International Journal of Computer Vision, 6:3, 197-226 (1991)

A Topological Stereo Matcher

MARGARET M. FLECK
Department of Engineering Science, Parks Rd., Oxford OX1 3PJ, United Kingdom

Received November 27, 1989. Revised April 5, 1991.

Abstract

Presented here is a new stereo algorithm that produces dense, high-quality, subpixel disparity maps. It offers two improvements over previous algorithms. First, it does not blur disparity values across sharp changes in depth. Second, it can reconstruct the correct correspondence between two images even when there is substantial vertical displacement between them: this algorithm has been tested with rotations up to 10 degrees and vertical translations up to 16 pixels. Although such image pairs require extra processing time, this ability is vital when exact calibration cannot be maintained.

The new algorithm depends on two new ideas. First, it exploits the fact that the correct vertical disparity field is due to camera misalignment and, thus, has only a few (significant) degrees of freedom. The algorithm passes camera alignment parameters, not raw disparity fields, between scales. Disparities at individual locations can diverge only slightly from this global model, greatly reducing the algorithm's search space.

Second, the new algorithm uses a pre-match filter that prevents two patches of image from matching if they do not have the same (local) topological structure. This constraint subsumes previous "figural continuity" proposals and can be checked by simple, local operations. The filter seems to improve the algorithm's ability to select the correct match from many alternatives and it suppresses intermediate values near sharp changes in disparity. This technique can be extended to other matching tasks, such as motion tracking, analyzing texture periodicity, and evaluating the performance of edge finders.

1 Introduction

Many recent stereo algorithms can produce dense, high-quality disparity maps similar to those in figures 1 and 2.[1] The algorithm that produced these results, however, offers two advantages. First, as table 1 illustrates, the new algorithm reconstructs sharp changes in disparity as sharp. Most previous algorithms generate blurred disparity values across such boundaries, widely recognized to be undesirable. Second, the new algorithm can match image pairs even if they are not in perfect vertical alignment.[2] Thus, it can fuse the pair in figure 2, despite a 4° rotation that would defeat most previous algorithms. In synthetic test examples, it has handled 16-pixel vertical translations and 10° rotations. Recovering from such gross misalignments clearly requires more time than matching perfectly aligned images, but this ability is necessary for camera systems subject to bumps and vibrations and for models of human perception.

Vertical disparities pose two problems for a matching algorithm. First, their presence greatly enlarges the space of possible disparities that must be considered: the correct disparity field for an image with 50 horizontal disparities and a 4-degree rotation could contain over 1500 distinct (d_x, d_y) pairs.[3] As all algorithms explore some plausible but wrong matches, a naive search rapidly becomes prohibitive. Second, as more and more possible matches are considered for each image location, increasing pressure is placed on the algorithms that evaluate matches and choose the best match for each location. Because previous stereo algorithms do not explore large search spaces, it is not known whether they can tolerate large numbers of competing matches without becoming confused.

The stereo algorithm presented in this article uses two new ideas. First, it takes advantage of the fact that a nonzero vertical disparity field arises from a misalignment of the two cameras, implying that the correct field

