THE TRINOCULAR GENERAL SUPPORT ALGORITHM: A THREE CAMERA STEREO ALGORITHM FOR OVERCOMING BINOCULAR MATCHING ERRORS

Charles V. Stewart
Charles R. Dyer

Computer Sciences Technical Report #768

May 1988

The Trinocular General Support Algorithm: A Three Camera Stereo Algorithm for Overcoming Binocular Matching Errors

Charles V. Stewart

Charles R. Dyer

Computer Sciences Department
University of Wisconsin
Madison, WI 53706

Abstract

The combined use of binocular and new trinocular matching constraints in the Trinocular General Support Algorithm's (TGSA) parallel relaxation computation is shown to overcome many of the problems in binocular stereo matching. These problems include: (1) ambiguity in matching in periodic regions, especially when such a region is partially occluded, (2) noise matches in occluded regions, and (3) missing and erroneous matches due to significant variations between the images. The TGSA employs cameras positioned at the vertices of an isosceles right triangle. Matching takes place between the horizontally aligned pair of images and the vertically aligned pair of images. Reformulated and locally-defined binocular constraints such as uniqueness, coarse-to-fine and fine-to-coarse multiresolution, figural continuity, detailed match and the disparity gradient are implemented for each matching pair. New trinocular constraints, trinocular uniqueness and the trinocular disparity gradient, relate vertical and horizontal matches. When combined in a relaxation algorithm these constraints help to overcome the binocular matching problems listed above. For example, the trinocular disparity gradient provides enough information to directly resolve ambiguity in periodic regions in many cases. The TGSA has been tested on a number of image triples to demonstrate its advantages over previous binocular and trinocular stereo matching algorithms.

The support of the National Science Foundation under Grant No. DCR-8520870 and NASA under Grant No. NAGW-975 is gratefully acknowledged.

Table of Contents

1. Introduction

An important goal of intermediate-level computer vision is to obtain three-dimensional information about a scene. Stereo vision, the matching of images obtained from a pair of cameras with slightly different views of the world, plays an important role in this process. This matching converts the two-dimensional information available in each image into a three-dimensional depth map. However, the process of determining this matching is itself difficult and computationally intensive. For each point in an image there is a large number of possible matching points in the other image. Usually, at most one of these matches is valid. The goal of the matching process is to select the valid matches and eliminate the invalid ones.

Many solutions to the stereo matching problem have been proposed in the computational vision literature. They tend to be partially successful, but they also have a number of problems. First, the algorithms tend to produce spurious matches at certain important locations, primarily at significant occluding boundaries. Second, they are sensitive to noise in the images, including extraneous edges and broken contours. More significant structural differences between the images cause even greater problems. Third, they often have difficulty producing correct matches when image features include repetitive patterns.

Our overall approach toward overcoming these problems and developing a general stereo matching algorithm includes the following steps: (1) Design an algorithm that matches low-level primitives using a number of different constraints. Higher-level primitives are not used because they are hard to reliably extract in general, and they can vary significantly between the images of a stereo pair. Thus, they can not be used as the basis for matching without restricting the domain of images. (2) Test the low-level matching algorithm on a wide range of images, from random-dot stereograms to real images. This will demonstrate the range of success of low-level matching and identify the situations where it fails. (3) Extend the algorithm by incorporating new mechanisms that overcome these remaining problems. The extended algorithm should maintain the successful performance of the original algorithm while improving upon its weaknesses.

In designing a low-level matching algorithm as part of step (1) we proposed the General Support Algorithm.[20] It matched oriented edges using reformulated versions of a number of existing stereo matching constraints: uniqueness, coarse-to-fine and fine-to-coarse multiresolution, figural continuity, the disparity gradient and detailed match. The General Support Algorithm combined the constraints by using, with only one exception, *only positive constraint influences* in selecting valid matches. The algorithm allowed the constraints

to interact cooperatively and in parallel to select valid matches. It was implemented using a *connectionist network.*

Testing the General Support Algorithm showed that it worked well in matching a wide variety of synthetic images and real images of both indoor and outdoor scenes. However, there were a number of matching situations that it could not always accurately handle. These corresponded to situations that *locally defined* constraints could not identify. They included: (1) partially-occluded periodic regions, (2) noise matches an unknown disparities in occluded regions, and (3) errors due to significant structural variations between the images.

In this paper we introduce the Trinocular General Support Algorithm (TGSA). Many of the problems with local binocular matching can be resolved through the use of the TGSA. It employs three cameras positioned on the vertices of an isosceles right triangle (see Figure 2). Matching takes place between the horizontally aligned pair of images as in the binocular case and between the vertically aligned pair of images as well. Binocular matching constraints are implemented between each pair of matching images and, in addition, new trinocular constraints, trinocular uniqueness and the trinocular disparity gradient, are used to specify relations between matches in the two pairs of images. Trinocular uniqueness constrains vertical and horizontal matches for an edge to have nearly identical disparities. The trinocular disparity gradient defines support relations between a pair of nearby vertical and horizontal matches whose distance is less than their disparity difference. These new constraints influence matches in the same way as the binocular constraints. As in our binocular algorithm, the TGSA is a relaxation algorithm that integrates all of the constraints cooperatively, positively and in parallel to select the valid matches.

The TGSA produces a higher percentage of correct matching decisions and overcomes many of the remaining problems in binocular matching: (1) In the case of periodic regions, the trinocular disparity gradient provides additional information to resolve ambiguities when the vertical and horizontal frequencies differ. In addition, when the frequencies are the same, the errors can be reduced if there is vertical or horizontal occlusion, but not both. (2) The algorithm improves the results in occluded regions by allowing matches to be accepted through binocular support only. Thus, in cases where a feature is occluded in one of the other two images, the correct match for that feature may be found. This produces known disparities in the occluded region and reduces both the number and significance of noise matches there. (3) Finally, the matching information from the third image overcomes many of the structural variations between pairs images by using

the trinocular disparity gradient.

In addition to improving the results of binocular matching, the TGSA offers an improvement over previous trinocular algorithms. Other three-camera algorithms employ two general methods for using the additional image. Some use it for confirmation of match disparities and require structured matching primitives.[10,25] In this case simple, general-purpose primitives such as edges do not allow unambiguous match confirmation because of the difficulty of matching along epipolar scanlines. To ensure that the necessary structured primitives are reliably extracted, confirmation algorithms require restricted domains. The second type of approach to trinocular stereo involves matching between two pairs of images separately and then smoothing together the results upon completion.[6,15] Unlike the TGSA, these algorithms do not employ relations between pairs of images during matching. Thus, they can not recover from ambiguities that cause errors in both matching pairs. An example of this is periodic image regions that are partially occluded.

In the remainder of this section we provide definitions for both binocular and trinocular stereo, examine prior trinocular stereo algorithms to show how our algorithm improves upon them, and outline the body of the paper.

1.1. Definitions

There are several stages to the binocular stereo problem. These include (1) determining the imaging geometry, (2) detecting image features for matching, (3) finding the matches between these features (the *correspondence problem*), and (4) interpreting the results. The focus of our work has been the correspondence problem in which a correct match must be selected from among the many possible matches for each feature (matching primitive). These valid matches are identified using **constraints** derived from assumptions about the continuity of surfaces as well as assumptions about the imaging geometry and image formation process.

The standard imaging geometry for the two camera stereo problem is shown in Figure 1. The lines of sight of the cameras are normal to the center of the image plane, intersect the focal point, and are parallel to the x-axis in the $x-z$ plane. Given a point (x_0,y_0) in one image, the **epipolar scanline** for (x_0,y_0) is the line in the other image containing candidate match points for (x_0,y_0). In our model the epipolar scanline is *parallel* to the x-axis for each point. When the matching primitives have an orientation component, those primitives that are parallel to the epipolar scanline are often ambiguous (horizontal edges in our model). In the binocular General Support Algorithm we only matched a horizontal edge when it was relatively isolated from other horizontal

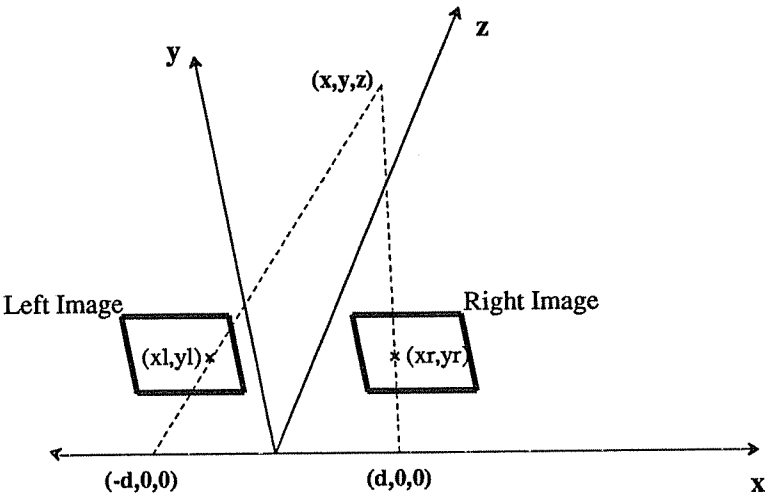

Figure 1. Binocular stereo imaging geometry.

edges. Finally, the **disparity** of a match is the distance between the matching points in the images. It is used to determine the depth of the scene feature corresponding to that point.

The imaging geometry employed in the Trinocular General Support Algorithm is shown in Figure 2. All three cameras are assumed to be identical and are positioned on the vertices of an isosceles right triangle. The lines of sight of the cameras are all perpendicular to the $x-y$ plane. The isosceles triangle assumption ensures that the same vertical and horizontal disparity indicates the same depth. Matches between the base image and the right image are called **horizontal matches**, while matches between the base image and the top images are called **vertical matches**. When we discuss the relationship between a vertical match and a horizontal match we will refer to them as **orthogonal matches.**

The Trinocular General Support Algorithm is implemented using a connectionist network. **Connectionist models** are motivated by the organization of the brain.[5,19] They employ simple computing elements (called nodes) and a massive interconnection network among these nodes. Each connection sends a weighted value of the activation of one node to another node. Each node determines its activation based on its prior activation and on the input it receives from its incoming connections. The overall computation of the network is organized around iterative changes in the activations of the nodes. This continues until the network settles on a solution. This style of computation is similar to relaxation labeling.[9]

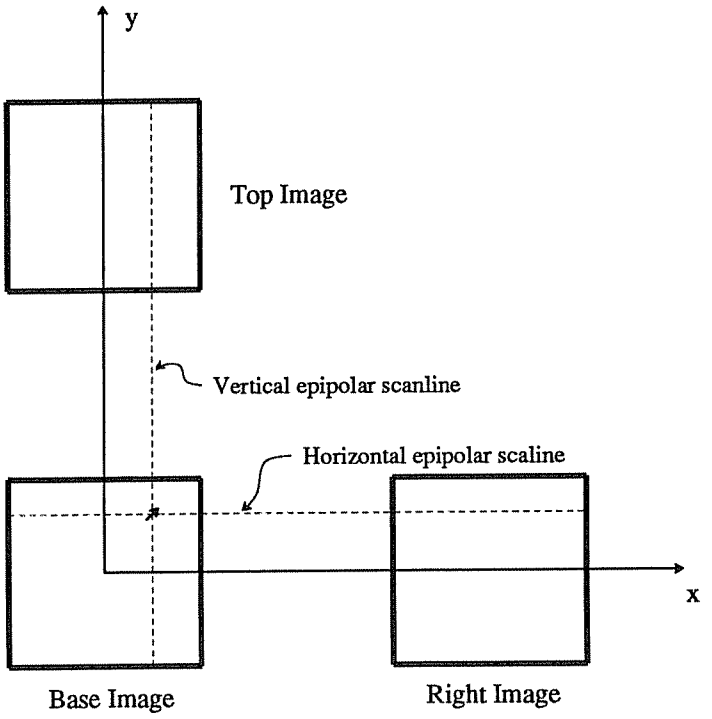

Figure 2. Trinocular stereo imaging geometry. The base image matches both the top and the right images.

1.2. Prior Work on Multi-Camera Matching

Recently there has been a significant amount of interest in the use of more than two cameras for stereo vision. The approaches described in the literature vary in the relative positions of the cameras and in the methods for using the additional information in matching. In this section we review these approaches, highlight some of their weaknesses, and show how the TGSA differs from them.

