilarity measure, θ, between candidate filter response vectors, subsequent iterations minimize

an error value, e.

$$e = e_m + \lambda_v e_v + \lambda_c e_c + \lambda_s e_s$$

Each term enforces one of the constraints discussed: similarity, viewing geometry, consistency, and smoothness. The λ parameters control the weight of each of these constraints, and their specific values are not particularly critical.

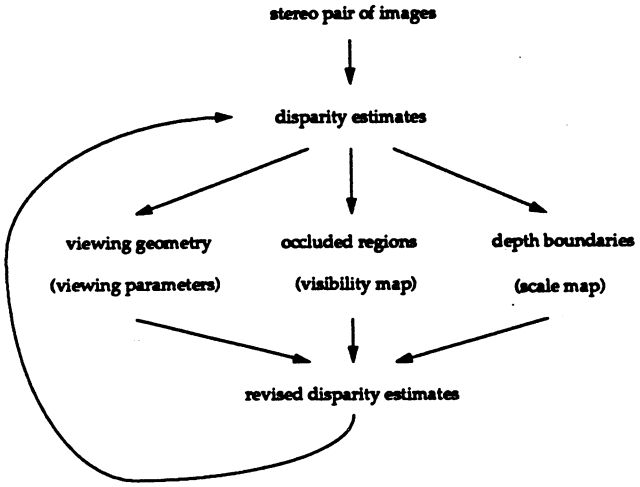


Figure 13: Iteratively refining estimates of stereo disparity.

7.2 Matching Error

For a point $P = (i, j)$ in one image of a stereo pair, and candidate horizontal and vertical disparities, $\hat{h}, \hat{v}$, the matching error e_m reflects the *dissimilarity* between the filter response vector at (i, j) in this image and the filter response vector at $(i + \hat{h}, j + \hat{v})$ in the other image. The details of how the value, θ, is computed to reflect the dissimilarity between filter response vectors were given in Section 3. Here a slight modification needs to be made since for some points the range of scales, recorded in the scale map, will have been reduced as described earlier. In this case, the spatial filters outside this range are ignored when computing θ. When a point has been labelled in the visibility map as being visible only in one view, there is no corresponding point in the other view. In this case the matching error is set to zero.

$$e_m = \begin{cases} 0 & \text{if the point is only monocularly visible} \\ \theta & \text{otherwise} \end{cases}$$

7.3 Viewing Geometry Error

For a given position in the image, (i, j), and a candidate horizontal disparity $\hat{h}$, only one value for vertical disparity, v^*, is consistent with the recovered viewing parameters. The difference between this and the candidate vertical disparity $\hat{v}$ provides a measure of how compatible the candidate disparities are with the viewing parameters. Since the viewing parameters are only approximately correct, small deviations from the estimated epipolar lines must be allowed.

$$e_v = |\hat{v} - v^*|$$

7.4 Consistency Error

Let $\hat{h}$, $\hat{v}$ be candidate horizontal and vertical disparity values for a point $P = (i, j)$ in one image of a stereo pair. If these disparity values are accurate, then the corresponding point in the other image will have coordinates $P' = (i + \hat{h}, j + \hat{v})$. For the disparity values, h', v', assigned to that point in the other image to be consistent, they will have equal magnitudes, but opposite signs (i.e., $h' = -\hat{h}$ and $v' = -\hat{v}$). When this is not so, a consistency error, e_c, can be used to penalize such inconsistent candidate disparities.

Assigning a reasonable consistency error is complicated slightly by the existence of occluded regions. If both P and P' have been labelled as binocularly visible, then e_c is just the magnitude of the mismatch between the candidate disparity values and those of the corresponding point in the other image. If both P and P' have been labelled as only monocularly visible, then clearly something is wrong, and the same error is assigned. If a point is visible only from one viewpoint, then the view from the other viewpoint must be blocked by a nearer, and binocularly visible surface.