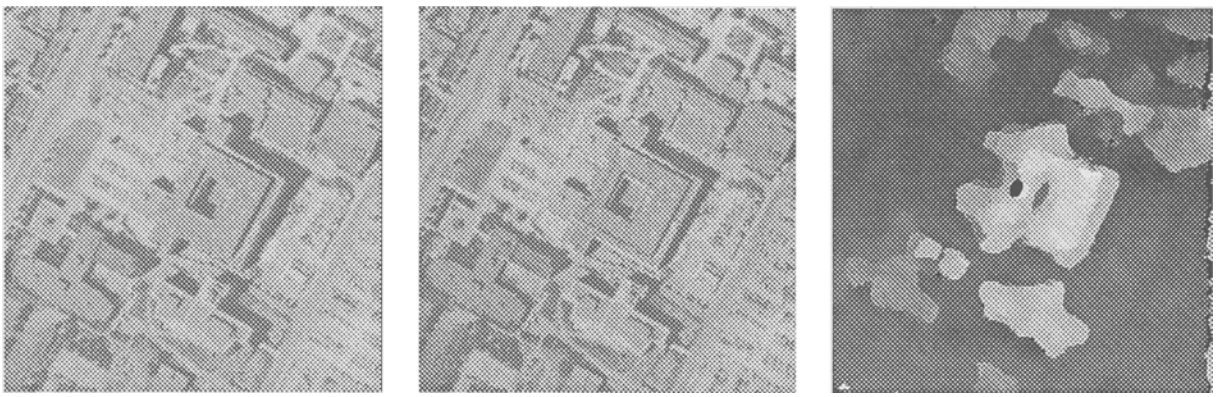

Fig. 1. A stereo pair of 320 by 320 images and computed disparity field. Nearer regions are shown in lighter shades; occluded regions as black. The computed range of nearest-cell disparities was [−8, 5] cells, the computed rotation was 0.17°, and the computed vertical translation was 0.0 cells.

has only a few degrees of freedom. Like many previous algorithms, the new matcher starts by matching blurred versions of the two images and proceeds gradually to finer-resolution versions. As it moves from one scale to the next, however, the new algorithm does not carry with it the whole complex vertical disparity field, but only the parameters required to realign the camera positions. The vertical disparity at each individual location is prevented from diverging more than slightly from the model of the whole field. This greatly reduces the number of incorrect disparities considered.

The second innovation is a prematch filter that eliminates patches of image from consideration at a given disparity if they do not have the same qualitative structure. Like many algorithms, the new matcher requires that boundaries match boundaries and that boundary polarity must be the same, that is, the lighter side of a boundary in one image must correspond to the lighter side of a boundary in the other image. In addition, however, I require that corresponding regions have the same local topological structure. This is defined precisely in section 3, but figure 3 illustrates the general idea. In particular, connected boundaries must be matched to connected boundaries, in line with previous *figural continuity* proposals (see figure 4). The topological filter, however, ignores small changes in boundary positions, such as those caused by noise or small differences in viewpoint.

Suppose that the matcher is comparing the two images in figure 1 at disparity (−5, 0), roughly appropriate for the background to the left of the central building. An edge finder is applied to both images and its outputs are compared. Figure 5 shows the regions of the image in which approximately corresponding positions (for disparity (−5, 0)) contain boundaries with the same polarity and the same topological structure. The area that is actually at this disparity is accepted as plausible, whereas the buildings, which are at a very different disparity, are largely rejected. As in previous algorithms, details of the two edge-finder outputs are then used to compute evaluations of the match near each location. However, this detailed evaluation is confined to connected areas accepted by the topological filter.

Adding the topological filter has two important consequences. First, it allows the matcher to distinguish small errors in boundary positions, however numerous, from real qualitative differences in image structure. This seems to improve the reliability of match evaluations to the point that they can choose correctly from among large numbers of competing matches. Second, suppose that there is a sharp change between disparities d_a and d_b. At intermediate disparities, very little of the image will satisfy the topological filter, so these disparities will be given low evaluations. Even at the edge, they will be rejected in favor of one of the two correct disparities. Thus, sharp changes will be reconstructed as sharp (though the 2D shape of the boundary may be rounded).