Most multi-camera stereo algorithms use three cameras positioned at the vertices of a triangle, usually either an equilateral triangle, or an isosceles right triangle. Some algorithms use the additional information simply for eliminating the horizontal matches problem.[16] That is, when an edge or matching segment is vertically oriented, its corresponding match is sought in the horizontal image; when the edge is horizontal the match is sought vertically. While this may produce a denser set of matches, the results are limited by the quality of matching between a pair of images. Therefore, it is susceptible to the same problems as binocular stereo matching.

Another approach uses the third camera to confirm the validity of a match.[10,25] Given a match between two images, the disparity of that match is used to predict the location of the corresponding point in the third image. Unfortunately, matching points may not appear in all three images because of occlusion and errors in the positions of edges (especially those oriented along horizontal or vertical epipolar scanlines). To account for this problem some trinocular algorithms employ more structured matching primitives.[1] These algorithms are not susceptible to the problem of matching along epipolar scanlines because the primitives are two-dimensional. However, they rely on domain-dependent assumptions (usually indoor or urban scenes) in order to allow the matching primitives to be reliably extracted and matched. Another problem with approaches that rely on confirmation of the match disparity is that no matching information is obtained when a feature does not appear in all three images. For these algorithms the reduced number of matches due to occlusions is more severe than in binocular matching.

The final type of approach to using three cameras is to match both vertically and horizontally using an existing stereo matching algorithm, and "smooth" the results together.[6,15] This assumes that the errors arising from the matching between each pair of images do not co-occur. In addition, it assumes that the smoothing process will favor correct matches over errors in a given region. This is not always the case in situations such as periodic regions where the errors tend to be systematic. Finally, this approach ignores the possibility that weaknesses in the support between two images can often be overcome by using the third image during matching.

1.3. Organization of the Paper

The remainder of this paper presents the Trinocular General Support Algorithm. Specifically, Section 2 describes the constraints and the computation employed in the TGSA. Section 3 presents some results of testing the earlier binocular version of this algorithm and outlines the circumstances in which it fails. Section 4 discusses how the TGSA overcomes problems in binocular matching. Section 5 presents our empirical results. Finally, Section 6 reviews the work and highlights additional areas of research.

2. The Trinocular General Support Algorithm

The Trinocular General Support Algorithm (TGSA) offers both an alternative to existing trinocular stereo algorithms and a method for overcoming the errors observed in the binocular General Support Algorithm. It does this by (1) employing binocular matching constraints both vertically and horizontally, (2) defining new

trinocular constraints between orthogonal matches, and (3) combining all of the constraints using a parallel relaxation algorithm. Thus, when a match is occluded only vertically or horizontally but not both, it can be accepted solely through binocular support. On the other hand, in circumstances where binocular support is weak, noisy or inconclusive, the addition of trinocular constraints can help to resolve ambiguous matches.

The TGSA matches oriented edges at multiple resolutions. When a contour in the base image is oriented along either the vertical or horizontal epipolar scanline, the edges along the contour may have matches in only one image (because of the ambiguity in matching edges oriented along an epipolar scanline). In the TGSA these matches are supported through binocular constraints and though the trinocular disparity gradient. Thus, in contrast to many prior trinocular algorithms, a primitive need not be located in all three images to take advantage of the trinocular information in determining its match.

The remainder of this section discusses the TGSA in detail. The reformulation of some existing binocular constraints is discussed in Section 2.1. Section 2.2 defines the two new trinocular constraints, trinocular uniqueness and the trinocular disparity gradient. Section 2.3 describes how all of these constraints are integrated in the TGSA and defines the connectionist network computation that implements the algorithm.

2.1. Binocular Stereo Matching Constraints

This section presents a number of stereo matching constraints that are used in the TGSA and were used in the binocular General Support Algorithm. In developing the binocular algorithm the constraints were reformulated based on an analysis that concentrated on the assumptions behind the constraints and their overall statements. (See Stewart[20] for details.) Except for uniqueness, the reformulated constraints are *locally-defined*, that is, they are defined as pairwise relations between candidate matches. In the binocular General Support Algorithm we relied on the combination of the constraints to produce effective matching. In the TGSA we rely on these same constraints, implemented vertically and horizontally, along with new trinocular constraints, to further improve the matching results.

The next few subsections describe the binocular constraints as they are implemented between matches from the base and right images. Implementing them between matches from the base and top images is straightforward and requires no extra discussion here. We assume that the constraints are incorporated into an edge-based matching algorithm. (Other types of matching primitives are not as general-purpose. Primitives extracted using an interest operator[3] are much sparser than edges. More structured primitives such as long

contours and corners[4, 8] may vary significantly between images.)

In an edge-based algorithm, compatibility predicates are used to determine which edges might match. In our case, a pair of edges, one from each image, form a candidate match if (1) they are in the same image row, (2) their disparity is within a maximum allowed disparity determined by the minimum distance to a point in the scene, and (3) they have compatible orientations.

2.1.1. Uniqueness

The uniqueness constraint has been used either explicitly or implicitly by nearly every stereo matching algorithm.[13] It is stated as follows: *Each matching primitive in an image should match at most one primitive from the other image.* It is derived from the assumption that each image point corresponds to one point in the world. Our use of uniqueness is similar to Marr and Poggio's *one-sided uniqueness.*[11] We define binocular uniqueness in terms of two separate influences on a candidate match between image edges b and r (where b is an edge in the base image and r is an edge in the right image). The first is from the strongest of the other horizontal matches for edge b. The second is from the strongest of the other horizontal matches for edge r. After matching is complete this formulation will usually enforce unique results. It also helps to select matches in ambiguous regions.

2.1.2. Detailed Match

The detailed match constraint states that candidate matches in which the edges have similar properties are more likely to be valid. This follows from the assumption that the images are taken from similar viewpoints, so that edges are likely to have a similar appearance in both images. Edge comparison techniques have been used in a number matching algorithms.[2] In our formulation of detailed match, two intensity comparison measures are employed. The first supports the match when the intensity values on each side of the edges are similar. The second provides support when the intensities on only one side of the edges match.

2.1.3. Multiresolution Matching

Correspondence between matches at multiple resolutions provides another useful constraint. It is based on a number of assumptions about the relationship between edges at different resolutions. It was first proposed by Marr and Poggio,[11] implemented by Grimson,[7] and has been used frequently since then. Our refinements of multiresolution matching involve restricting it to a local definition, using fine-to-coarse as well as coarse-to-fine

matching, and computing it simultaneously at all levels. Specifically, in coarse-to-fine multiresolution, a match at a given resolution supports a match at the next finer resolution when their disparities agree and when the edges in each image are in nearly the same spatial position. This is based on the observation that the position of a particular edge only changes locally between resolution levels.[24] Fine-to-coarse multiresolution is defined similarly. This is an important new use of multiresolution since coarse level matches often receive little support through other constraints.[22] Note that because our definition involves pairwise interactions between matches, many matches will not receive any multiresolution support. However, they will not be eliminated either; they may still be accepted through support from other constraints.

2.1.4. Figural Continuity

The figural continuity constraint was developed by Mayhew and Frisby.[14] It states that edges along a contour should match edges along a similar contour in the other image. It is based on two assumptions: (1) contours in the scene appear nearly the same in each image, and (2) contour extraction and matching can occur cooperatively. This is the Binocular Raw-Primal Sketch (BRPS) conjecture.[14] Our formulation of figural continuity defines pairwise relations between matches. The edges in each image must appear to be part of a contour and the hypothesized contours, as defined by the pairs of edges, must appear similarly. Because this pairwise definition is accurate only over a limited spatial extent, we restrict figural continuity to nearby matches.

2.1.5. Disparity Gradient and Gaussian Support

Both the disparity gradient[17] and Prazdny's Gaussian-support model[18] are based on the assumption that objects occupy a well-defined three-dimensional volume. Both constraints assume that points from the surface of an object appear with relatively similar disparities in an image. The disparity gradient is formally defined as follows. Suppose that $m_1 = (b_1, r_1)$ and $m_2 = (b_2, r_2)$ are horizontal candidate matches. Also, let $d\ (m_i)$ be the disparity of a match. Then the matches support each other if

$$\frac{|\, d\ (m_1) - d\ (m_2)\, |}{D\,(m_1, m_2)} \leq 1$$

where $D\ (m_1, m_2)$ is the distance between two matches. (The value 1 is the disparity gradient limit.) According to this definition, nearby matches must have similar disparities. More distant matches can have a greater disparity difference.

The disparity gradient is much less discriminating in its definition of support than other constraints. It often provides only weak discrimination between matches, and contrary to its assumption, it allows support between matches across surface boundaries. To make it more discriminating we will skew the strength of the relations it defines in favor of matches for fronto-parallel surfaces. Also, we reduce its strength relative to the other less ambiguous constraints, i.e. multiresolution and figural continuity. Finally, because the disparity gradient involves no orientation information, we use figural continuity to specify relations between matches for edges that locally appear to be part of a contour. Thus, figural continuity and the disparity gradient complement each other in their definition of support.

2.2. Trinocular Constraints

In addition to the binocular matching constraints discussed above that are implemented both vertically and horizontally, the TGSA incorporates two new trinocular constraints. Computationally these constraints are used in a similar manner as the binocular constraints. The only difference is that the trinocular constraints are implemented between orthogonal matches. The new constraints are discussed in the next two subsections below.

2.2.1. Trinocular Uniqueness

The trinocular uniqueness constraint is based on the same assumption as binocular uniqueness. In the binocular case we assume that each edge point corresponds to one scene feature, so there should be at most one valid match per edge. Extending this to the trinocular case, we assume that the valid matches found both vertically and horizontally for a given edge should correspond to a single scene feature. Thus, because the cameras are positioned on an isosceles right triangle, these matches should have the same disparity.

Using these assumptions we define the trinocular uniqueness for a horizontal match $m = (b, r)$, where b is the point in the base image and r is the point in the right image, as

$$TU_m = \beta_t \times \max(O_n \mid n \in Top_b \text{ and } |d(m) - d(n)| > \varepsilon)$$

where Top_b is the set of vertical matches for base image point b, β_t is a constant that scales the strength of the relation, $d(m)$ is the disparity of a match, and ε is a small constant. (See Section 2.3.3.2 for a similar statement of binocular uniqueness.) In other words, the inhibiting trinocular uniqueness for a horizontal match

is the maximum of the vertical matches for the base image edge whose disparities differ from the horizontal match disparity by more than a small constant. The trinocular uniqueness for a vertical match is defined similarly.

There are several important points to note about this definition of trinocular uniqueness. First, ideally, the constant ε should be 0 to reflect the fact that the vertical and horizontal disparities should be identical. However, because of potential errors in the position of an edge, there is a slight tolerance for minor differences in disparity. Second, there is *only one* trinocular uniqueness influence on a given match, whereas there are *two* binocular influences on a match. Finally, and most importantly, a large percentage of the matches have no trinocular uniqueness. In binocular matching, edges oriented along an epipolar scanline are not matched unless they are isolated from other horizontal edges. Empirically we found that approximately 30% of the edges in random-dot stereograms were not matched because of this. reason Therefore, in the trinocular algorithm approximately 60% of the edges in the base image will have only vertical or only horizontal matches; approximately 40% of the edges will have matches both vertically and horizontally. These are the only matches that will be influenced by trinocular uniqueness.

2.2.2. Trinocular Disparity Gradient

The trinocular disparity gradient (TDG) is motivated by the same assumptions as the original disparity gradient. The main assumption is that points from a surface appear at similar positions and with similar disparities in the images. Thus, the trinocular disparity gradient defines relations between nearby, orthogonal matches (one match is vertical and the other is horizontal) based on the relative distance between them and on the difference in their disparities.

In defining the trinocular disparity gradient equation, we generalize the original disparity gradient. In the latter, matches $m_0 = (l_0, r_0)$ and $m_1 = (l_1, r_1)$, where the l_i are left (or base) image points and the r_i are right image points, will support each other if

$$\frac{| d(m_0) - d(m_1) |}{D(m_0, m_1)} \leq 1.0$$

where $d(m)$ is the disparity of a match, and D is the distance between matches. This distance is computed by taking the midpoint between the image points for each match, and measuring the distance between these midpoints as shown in Figure 3. To obtain the distance between vertical matches we apply a similar measure.