If only one of P and P' is labelled as monocularly visible, then this is consistent only if the horizontal disparities place this point *further* than the binocularly visible point. In this case, $e_c = 0$, otherwise, the magnitude of the mismatch between disparity values is used.

$$e_c = \begin{cases} 0 & \text{if one point is occluded and further than the other} \\ |\hat{h} + h'| + |\hat{v} + v'| & \text{otherwise} \end{cases}$$

7.5 Smoothness Error

Let $\bar{h}, \bar{v}$ be the representative values of horizontal and vertical disparity computed at a point. These are computed either by a local median filter, within binocularly visible regions, and by a local smoothing operation withing monocularly visible regions. These operations preserve boundaries of binocularly visible surfaces while providing stable depth estimates near occluded regions. The smoothness error, e_s, is used to penalize candidate disparity values $\hat{h}, \hat{v}$ that deviate significantly from their neighbors.

$$e_s = |\hat{h} - \bar{h}| + |\hat{v} - \bar{v}|$$

7.6 Experimental Results

This refined, iterative approach to solving the correspondence problem has been implemented and tested on a variety of natural and artificial images. In practice, this process converges (i.e., stops producing significant changes) in under ten iterations.

Applying the additional constraints described in this chapter, the depth information in random-dot stereograms is easily recovered. Fig. 14 shows the horizontal disparity maps for the left and right views. In the bottom panel, the monocular regions are shown. The reader who is able to fuse this stereo pair can easily verify these.

Fig. 15 shows the results of applying the iterative stereo algorithm to the same stereo pair shown in Fig. 7 Comparing that figure to this one, it is clear that recovering the viewing geometry has