This article is divided into four sections. The first describes the overall structure of the stereo matcher, finessing details of the search strategy and the topological filter. The second defines what I mean by "topological structure" and details the topological filter. The third section describes how the stereo matcher searches the space of possible displacements. Finally, the fourth section shows how to interpolate matcher results to subpixel accuracy and suggests how the matcher might be adapted to other problems such as motion tracking,

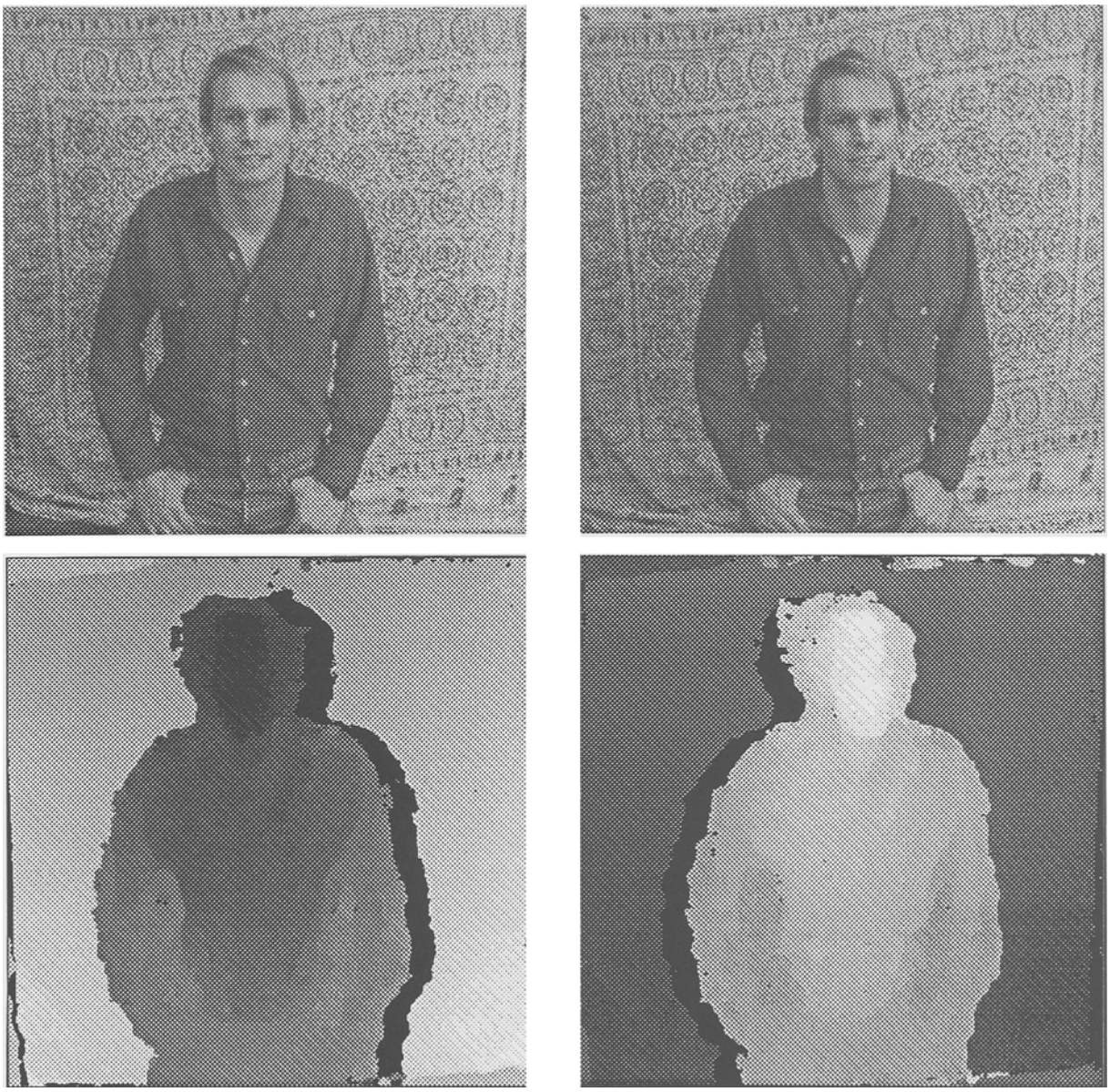

Fig. 2. A stereo pair of 450 by 450 images and both halves of the disparity map. Occluded areas of background (black) are correctly identified next to the researcher. On the sides without occlusions, the boundary between the researcher and the background is correctly reconstructed as a sharp change in disparity values. The computed vertical translation was 0.5 cells.

analyzing the periodicity of textures, and evaluating the performance of edge finders.