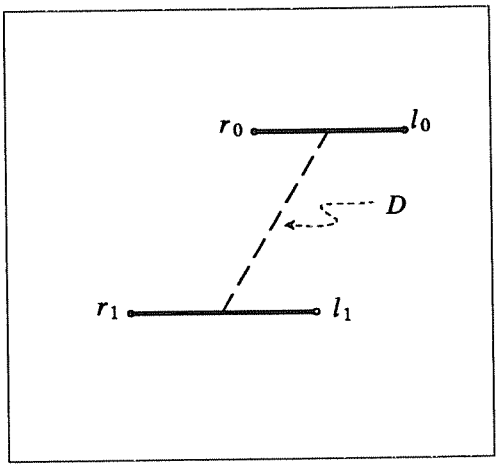

Figure 3. The distance D between two matches in the disparity gradient is the distance between the midpoints of the matches. In this figure the matching points in the two images are graphed in the same coordinate system. (The distance is measured similarly for a vertical implementation of the disparity gradient in matching the base and top images. However, the figure would be rotated by 90^0.) Thus, because the lines of sight of the cameras are parallel, the right image points appear *to the left* of the left image points.

The straightforward geometric generalization of the distance relation to the trinocular disparity gradient is shown in Figure 4. The problem with this definition is that the line segment between the points forming the vertical match and the line segment between the points forming the horizontal match are orthogonal. When these segments cross, the distance between midpoints is no longer an accurate measure of the desired property.

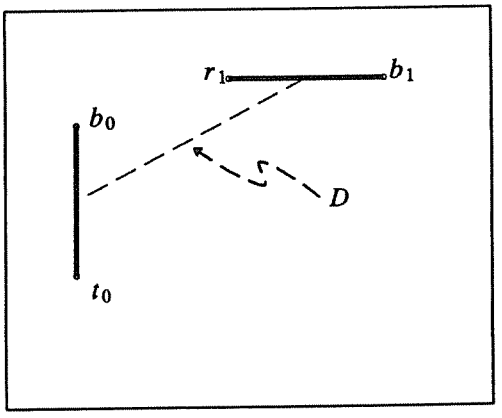

Figure 4. Measuring the distance between matches in the trinocular disparity gradient using a similar measure as in the original disparity gradient. The b_i are points in the base image; r_1 is a point in the right image; t_0 is a point in the top image.

An example of this is seen in Figure 5.

We resolve the problem by examining the distance equations. Let $b_0 = (x_0, y_0)$ and $b_1 = (x_1, y_1)$ be points in the base image and let d_0 and d_1 be the disparities for those points. When these points and disparities define horizontal matches, the distance between the matches is (letting $\Delta d = d_0 - d_1$, $\Delta x = x_0 - x_1$, and $\Delta y = y_0 - y_1$):

$$D_h = \sqrt{\Delta y^2 + (\Delta x - \Delta d / 2)^2}.$$

If the points and disparities describe vertical matches, then the distance between the matches is:

$$D_v = \sqrt{(\Delta y - \Delta d / 2)^2 + \Delta x^2}.$$

There is no logical inconsistency between these two distance measures. This is because the image pairs, including the epipolar scanlines, are rotated 90^0 with respect to each other. We adapt these equations to the trinocular disparity gradient as follows. If the points and disparities above describe one vertical and one horizontal match, then the simplest distance measure between the matches is:

$$D_t = \sqrt{\Delta y^2 + \Delta x^2}.$$

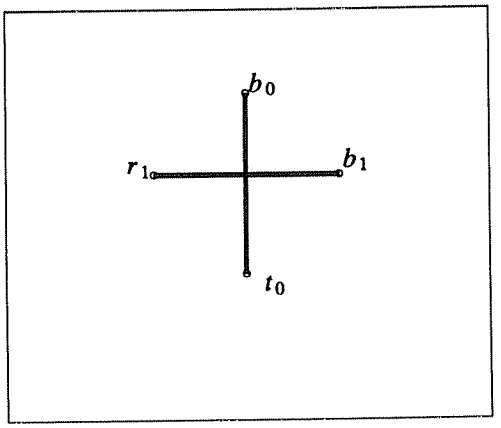

Figure 5. An example where the straightforward extension of the disparity gradient distance definition is problematic. The disparity difference between these matches is quite small, but the measured distance is 0. Thus, this distance measure does not accurately reflect the true separation of the underlying scene points. A more accurate measure of distance is shown in Figure 6.

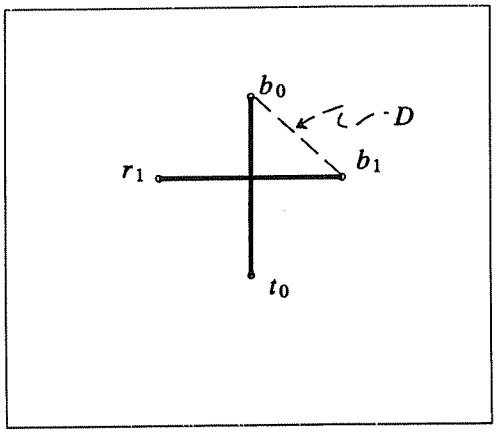

Figure 6. The trinocular disparity gradient distance between a vertical and a horizontal match.

Because one vertical match and one horizontal match are used, the difference in disparity does not enter into either the x distance or the y distance. This is shown geometrically in Figure 6.

The final problem in defining the trinocular disparity gradient concerns the relationship between vertical and horizontal matches for a given point in the base image. Specifically, because the distance between the matches is 0, the trinocular disparity gradient equation is undefined. This situation is handled by defining the minimum distance in trinocular disparity gradient between matches to be 1. In addition to handling the problem, it introduces a minor tolerance in the difference between the vertical and horizontal match disparities for an edge. Note that when the disparity difference is greater than 1 for a pair of orthogonal matches for the same edge, they may inhibit each other through trinocular uniqueness.

2.3. Constraint Integration in the TGSA

None of the binocular or trinocular constraints discussed above provides accurate matching information in all circumstances. To reduce the false support made by the binocular constraints, we redefined a number of them, yielding local (pairwise) definitions of each constraint. The trinocular disparity gradient is also locally defined. Our next step is to describe the integration of these constraints in the trinocular algorithm. The important features of this are: (1) the local definition of each constraint, (2) the *cooperative, parallel* integration of a number of constraints using only *positive* constraint influences, (3) the combination of binocular and trinocular constraints in a single matching network, and (4) the implementation of the algorithm in a connectionist network.

The TGSA is essentially an iterative, parallel, relaxation algorithm. In the algorithm, each candidate match is represented by a node whose strength represents the strength of the support for that match. The candidate matches are formed by pairs of compatible edges from the base and top images, and from the base and right images. Using both the binocular and trinocular constraints, each candidate match propagates its strength to other matches, and in turn is influenced by those matches. This is repeated until the TGSA determines which matches are valid. Once the relaxation algorithm is complete the vertical and horizontal disparities of the matches for the edges in the base image are smoothed together to yield a single disparity map. (We only discuss this last part briefly at the end of Section 5.)

2.3.1. The General Support Principle

The TGSA organizes the influence of the constraints using the *General Support Principle (GSP)*. This principle states that, except for binocular and trinocular uniqueness, *constraints are only used to support candidate matches*. We provide both theoretical and practical reasons for defining the GSP and using it to organize the influence of the constraints. Our theoretical observation is that the constraints are derived from assumptions about objects and their appearance in images. These assumptions lead to assertions concerning relationships between valid matches. A more practical justification for the GSP follows from the local definition of the constraints. Specifically, since the constraints are defined as pairwise relations between candidate matches, there is no way to locally distinguish between the following possibilities: (1) the constraints are outside the assumptions of the constraints, or (2) one or both of the matches are invalid.

2.3.2. Constraints Used

The TGSA incorporates all of the constraints discussed in Sections 2.1 and 2.2. Binocular constraints are implemented between matches from the base and top images, and between matches from the base and right images. Trinocular constraints are implemented between orthogonal matches. Uniqueness is necessary for any stereo matching algorithm matching primitives as simple as edges since there are usually multiple candidate matches for each edge. Trinocular uniqueness helps to ensure consistency between vertical and horizontal matches for a given edge in the base image. The disparity gradient, trinocular disparity gradient and figural continuity are complementary surface structure constraints. The trinocular disparity gradient relates orthogonal matches thereby providing additional information that can resolve conflicting matches and overcome errors due to noise. Multiresolution is most useful for identifying persistent features of the images and propagating

support between appropriate matches for these features at various resolution levels. Detailed match is a comparison of the properties of the two edges involved in a match. Thus, the TGSA employs structural, hierarchical and edge-appearance constraints in matching.

2.3.3. Connectionist Implementation of the TGSA

Matching in the TGSA is implemented using a connectionist network. The network is designed so that (1) each node represents a distinct potential match, and (2) the constraints are implemented directly in the connections between candidate match nodes. The matching works iteratively. During each iteration each node computes its new *activation* and *output* values. The new activation is based on the old activation, a decay rate, the supporting input and uniqueness.

2.3.3.1. Node Activation and Output Functions

The activation of a node depends on the combined influences of the supporting constraints, uniqueness, decay and the node's prior activation. The supporting input to a match node, i, is a weighted, linear combination of the constraint input, given by:

$$I_i = \sum_{j=1}^{N_i} O_j \, w_{ji}$$

where the O_j are outputs from other match nodes, and the w_{ji} are the connection weights. Note that no distinction is made at this level between the influences of the trinocular and binocular support constraints. Uniqueness input for a horizontal match is given by:

$$U_i = \beta \max(O_j \mid j \in Base_i) + \beta \max(O_k \mid k \in Right_i)$$

where $Base_i$ is the set of competing matches for the base edge of match i, and $Right_i$ is the set of competing matches for the right edge. A similar algebraic statement defines the uniqueness input to a vertical match. The definition of trinocular uniqueness was given in Section 2.2.1. The new activation is:

$$A_i = (1 - \delta) A_i + I_i + U_i + TU_i$$

where δ is a decay factor and TU_i is the trinocular uniqueness value as defined in Section 2.1.1. It is used to help eliminate noise matches. The output, O_i, is simply a threshold function of the activation, A_i. The activation is limited to the range [-1..1], and the output is allowed to be in the range [0..1]. When the activation of a node reaches 1.0 it is said to be *saturated*, and it remains at 1.0 for the duration of the matching procedure.

2.3.3.2. Constraint Weight Equations

As noted previously, most of the constraints are implemented locally as pairwise relations between candidate matches or as functions of individual matches. In this subsection we consider the weight functions of some of the constraints. (The uniqueness constraint and the trinocular uniqueness constraint are built directly into the activation function for a match node described previously.)

For the *disparity gradient*, two matches, m_i and m_j (m_i and m_j are both vertical matches or they are both horizontal matches), support each other if they are within the disparity gradient limit as defined in Section 2.1.5 and if they are within a maximum allowed distance. The weight of a connection is given as follows. Let $d_{i,j}$ be the disparity difference and let $D_{i,j}$ be the distance between the matches. Then the weight is:

$$w_{i,j} = \frac{BASE\_DG}{D_{i,j}} * \frac{c}{(d_{i,j} + c)}$$

where $c \geq 1$ is a constant, and $BASE\_DG$ is a parameter controlling the overall strength of the disparity gradient. The $c\ /\ (disp\_diff + c)$ term scales the support as the disparity difference between the matches increases.

For the *trinocular disparity gradient* let v_i and h_j be orthogonal matches that meet the definition given in Section 2.2.2. The weight of the connection between them is defined as follows. Let $d_{i,j} = |\, d(v_i) - d(h_j)\,|$. Then the weight is given by:

$$w_{i,j} = \frac{BASE\_TDG}{D(v_i, h_j)} * \frac{c}{(d_{i,j} + c)}$$

where c is used as above. (Recall that the binocular and trinocular disparity gradient distance functions differ slightly.)

The weight equations for the other constraints are straightforward. Figural continuity is weighted inversely by the distance between two supporting matches and is scaled using the parameter $BASE\_FC$:

$$w_{i,j} = \frac{BASE\_FC}{D_{i,j}}.$$

The weights for the two multiresolution constraints do not involve a distance term and are the constants $BASE\_COARSE\_TO\_FINE$ and $BASE\_FINE\_TO\_COARSE$. The weights for detailed match are also constants and are relatively small compared with the other constraints.