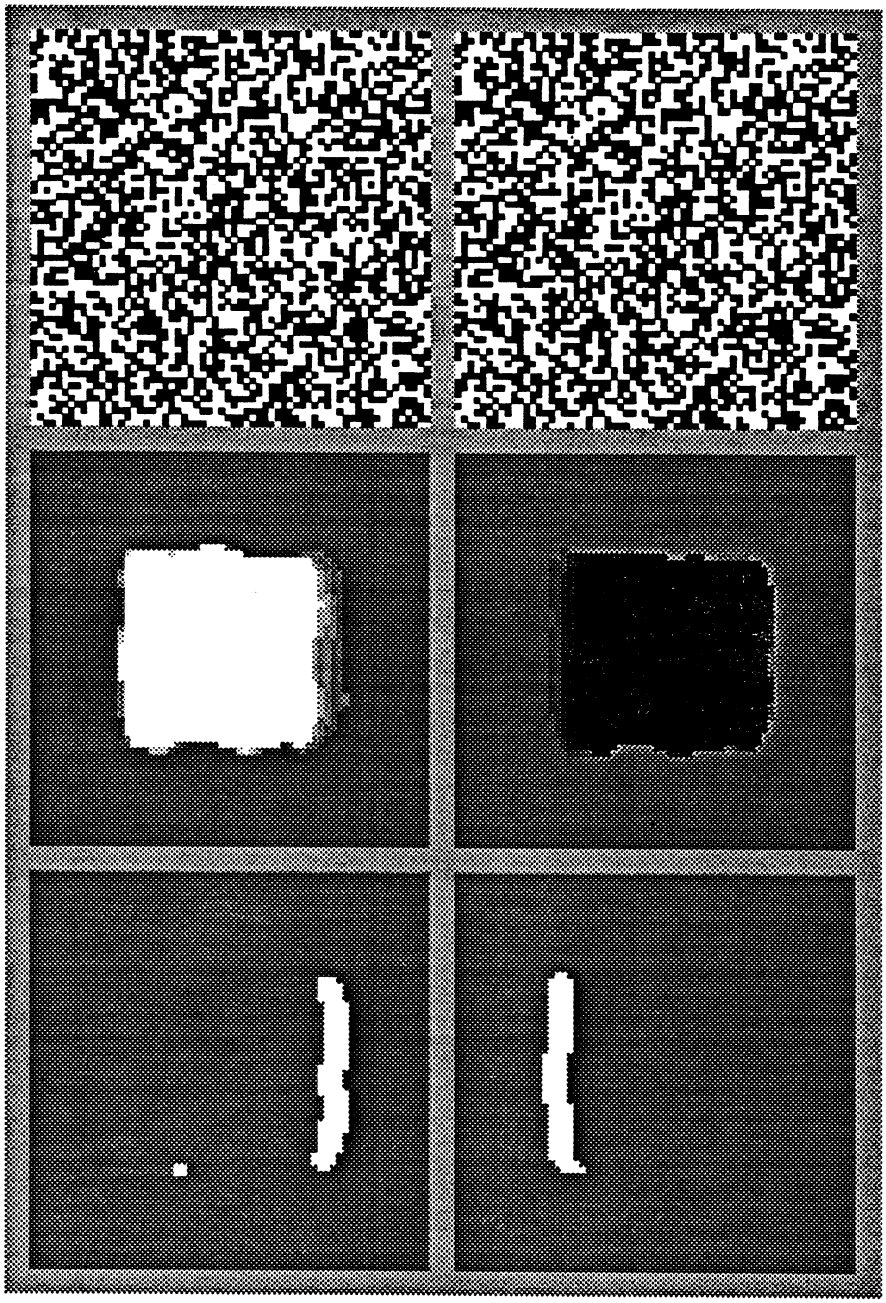


Figure 14: Refined disparity estimates. For the stereo pair shown at the top, the recovered horizontal disparity is shown in the middle. The visibility maps for each view are shown at the bottom. White areas indicate regions seen only in one view.

helped to eliminate some initially poor disparity estimates. The visibility map for this scene is shown in Fig. 11. Recall that in monocular regions, the disparity map has been smoothed to yield a stable solution.

As just another way of looking at the recovered depth information, it is possible to portray, using standard computer graphics techniques, what this scene would look like from another viewpoint (Fig. 16). Some minor errors occur right at depth discontinuities, but otherwise the results are