2 Overview of the Stereo Matcher

The overall structure of the new matching algorithm is similar to many previous algorithms. To start, each image is repeatedly smoothed and subsampled by a factor of two in each dimension, to produce a pyramid of increasingly coarse versions of the image. The matching algorithm proceeds from coarser to finer scales, using edge-finder output at the current scale to refine the disparities computed at the previous scale. Figure 6 summarizes processing at each scale. This section will describe the more standard and straightforward parts of the new algorithm.

Table 1. Disparity values (in cells) for a patch on the lower right-hand corner of the large (middle) building in figure 1.

−1.3	−1.2	−1.2	−1.2	−1.2	−1.2	−1.2	−1.2	−1.2	−1.2	−1.1
−1.3	−1.2	−1.2	−1.2	−1.2	−1.2	−1.1	−1.2	−1.2	−1.2	−1.2
−1.2	−1.2	−1.2	−1.2	−1.2	−1.2	−1.2	−1.2	−1.2	−1.2	3.8
−1.2	−1.2	−1.2	−1.2	−1.2	−1.2	−1.2	−1.2	−1.2	4.0	3.9
−1.2	−1.1	−1.2	−1.2	−1.2	−1.2	−1.3	−1.2	4.2	4.1	4.0
−1.2	−1.2	−1.2	−1.3	−1.2	−1.2	−1.2	4.2	4.1	4.1	4.1
−1.1	−1.2	−1.3	−1.2	−1.1	−1.2	−1.2	4.2	4.1	4.1	4.1
−1.2	−1.2	−1.2	−1.2	−1.2	−1.2	3.8	4.1	4.1	4.1	4.1
−0.9	−1.0	−1.0	−1.0	−0.9	3.8	4.3	4.2	4.1	4.0	4.1
−0.9	−0.9	−1.0	2.0	4.3	4.3	4.3	4.2	4.1	4.0	4.0
−0.9	−0.9	4.3	4.3	4.3	4.2	4.2	4.2	4.1	4.0	4.0