2.3.3.3. Weight Assignment

The remaining problem in defining the network is to establish values for all of the parameters and thereby set the connection weights. These parameters are associated with the implementation of the constraints and the definition of the activation function, including the uniqueness parameters, β and β_t, and the decay parameter, δ. Our approach to parameter assignment involves three steps: (1) develop qualitative heuristics for the relative values of the parameters and their approximate values, (2) use simple synthetic images representing difficult matching problems to refine the values, and (3) fine tune the values through experimentation with random-dot stereograms and real images. For details of the weight assignment analysis, see Stewart.[21] It is important to note that the algorithm was tolerant of minor changes in the values of the parameters.

We developed the following heuristics for the trinocular constraints. First, the trinocular uniqueness parameter β_t must be set fairly low. This follows from the observation that approximately 60% of the edges in the base image will have only vertical or horizontal matches. For an edge with both vertical and horizontal matches there is only minor additional support that its matches will receive as a consequence. This will be trinocular disparity gradient input from an orthogonal match for the edge. Thus, matches for edges having both vertical and horizontal candidates should not be penalized by strong trinocular uniqueness input.

The parameter *BASE_TDG* is given a similar value as *BASE_DG*, the binocular disparity gradient parameter. This is because both constraints gather support by searching over a small neighborhood around a match for other matches with similar disparities. The disparity gradient searches for support from matches between the same pair of matching images, while the trinocular disparity gradient searches for support from orthogonal matches. The strengths of these two constraints are lower than coarse-to-fine multiresolution and figural continuity because the disparity gradient constraints are much less discriminatory in their definition of support.

2.4. Size of the Trinocular General Support Algorithm Network

As a final consideration in the definition of the TGSA, we examine the size of the network defined by it. Let the number of nodes defined by the original binocular GSA be N_b and let the number of connections be C_b. From Stewart[21] we know that $N_b = k\,L\,M^2$, where k is the number of distinct disparity values represented in the network, L is the number of resolution levels, and the images are each M *by* M. In the trinocular algorithm the number of nodes is: $N_t = 2 * N_b$. This is because the number of nodes in the TGSA network is

equal to the number of nodes in two separate GSA networks.

3. Binocular Results

The binocular General Support Algorithm was previously tested on a wide variety of random-dot stereograms and natural images. The results demonstrated the algorithm's success on a wide range of image pairs. They also led to an understanding of where it failed to produce accurate results. For the most part these problems are common to all current stereo matching algorithms.

Figures 7 through 10 show samples of the binocular matching results. See Stewart[21] for complete details. The results of matching were recorded in two ways. First, the disparities of the matches that the algorithm accepted were used to generate intensity images. Second, statistics were collected to determine the percentage of correct matching decisions (finding a correct match for an edge or correctly rejecting all matches for it). For random-dot stereograms this was done automatically. For real images it required subjective judgment. An

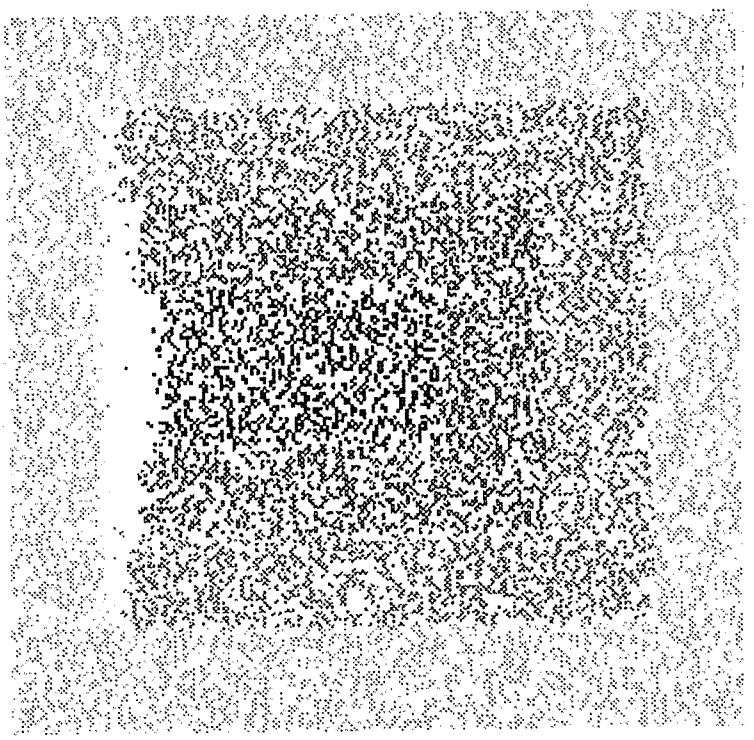

Figure 7. Layered stereogram matching results. The disparity found is encoded by intensity where dark points represent large disparities. There are four layers in the stereogram. The blank areas indicate the reduced number of matches for points that appeared in the left image, but were occluded in the right.

example of the matching results for a multilayer stereogram is shown in Figure 7 (98% correct decisions). Two examples of matching natural image pairs are shown in Figures 9 (98% correct) and 10 (93% correct, this was by far the worst result). Overall, the algorithm averaged 97% correct decisions. Figure 8 graphs the percentage of nodes with intermediate output values as a function of the number of iterations. This shows that the binocular algorithm converged quite quickly.

In spite of the high-quality results there were still some situations that locally-defined binocular constraints could not handle. These corresponded to problems that could not be explicitly identified using the constraint definitions. Thus, the algorithm had to resolve the problems implicitly. The problems were the following:

- Partially-occluded periodic regions. The constraints relied on structural or hierarchical variations in support to disambiguate the matches for an edge. In the middle of periodic regions these variations did not occur. Thus, the GSA relied on matching results at the boundaries of these regions to resolve the ambiguity. The binocular uniqueness formulation was explicitly designed to help this occur.

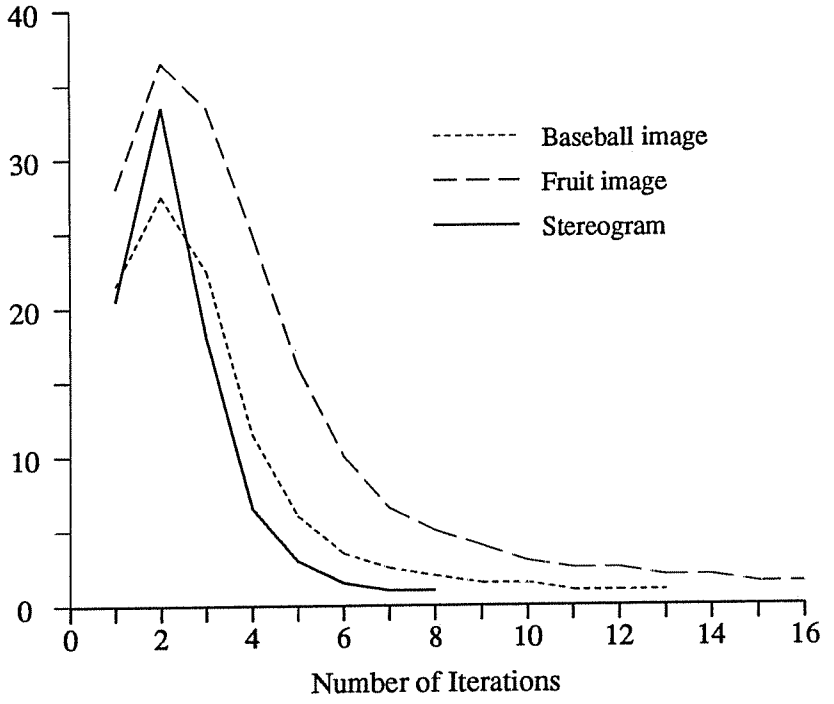

Figure 8. The percentage of nodes with intermediate activations as a function of the iteration number for all of the examples in Section 3. A node is in the intermediate range if its output is between 0.25 and 0.75.

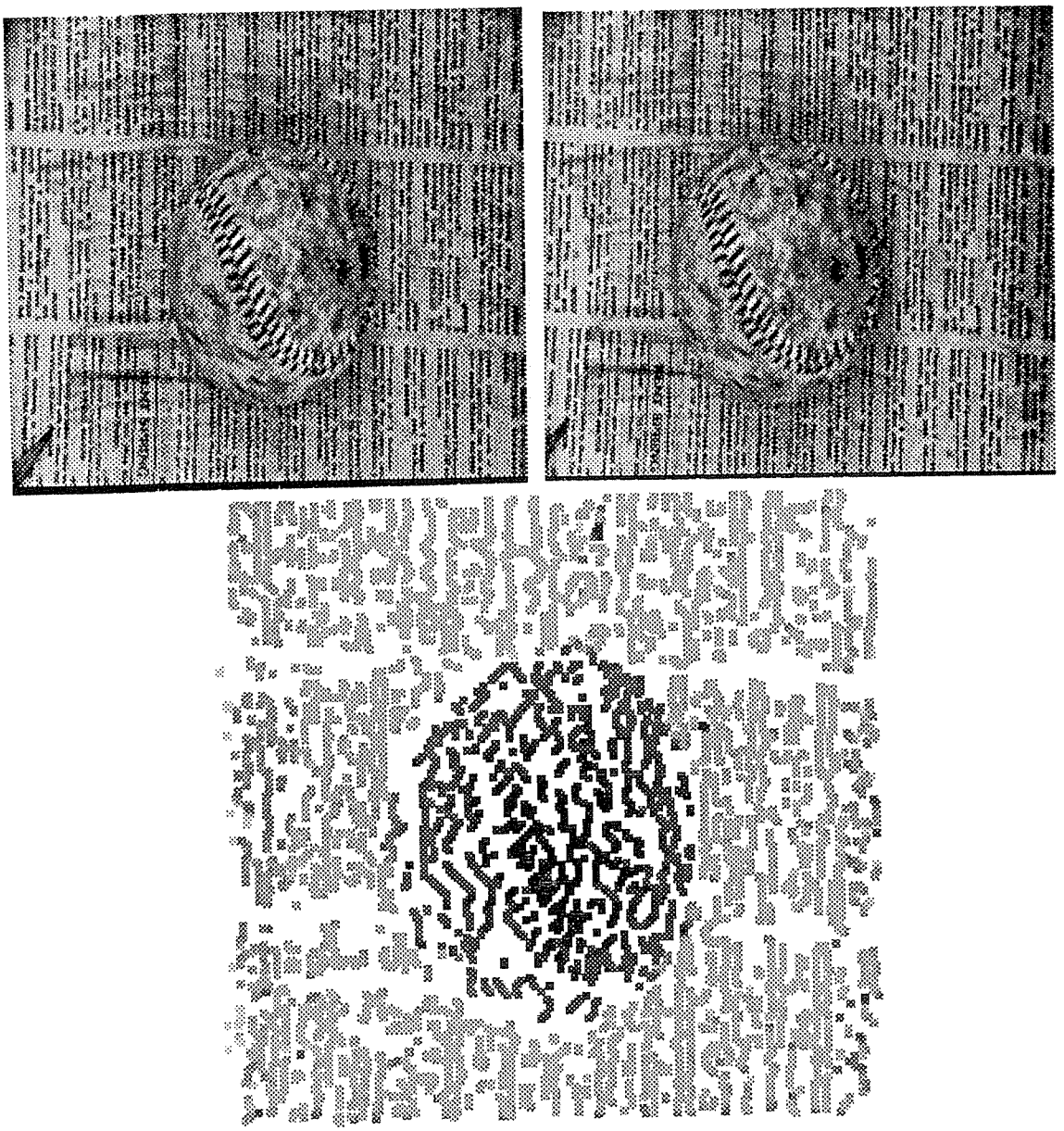

Figure 9. Matching example with a baseball on a newspaper. The disparity results from the perspective of the left image are shown. The algorithm made approximately 98% correct matching decisions for this example.