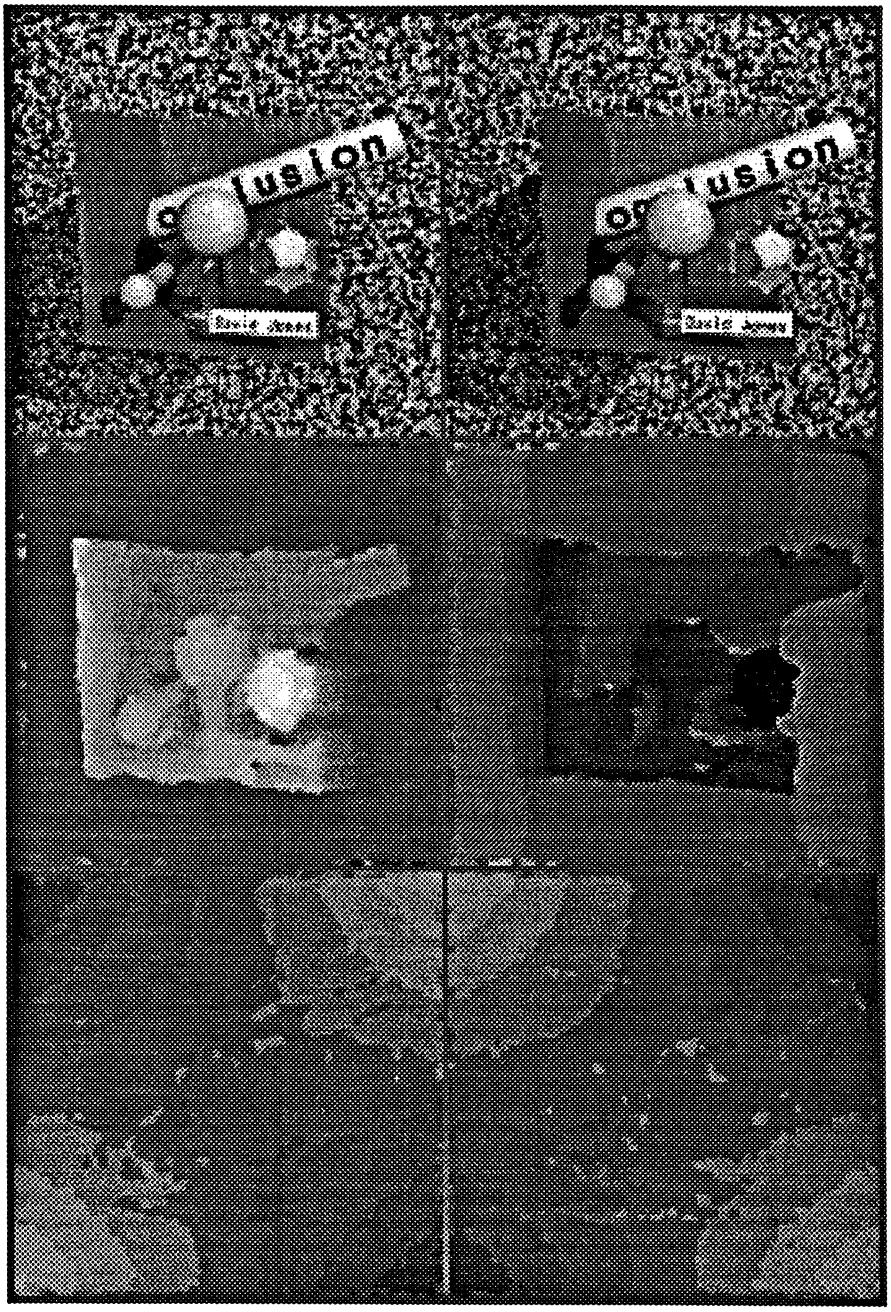


Figure 15: Refined disparity estimates. For the stereo pair shown at the top, the recovered horizontal (*middle*) and vertical (*bottom*) disparities are shown.

quite accurate.