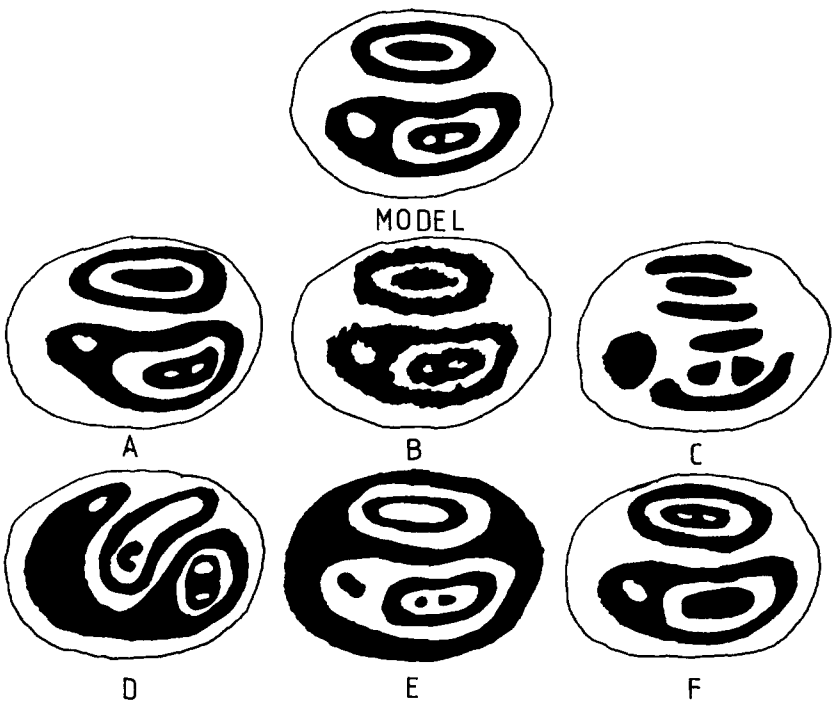

Fig. 3. A and B have the same qualitative structure as the model, despite small distortions and noise. The other four patches do not. C has entirely different topological structure. D involves such a large distortion that the implemented matcher cannot find the topology-preserving correspondence. E has the wrong boundary polarity. The individual regions in F match those in the model, but they are embedded differently.

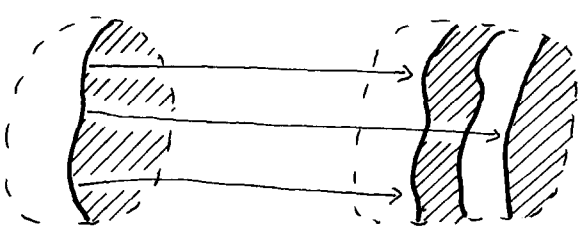

Fig. 4. The figural continuity constraint forbids (or discourages) matches like this, in which a connected boundary in one image matches to parts of two boundaries in the other.

The new matcher assumes that the correct, continuous disparity field satisfies five constraints:

- The correspondence is bijective.
- At each scale, disparities are nearly constant over small patches of image.
- The match preserves the topological structure of the images.
- Disparities suggested by coarse-scale cues are approximately correct.
- The images are in approximately correct vertical alignment.

The first two constraints are standard and will be reviewed in this section. The topological constraint is new and will be discussed in section 3. The final two