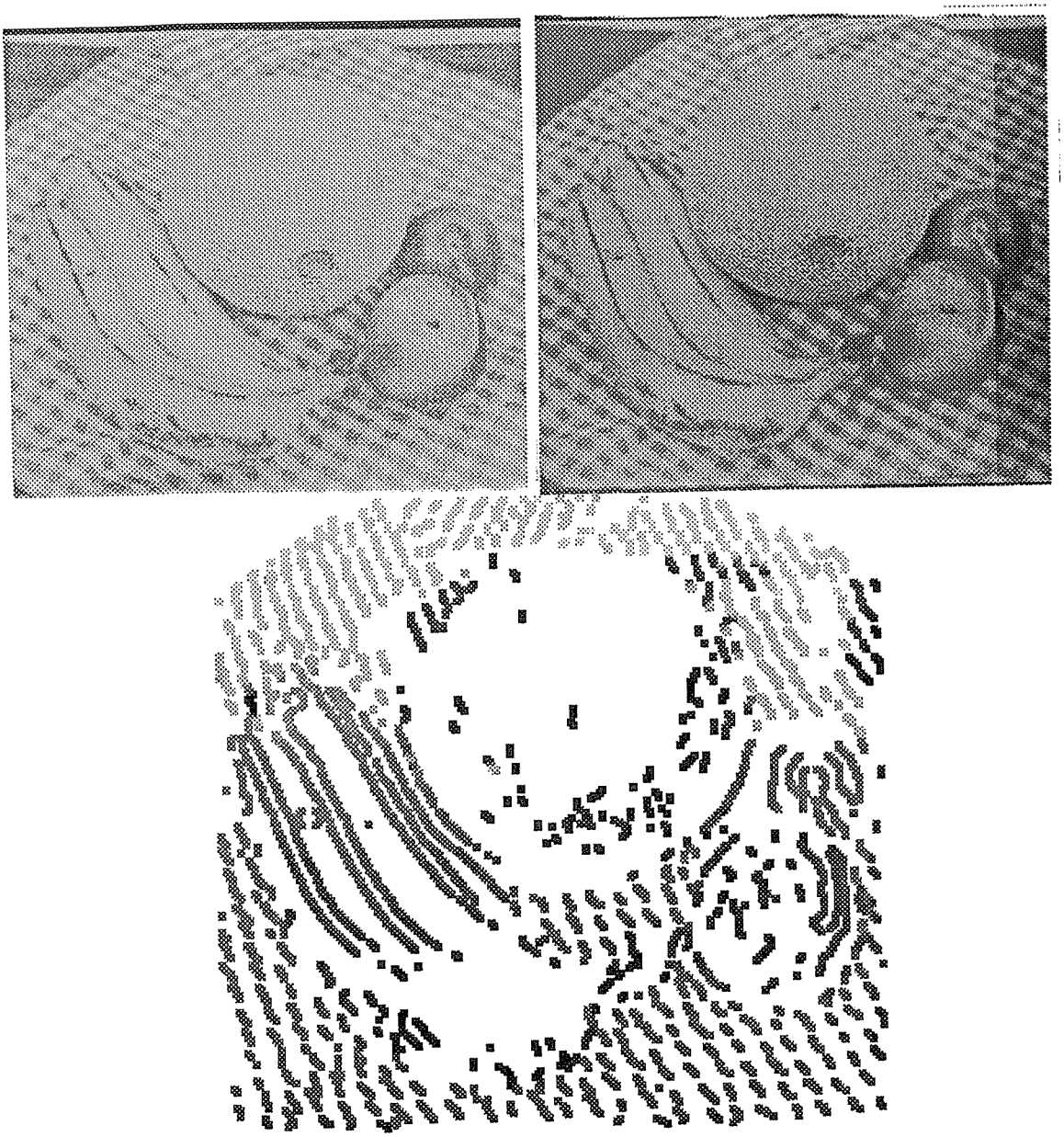

Figure 10. Matching example with fruit in front of a tablecloth. The disparity results from the perspective of the left image are shown. The algorithm produced 93% correct matching decisions here, by far the worst of any of the real images tested. Most of the errors were caused by a significant occluding surface in front of a periodic pattern.

Unfortunately, when a region was partially occluded, as in the fruit image (Figure 6), the boundaries that aligned in each image were often incorrect, and so errors occurred.

- Occluded regions. The GSA was fairly successful in eliminating matches for edges in occluded regions (regions that appear in one image but not in the other). Examples of this are seen in the blank regions to the left of the occluding surfaces in Figures 7 and 9. However, matches could occur between occluded edges and noise edges that were supported strongly enough to be accepted. The main reason for this

problem was support that crossed occluding boundaries.

- Significant structural variations between images. Structural differences caused two types of errors. First, a correct match may not have received enough support because of the reduced similarity between the images. Second, an incorrect match may have been accepted for an edge when the correct match was missing.

These problems caused most of the errors that occurred in the binocular results. Of the problems, occluded periodic regions caused the greatest difficulty because the errors appear in clusters (see Figure 10). In the remainder of this paper we consider how the three cameras used in the Trinocular General Support Algorithm overcome many of the above errors. These solutions arise naturally from matching both vertically and horizontally using binocular constraints along with the new trinocular constraints that relate vertical and horizontal matches.

4. Solutions to Problems in Binocular Matching

The Trinocular General Support Algorithm (TGSA) was designed to solve a number of the problems in matching binocular image pairs. These include ambiguities in matching periodic regions, especially those that are partially occluded, noise matches near occlusions, and erroneous matches when there are significant structural differences between the images. In addition, the matching of horizontal edges is not a problem in the TGSA since it matches both vertically and horizontally. When an edge in the base image is part of a horizontal contour, it will have candidate matches only with edges in the top image. Instead of requiring direct confirmation in the right image, one of these matches may be accepted through support defined by binocular and trinocular constraints.

The most dramatic improvement that the TGSA offers is in handling the periodic structures problem. As the fruit example showed in Section 3, in binocular matching this problem can result in clusters of erroneous matches near an occluding boundary. These errors arise because there are no variations in support in periodic regions that allow the matches to be distinguished. Therefore the disparities for matches found at the boundaries of these regions must be propagated into the interior of the region to resolve the ambiguities. When the boundary is occluded, boundary matches may be incorrect and errors may occur.

The TGSA can overcome problems in matching periodic regions through the trinocular disparity gradient. In a periodic region where there is either is no vertical periodicity or where the vertical and horizontal

frequencies are not equal (e.g. a brick pattern), the trinocular disparity gradient provides stronger support for the valid matches than it does for the invalid matches. This differential implies that the ambiguity in structural support that appeared in binocular matching no longer exists, and the valid matches can be identified without relying on the propagation of the disparities found at the boundaries. In addition, when the vertical and horizontal frequencies are equivalent, the trinocular disparity gradient can help to overcome ambiguities since region boundary matching results are propagated from all four boundaries instead of two. Thus, if only one boundary is occluded, fewer errors are likely to occur.

The following shows how the trinocular disparity gradient establishes the differential in support necessary to resolve the ambiguities in periodic regions. Assume that the correct disparity in a periodic region is d and the vertical and horizontal frequencies in the region are v and h, respectively. The competing horizontal matches will have disparities $d-h$, d and $d+h$, etc. The competing vertical matches will have disparities $d-v$, d and $d+v$, etc. Through the binocular disparity gradient each of the competing disparities for an edge on the interior of the region will receive approximately the same support. However, the trinocular disparity gradient will support the matches at disparity d much more strongly than the others because there are both vertical and horizontal matches with this disparity. (This assumes that h is not an integer multiple of v and v is not an integer multiple of h.) The incorrect matches receive much less support through the trinocular disparity gradient.

The TGSA also solves some of the other problems in binocular matching. In matching only the base and right images there are a number of regions in the base image that do not appear in the right image. There are no correct matches for edges in these regions, but in binocular matching noise matches are occasionally accepted for them. When a third camera is added, some of these occluded regions in the base image will appear in the top image. However, in general there are more regions in the base image that are occluded in one of the other two images (we call these regions half-occluded). Because of this, the trinocular algorithm must be able to resolve matches using the information available only from matches in either the right or top images. This is accomplished through the use of binocular constraints. Thus, the TGSA finds correct disparities in half-occluded regions and eliminates the problem of arbitrary noise matches found there.

The final problem with binocular matching involves significant appearance differences between the images. When these differences arise from random variations, the use of the additional image can improve the results of matching. The improvement obtained in overcoming stochastic noise through the use of many images

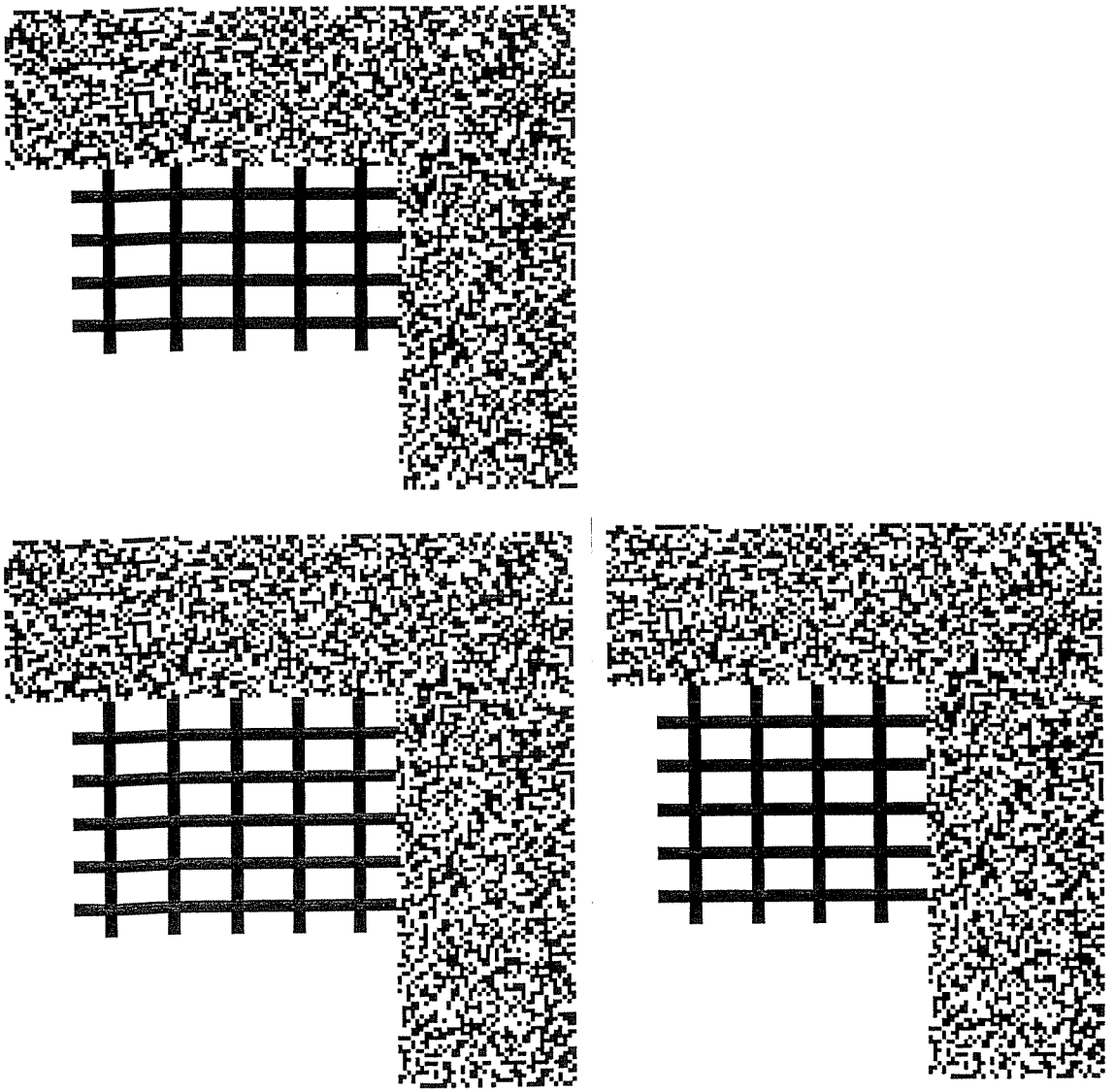

Figure 11. Top, base and right images for the first trinocular example. The random-dot texture is at disparity 16 and the bars are at disparity 4. The top bar is occluded in the top image. The right bar is occluded in the right image.

in matching has been demonstrated by Tsai.[23] In the TGSA the noise variations are overcome through the use of trinocular constraints. Trinocular uniqueness helps to suppress noise matches that occur when the correct match is missing for an edge since the support for such matches is usually relatively weak. Because the trinocular disparity gradient gathers support from orthogonal matches, it can strengthen matches whose support is eroded by noise variations between binocular image pairs. This orthogonal support may also be helpful in overcoming errors that are more systematic as well.

5. Experimental Results

The previous section argued that the Trinocular General Support Algorithm overcomes many of the problems inherent in binocular matching. In this section we present experiments demonstrating the TGSA's success on synthetic images. The first group of images, discussed in Section 5.1, presents results on matching occluded, periodic regions. The second group of images, discussed in Section 5.2, describes the algorithm's performance on random-dot stereograms involving occlusions and noisy data.