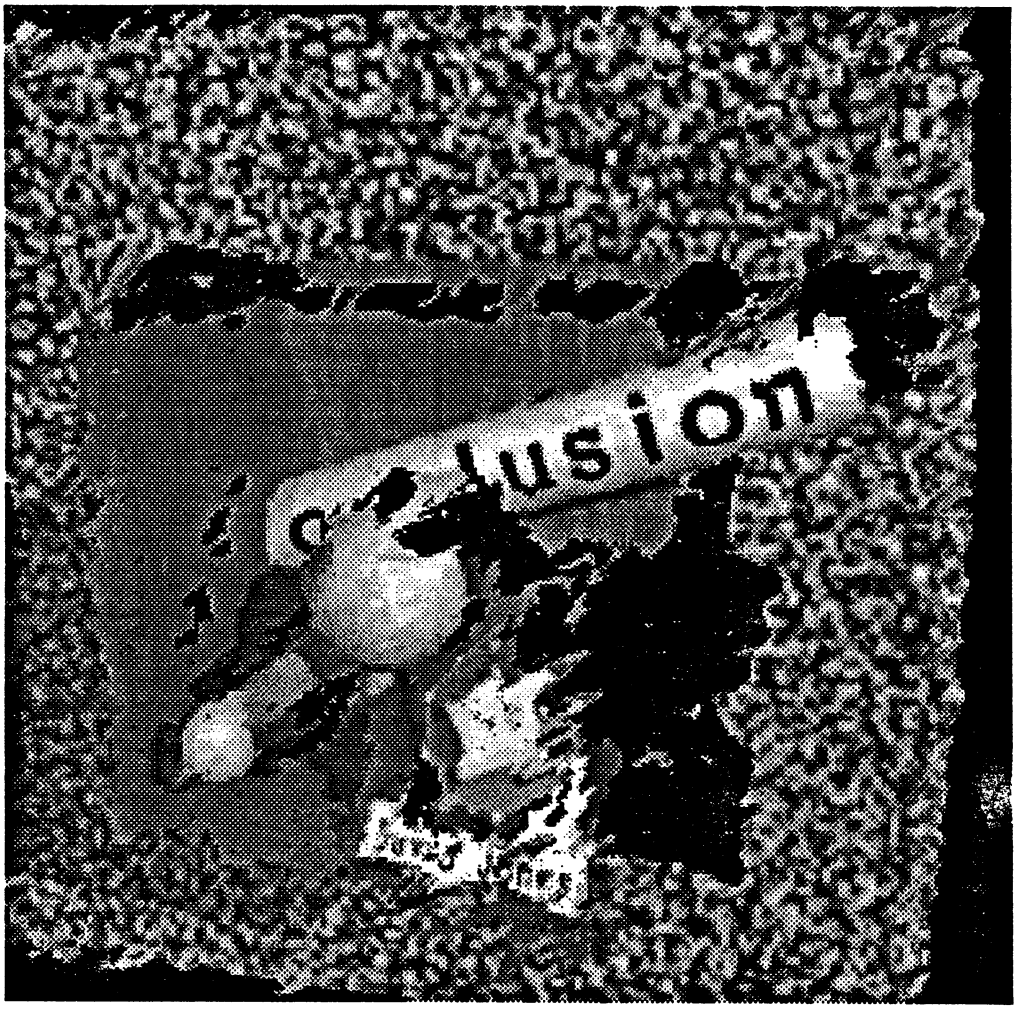


Figure 16: Recovered depth information from an new viewpoint. The depth information shown in Fig. 15 has been used to depict what the scene might look like from a new viewpoint. Only surfaces visible from the original viewpoint are known; other regions appear black.

References

Baker HH, Binford TO (1981) Depth from edge- and intensity-based stereo. Proc 7th IJCAI 631–636

Barnard ST, Thompson WB (1980) Disparity analysis of images. IEEE PAMI 2(4):333–340

Blake A, Zisserman A (1987) Visual reconstruction. MIT press

Burt P, Julesz B (1980) A disparity gradient limit for binocular function. Science 208:651–657

DeValois R, DeValois K (1988) Spatial vision. Oxford Univ Press

Freeman W, Adelson E (1990) Steerable filters for image analysis. MIT Media Lab TR #126

Geman S, Geman D (1984) Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. IEEE Trans PAMI 6:721–741

Gennery DB (1977) A stereo vision system for autonomous vehicles. Proc. 5^{th} IJCAI 576–582

Gillam B, Lawergren B (1983) The induced effect, vertical disparity, and stereoscopic theory. Perception and Psychophysics 36:559–64

Golub GH, Van Loan CF (1983) Matrix computations. The Johns Hopkins Univ Press, Baltimore, MD

Grimson WEL (1981) From images to surfaces. M.I.T Press, Cambridge, Mass

Hannah MJ (1974) Computer matching of areas in images. Stanford AI Memo #239

Jones, DG (1991) Computational models of binocular vision. PhD Thesis, Stanford University

Kemp M (Ed) (1989) Leonardo on painting. Yale Univ. Press: New Haven 65–66

Koenderink JJ, van Doorn AJ (1987) Representation of local geometry in the visual system. Biol. Cybern. 55:367–375

Koenderink JJ (1988) Operational significance of receptive field assemblies. Biol. Cybern. 58:163–171

Marr D, Poggio T (1979) A theory for human stereo vision. Proc. Roy. Soc. Lond. B 204:301–328

Mayhew JEW (1982) The interpretation of stereo disparity information: the computation of surface orientation and depth. Perception 11:387–403

Mayhew JEW (1983) Stereopsis. in Physiological and Biological Processing of Images. Braddick OJ, Sleigh AC (Eds) Springer-Verlag, Berlin.

Moravec HP (1977) Towards automatic visual obstacle avoidance. Proc. 5^{th} IJCAI

Nakayama K, Shimojo S (1990) DaVinci Stereopsis: Depth and subjective occluding contours from unpaired image points Vision Res. 30(11):1811–1825

Numerical Recipes (1987) Cambridge University Press

Perona P (1991) Deformable kernels for early vision. IEEE Proc CVPR 222–227

Poggio T, Torre V, Koch C (1985) Computational vision and regularization theory. Nature 317:314–319

Pollard SB, Mayhew JEW, Frisby JP (1985) PMF: a stereo correspondence algorithm using a disparity gradient limit. Perception 14:449–470

Young R (1985) The Gaussian derivative theory of spatial vision: analysis of cortical cell receptive field line-weighting profiles. General Motors Research TR #GMR-4920