The first step in testing the TGSA was designing an efficient, realistic simulation of it. The main observation used in doing this was that matches are sparsely distributed. Thus, for a given triple of input images, we built only those nodes that represent candidate matches, and we only defined connections (binocular or trinocular) between these candidate matches. The nodes and connections were built based on the results of edge detection. Edges were located using different sizes of the Marr-Hildreth edge operator.[12] The initial activation for each match was set at a base level plus any additional support from detailed match. The simulation was allowed to run for either a given number of iterations (usually a max of 16) or until only a small percentage (usually 1%) of the matches had outputs in the intermediate range 0.25 to 0.75.

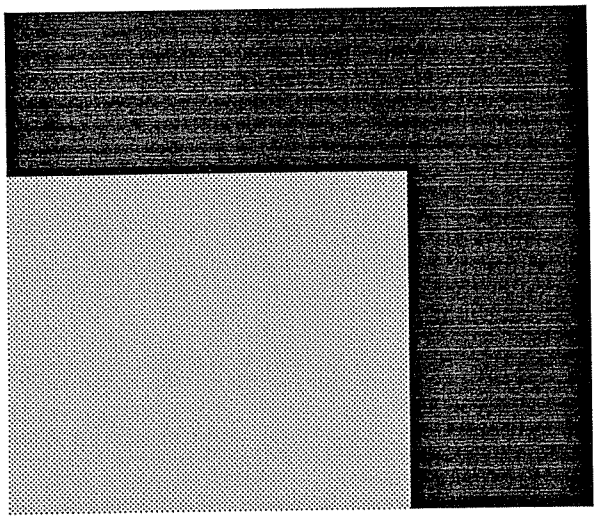

Figure 12. Ground truth disparity for the base image.

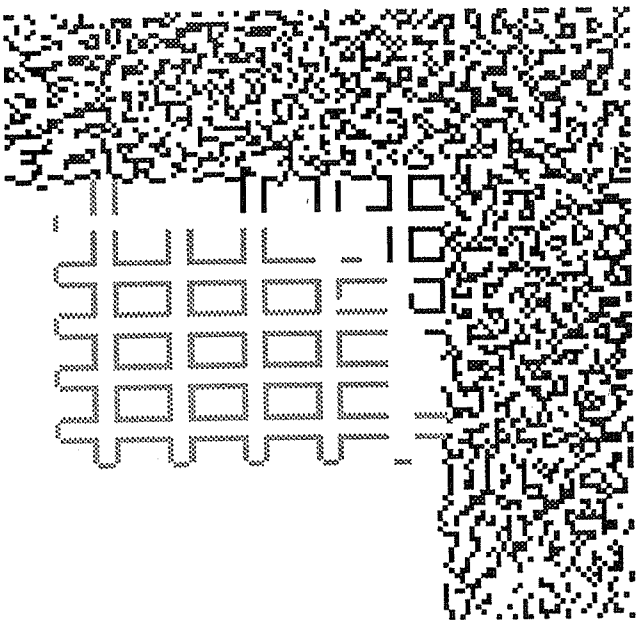

Figure 13. The base disparity image results for trinocular matching.

5.1. Occluded Periodic Regions

Results on two different stereograms containing occluded, periodic regions are presented in this section. The periodic regions in the images contain multiple vertical and horizontal bars. In the first example, the bars in the base image are occluded by a surface that is above and to the right of the region. In the second example, the occluding surface completely surrounds the bars. The occluding regions are textured with a random-dot pattern. Because the algorithm makes nearly 100% correct decisions within the random-dot pattern, the matching results are not presented numerically, only pictorially.

The base, top and right images for the first example are shown in Figure 11. Figure 12 gives the ground truth disparity for the base image encoded as intensity. (The white holes at the boundaries occur because those points do not appear in either the right image or the top image. This is because they are either shifted off the image by their large disparities, they are occluded, or both.) The occluding region, represented by the random-dot pattern, is at disparity 16, while the bars have disparity 4. Because of the occluding surface, the top horizontal bar does not appear in the top image, and the right vertical bar does not appear in the right image. The horizontal frequency of the bars is 14 pixels; the vertical frequency is 10.

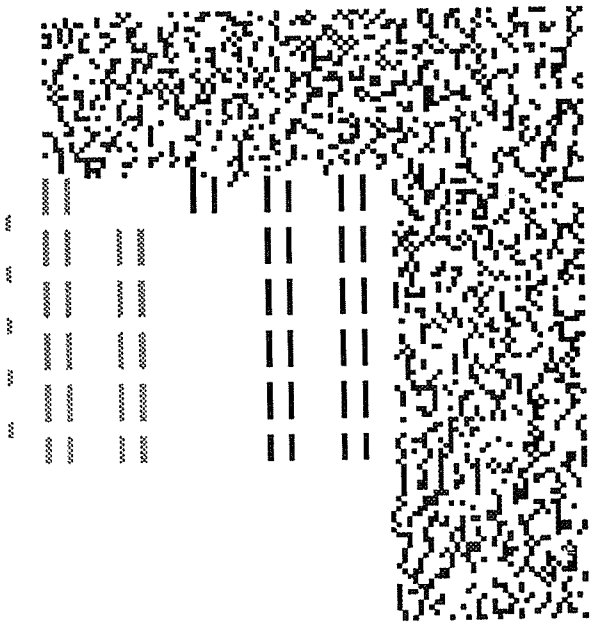

Figure 14. The base disparity image results for binocular matching. Note that because of the large disparity of the occluding surface the results on the left side of the random-dot texture did not extend as far as in the trinocular case.

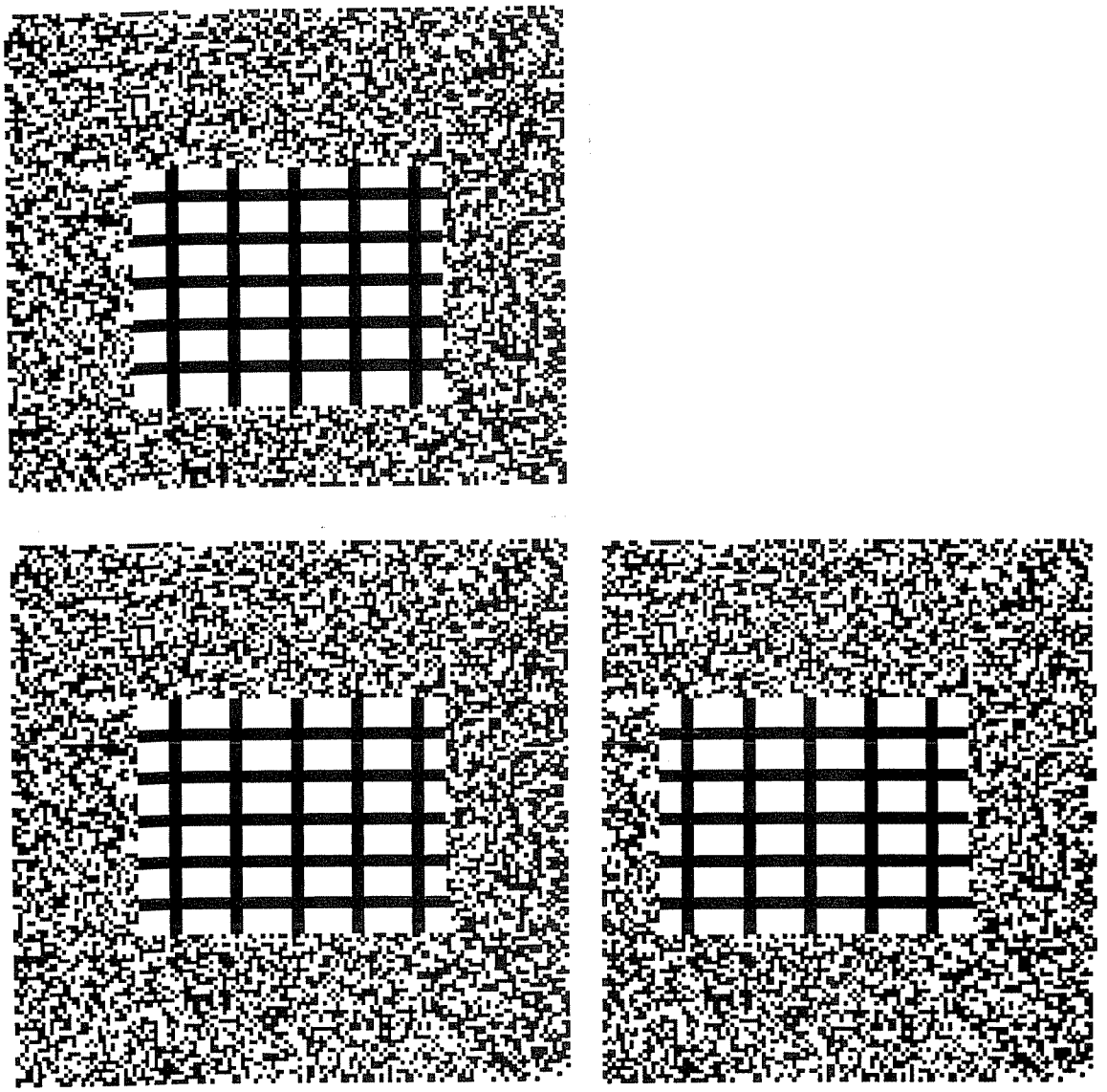

Figure 15. Top, base and right images for the second trinocular example. The random-dot texture is at disparity 16 and the bars are at disparity 4. Of the five vertical bars in the base image, the right one does not appear in the right image. Similarly, of the five horizontal bars in the base image, the top one does not appear in the top image.

The results of trinocular matching are shown in Figure 13 for the base image; the binocular matching results are shown in Figure 14. In binocular matching, the interior of the region is ambiguous, so the only way the algorithm could disambiguate the matches was to propagate the results from the horizontal boundaries of the region. The correct matches were found on the left boundary; the incorrect matches were found on the right. In addition, disparity gradient support from the occluding regions (disparity 16) favored the incorrect matches (disparity 18) near the top of the periodic region.

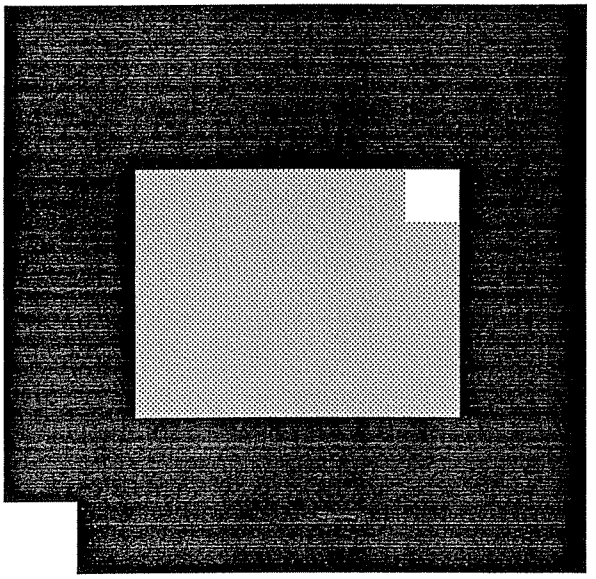

Figure 16. The correct disparities in the base image for the second trinocular example. The blank areas represent points that appear in neither the right nor the top images.

The only errors in trinocular matching for the first example occurred near the upper-right corner of the periodic region. In this area the incorrect matches (at disparity 18 vertically, and 14 horizontally) received strong disparity gradient support from the textured surface (disparity 16). In addition, the incorrect matches at the right and top borders received less uniqueness inhibition. The combination of these two factors caused the errors. In the remaining part of the periodic region, the strength of the trinocular disparity gradient support for the correct matches resolved the ambiguity and allowed the correct matches to be accepted.

The images for the second periodic-image example are shown in Figure 15. The ground truth disparity for the base image is shown in Figure 16. The periodic region is at disparity 4; the occluding region is at disparity 16. Note that five vertical bars and five horizontal bars appear in each image. However, all of these bars should not be matched. The top bar and the right bar in the base image should not be matched; the bottom bar in the top image should not be matched; the left bar in the right image should not be matched. The remaining bars should match to yield the correct disparity of 4.

The results of matching using the trinocular algorithm are shown in Figure 17; the results using the binocular algorithm are shown in Figure 18. As expected, the matches for bars in the binocular case were *all incorrect*. There was no information in the center of the region to disambiguate the matches, so the matching

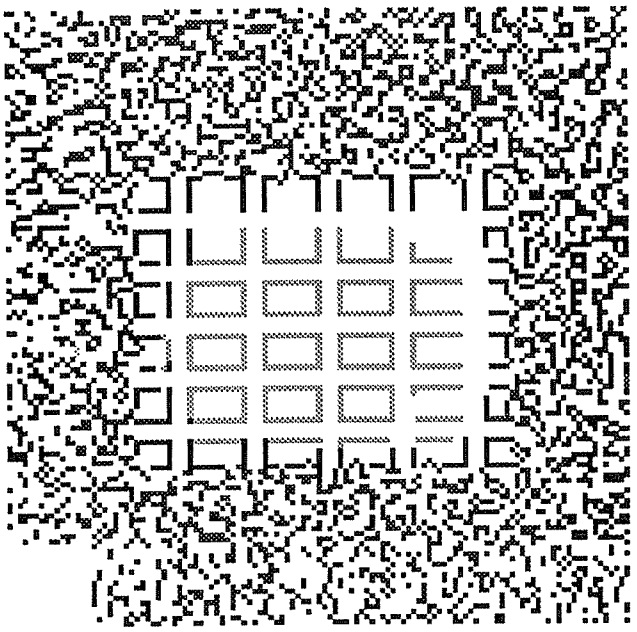

Figure 17. The disparity results of matching using the trinocular algorithm for the base image.

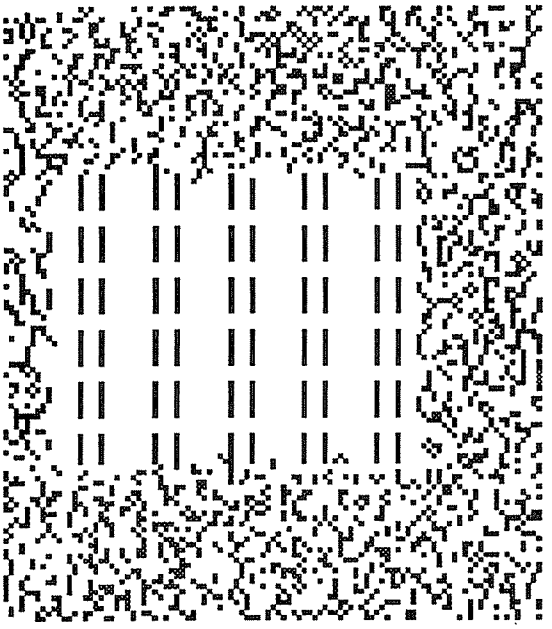

Figure 18. The disparity results of matching using the binocular algorithm for the base image. None of the disparities for the bars in the center of the image were correct.

results at the boundaries of the region were propagated inward. The disparities found at the boundaries of the region were incorrect. Thus, the disparity 18 matches were accepted for each edge in this region. In trinocular matching, the trinocular disparity gradient provided enough support for the correct matches in the center of the

regions to overcome the ambiguity and resist the propagation of erroneous disparities from the boundaries. At the boundaries of the periodic region, matches with incorrect disparities were found because the binocular uniqueness inhibition was weak and there was additional support for these matches through the disparity gradient from the occluding surface.

These results demonstrate the success of the TGSA in handling occluded, periodic regions. The algorithm worked by matching vertically and horizontally using both binocular and trinocular constraints. Trinocular algorithms that only combine results as a post-matching process can not overcome these difficulties.[6,15] In matching a triple of images such as in Figure 15, these algorithms find incorrect matches both vertically and horizontally. When the results are combined, the disparities disagree and none of them are correct.

5.2. Random-Dot Stereograms

In this section we present results for experiments in trinocular matching of random-dot stereograms. These examples are intended to demonstrate the trinocular algorithm's usefulness in handling problems due to noise and occlusion. The test cases included stereograms with 0%, 2% and 5% noise added to each image. For each stereogram there were three surfaces. The disparities of the surfaces were 0, 6, and 12 units.

The disparity images resulting from matching these images for both the trinocular and binocular algorithms are shown in Figures 20 - 25. These show the advantages of the trinocular algorithm in providing a dense and accurate set of match results even in the presence of noise. The numerical results are summarized in Table 1. Figure 19 shows a graph of the percentage of nodes with intermediate activations as a function of the iteration number for the three trinocular results and the binocular algorithm on the images with 2% noise.

	Noise %	% Correct Decisions: Total	% Correct Decisions: No Occlusion	% Correct Decisions: Right Image	Number of Unmatched Edges
Trinoc	0	97.6	98.5	97.9	16
Binoc	0	97.9	98.3	97.4	1341
Trinoc	2	93.1	94.6	94.6	60
Binoc	2	93.2	93.4	93.2	1482
Trinoc	5	87.9	89.3	90.1	94
Binoc	5	85.0	84.6	85.7	1528

Table 1. Matching results for stereograms in Figures 20 - 26.

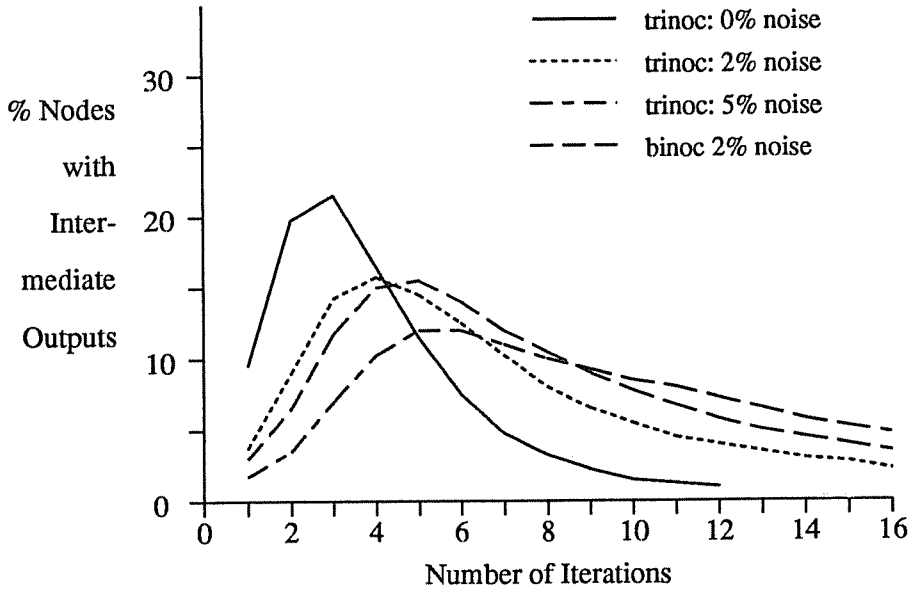

Figure 19. Graph showing the percentage of nodes with intermediate activations as a function of the iteration number. All three trinocular examples are shown, but only the 2% noise binocular example is given.

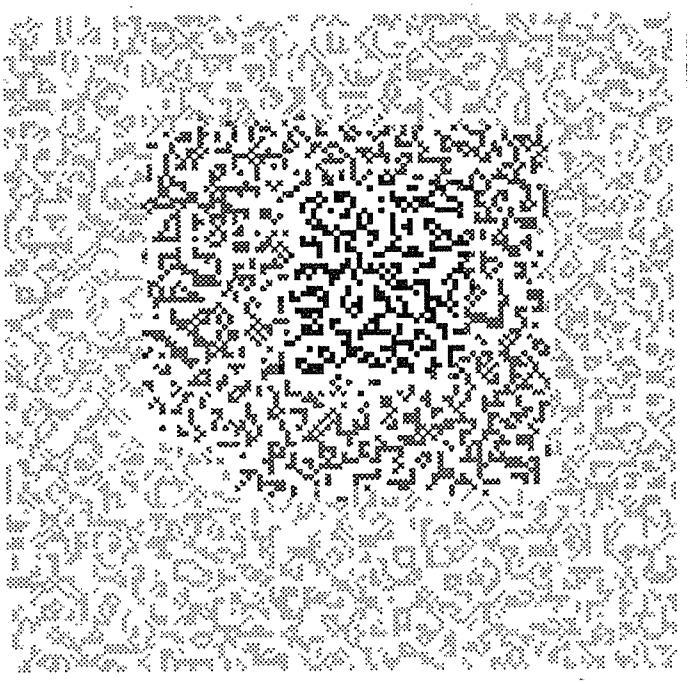

Figure 20. Results of trinocular matching for the base image with 0% noise in each image. When an edge had both a vertical and horizontal disparity the values were averaged. In practice, if the values disagree the value nearer to the average in the region surrounding the edge should be used.

In Table 1 there are four results presented for each run. The first column shows the overall percentage of correct matching decisions. In trinocular matching, when both vertical and horizontal matches were involved, a correct matching decision for an edge was defined as either (1) both a vertical and horizontal correct match were accepted, (2) a correct match was accepted either vertically or horizontally and no incorrect matches were accepted, or (3) no matches were accepted when there were no correct matches available. All other decisions were counted as incorrect. Measuring the performance in this way, the results for binocular and trinocular matching were similar except when 5% noise was added. The second column in the table shows the percentage of correct decisions when no occlusions were involved. Perhaps the clearest indication of the improvements made by using trinocular matching is in the third column of Table 1 which compares the results with respect to the right image. The trinocular algorithm was superior in all examples. The fourth column shows the number of edges which had no candidate matches. This demonstrates that a much denser set of matches is found when using the trinocular matching algorithm.

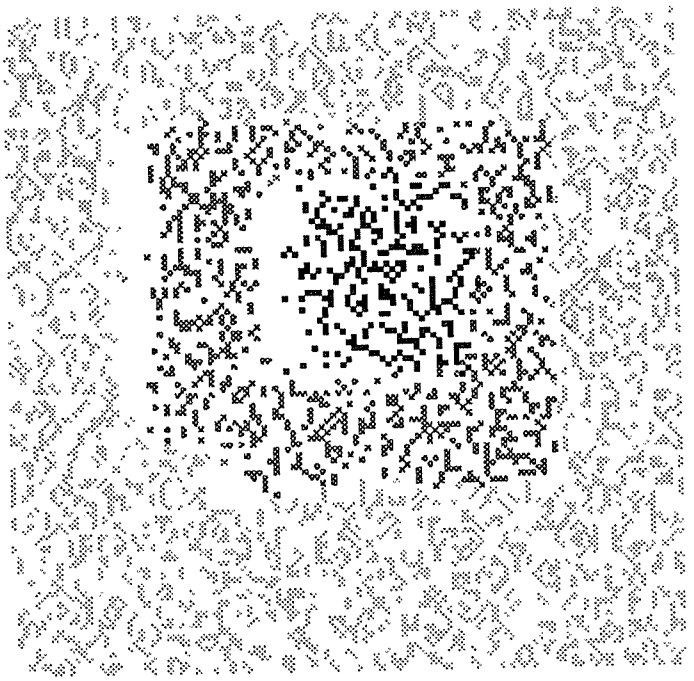

Figure 21. Results of binocular matching for the base image with 0% noise in each image.

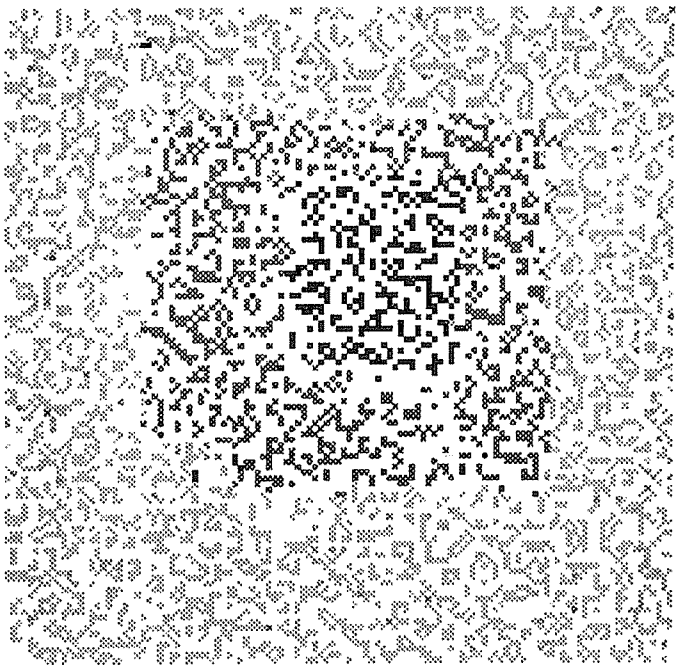

Figure 22. Results of trinocular matching for the base image with 2% noise in each image.

One of the advantages of trinocular matching is that there are fewer regions that appear in the base image, that do not appear in at least one other image. In binocular matching when a region appears in the base image but is occluded in the right, there are no correct matches for the edges and therefore there is no known disparity for the region. In the trinocular case correct matches and disparities may be found in such regions by matching with the third image. The following quantifies the algorithm's performance in half-occluded regions (regions that appear in the base image and *one* of the other two images). The data are taken from the results of matching for the stereogram with 5% noise added to each image. First, consider matches between edges in the region and the image where they *did not* appear. Of the edges with candidate matches, the trinocular algorithm accepted no matches for 81% of them. Fortunately, of the 53 incorrect matches found, 36 had the correct surface disparity. Thus, these incorrect matches did not give incorrect disparity measures. Next, consider matches between edges in the partially occluded region and edges in the image where the region did appear. For edges with candidate matches, the trinocular algorithm produced 89% correct matching decisions. Of the incorrect decisions, 11 were incorrect matches, and 24 were due to missing the correct match. Thus, although statistically the trinocular algorithm's performance in partially occluded regions was not great, the errors did

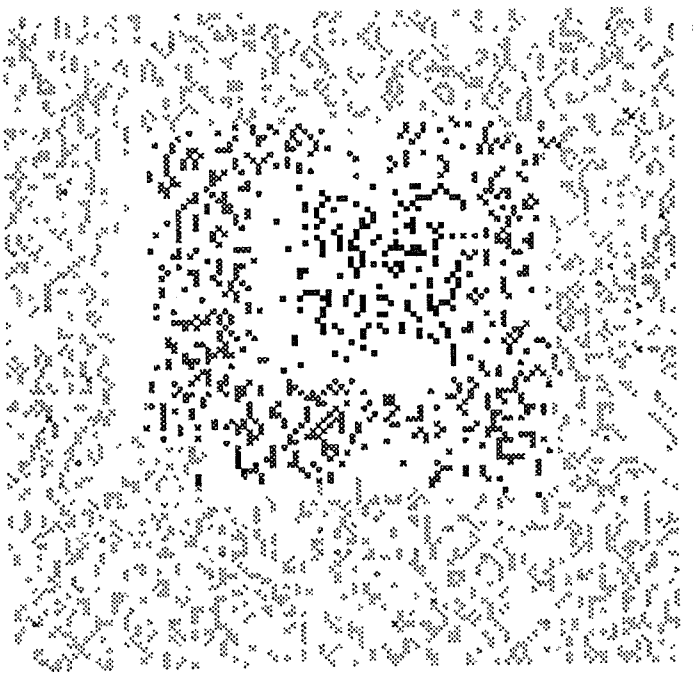

Figure 23. Results of binocular matching for the base image with 2% noise in each image.

not produce incorrect estimates of disparity. These quantitative results are confirmed by examining the disparity image results in Figures 20-25.

The results on the random-dot stereograms demonstrate that the trinocular algorithm accommodated noise in the images significantly better than the binocular algorithm did. In addition, because points could match both vertically and horizontally, a denser set of matches was found. Finally, the algorithm produced correct matches in occluded regions, although a number of erroneous matches were found as well.

As a final consideration, note that the results have been presented using raw disparity values. For the trinocular algorithm, conflicting horizontal and vertical matches for an edge should be resolved by choosing the disparity closest to the average disparity in the surrounding region. This is similar to Ohta's[15] post-matching smoothing algorithm. For both the binocular and trinocular algorithms, some noise matches that could not be eliminated during matching may be removed later through smoothing and surface fitting processes.

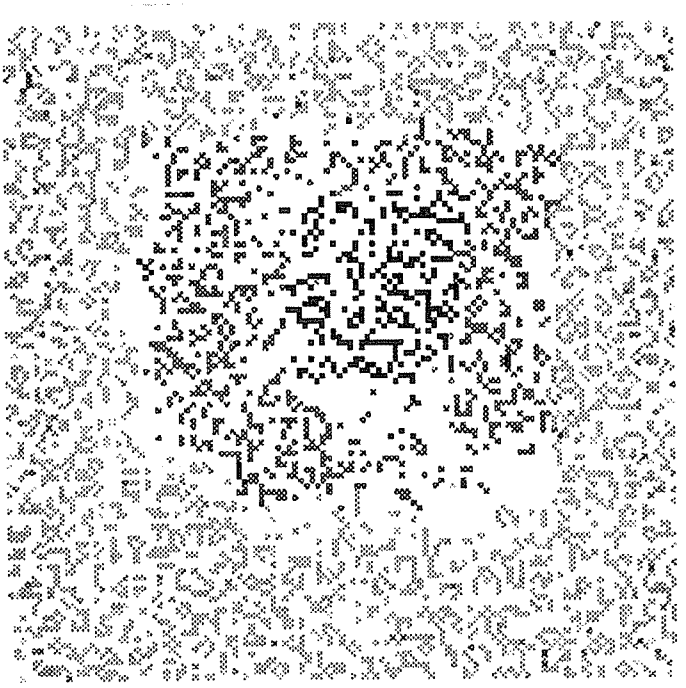

Figure 24. Results of trinocular matching for the base image with 5% noise in each image.

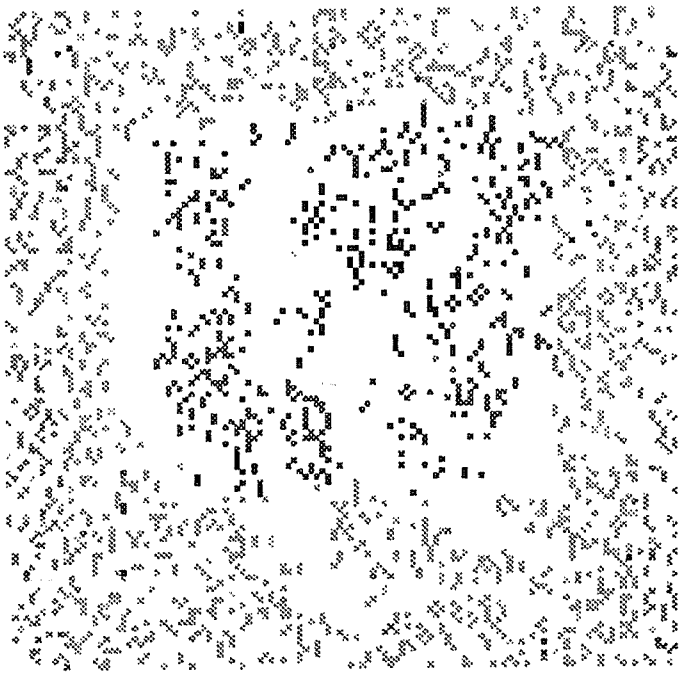

Figure 25. Results of binocular matching for the base image with 5% noise in each image.

6. Summary and Discussion

The Trinocular General Support Algorithm has been defined and shown to be successful in overcoming many of the problems in binocular stereo matching through the use of vertical and horizontal matching with both binocular and new trinocular constraints. New constraints were defined for trinocular uniqueness and the trinocular disparity gradient. The TGSA resolved the binocular problems as follows: (1) The trinocular disparity gradient provided additional information in order to eliminate ambiguity in periodic regions in many cases. (2) Because the algorithm gathered support for a match both vertically and horizontally, the trinocular algorithm performed better in the presence of noise. (3) Regions of the base image that were occluded in the right image were matched vertically solely through information from the base and vertical images. This provided accurate disparities in these half-occluded regions and therefore reduced the problem of noise matches involving occluded edges. (4) The use of both vertical and horizontal binocular matching implied that the algorithm did not rely on an edge being present in all three images in order to match it. This allowed the algorithm to match simple edge primitives and still overcome the difficulty of matching primitives oriented along epipolar scanlines.

The major remaining question is to determine how well the algorithm performs when erroneous variations between images are not stochastic. This issue requires further testing on real image data to determine how important a concern it is and how well our algorithm performs in overcoming it.

References

1. N. Ayache and F. Lustman, "Fast and reliable passive trinocular stereovision," *Proc. First Int. Conf. on Computer Vision*, pp. 422-427 (1987).
2. H. H. Baker and T. O. Binford, "Depth from edge and intensity based stereo," *Proc. Seventh Int. Joint Conf. on Artificial Intelligence*, pp. 631-636 (1981).
3. S. T. Barnard and M. A. Fischler, "Computational Stereo," *Computing Surveys* **14** pp. 553-572 (1982).
4. K.L. Boyer and A.C. Kak, "Symbolic stereo from structural descriptions," *Proc. Conf. on Artificial Intelligence Applications*, pp. 82-87 (1985).
5. J. A. Feldman and D. H. Ballard, "Connectionist models and their properties," *Cognitive Science* **6** pp. 205-254 (1982).
6. A. Gerhard, H. Platzer, J. Steurer, and R. Lenz, "Depth extraction by stereo triples and a fast correspondence estimation algorithm," *Proc. Eighth Int. Conf. on Pattern Recognition*, pp. 512-515 (1986).
7. W.E.L. Grimson, *From Images to Surfaces: A Computational Study of the Human Early Visual System*, MIT Press, Cambridge, Massachusetts (1981).
8. M. Herman, T. Kanade, and S. Kuroe, "Incremental acquisition of a three-dimensional scene model from images," *IEEE Trans. on Pattern Analysis and Machine Intelligence* **6** pp. 331-340 (1984).
9. R. A. Hummel and S. W. Zucker, "On the foundations of relaxation labeling processes," *IEEE Trans. on Pattern Analysis and Machine Intelligence* **5** pp. 267-287 (1983).
10. M. Ito and A. Ishii, "Range and shape measurement using three-view stereo analysis," *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 9-14 (1986).
11. D. Marr and T. Poggio, "A computational theory of human stereo vision," *Proc. Royal Society of London, B.* **204** pp. 301-328 (1979).
12. D. Marr and E. Hildreth, "Theory of edge detection," *Proc. Royal Society of London B.* **207** pp. 187-217 (1980).
13. D. Marr, *Vision*, W.H. Freeman and Company, New York (1982).
14. J. E.W. Mayhew and J. P. Frisby, "Psychophysical and computation studies towards a theory of human stereopsis," *Artificial Intelligence* **17** pp. 349-385 (1981).
15. Y. Ohta, M. Watanabe, and K. Ikeda, "Improving depth map by right-angled trinocular stereo," *Proc. Eighth Int. Conf. on Pattern Recognition*, pp. 519-522 (1986).
16. M. Pietikainen and D. Harwood, "Depth from three camera stereo," *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 2-8 (1986).
17. S. B. Pollard, J. E.W. Mayhew, and J. P. Frisby, "PMF: a stereo correspondence algorithm using a disparity gradient limit," *Perception* **14** pp. 449-470 (1985).
18. K. Prazdny, "Detection of binocular disparities," *Biological Cybernetics* **52** pp. 93-99 (1985).
19. D. E. Rumelhart, G. E. Hinton, and J. L. McClelland, "A general framework for parallel distributed processing," pp. 45-76 in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1: Foundations*, ed. D. E. Rumelhart and J. L. McClelland,The MIT Press, Cambridge, MA (1986).
20. C. V. Stewart and C. R. Dyer, "Local constraint integration in a connectionist model of stereo vision," *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, (1988).
21. C.V. Stewart, "Connectionist models of stereo vision," PhD Disseration, University of Wisconsin, Madison, WI (1988).
22. D. Terzopoulos, "Image analysis using multigrid relaxation," *IEEE Trans. on Pattern Analysis and Machine Intelligence* **8** pp. 129-139 (1986).
23. R. Y. Tsai, "Multiframe image point matching and 3-D surface reconstruction," *IEEE Trans. on Pattern Analysis and Machine Intelligence* **5** pp. 159-173 (1983).
24. A. P. Witkin, "Scale-space filtering," *Proc. Eighth Int. Joint Conf. on Artificial Intelligence*, pp. 1019-1022 (1983).

25. M. Yachida, Y. Kitamura, and M. Kimachi, "Trinocular vision: a new approach for correspondence problem," *Proc. Eighth Int. Conf. on Pattern Recognition*, pp. 1041-1044 (1986).