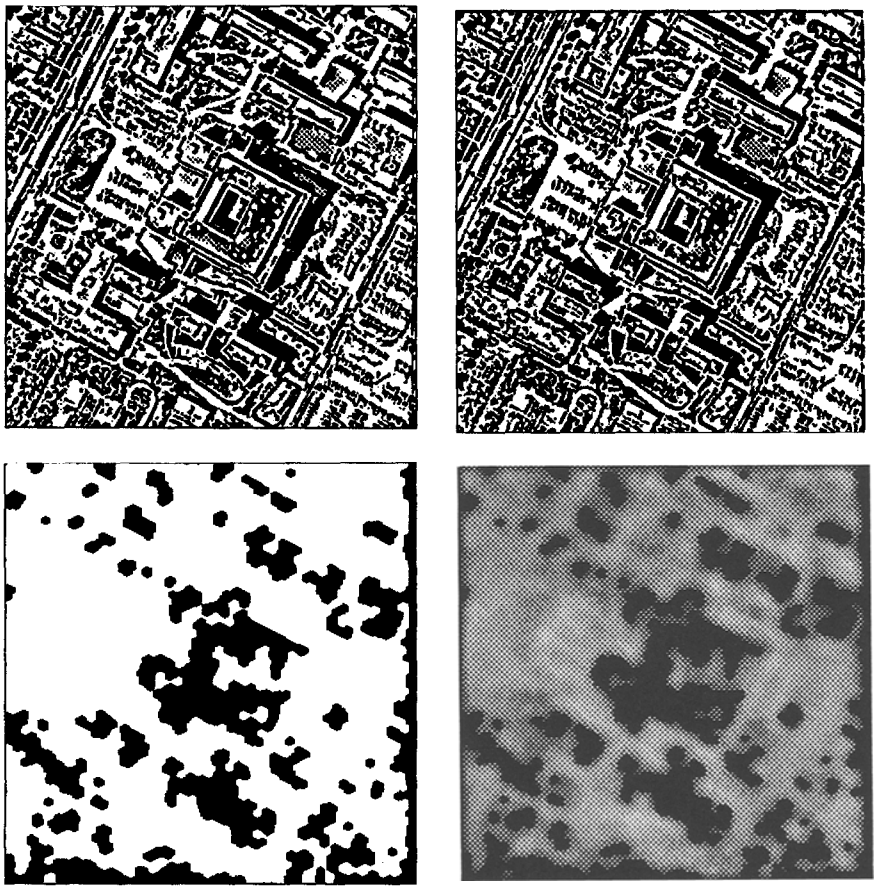

Fig. 5. To compare the images from figure 1, the edge finder is first run, yielding a dark/light labeling of each image (top row). At disparity (−5, 0), the topological filter classifes the area left of the central building as *possible* match (bottom left, white) but the buildings as largely *impossible* (black). Detailed evaluations are then computed within *possible* regions (bottom right).

constraints limit the space of disparities considered in matching and will be discussed in section 4.

2.1 Local Constancy and the Main Search Loop

The new matcher assumes that disparities vary sufficiently slowly that they can be approximated as translations within the neighborhoods used in matching. Therefore, at each scale, it compares the two images at a set of translations determined by the search routine described in section 4. Each translation is represented by a (d_x, d_y) disparity vector. Evaluations are computed for each disparity, about each pixel. Finally, each pixel is assigned the disparity that had the highest evaluation.

To be more precise, the new algorithm maintains two disparity maps, one viewing the scene from the point of view of the right eye and one from the point of view of the left eye (figure 2). Suppose that it has compared the left-hand image to a version of the right-hand one translated by $(-d_x, d_y)$. The matching strength at (x, y) is then used to update location (x, y) in the left-hand disparity map, but location $(x + d_x, y + d_y)$ in the right-hand map. Thus, the two output-disparity maps may differ. It is easy to produce a single "cyclopean" map [Barnard 1989], in which location $(x + d_x/2, y + d_y/2)$ is updated, but this makes occlusion detection (section 2.2) difficult.

The new matcher can give a good match evaluation to a cell only if it belongs to a large enough neighborhood in which the disparities are nearly constant. Most other stereo algorithms use either a similar local constancy constraint [Drumheller & Poggio 1986; Grimson 1981a, b, 1985; Marr & Poggio 1976], a "smoothness" or "continuity" constraint[4] [Hoff & Ahuja 1989], local applications of a *disparity gradient* bound[5] [Ayache & Faverjon 1987; Pollard, Mayhew, & Frisby 1985], or similar constraints [Marr, Palm, & Poggio 1978; Marr

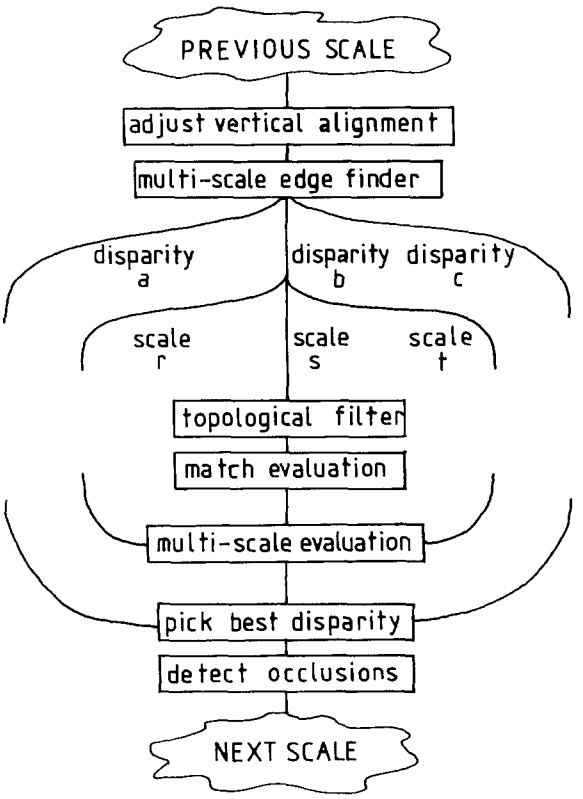

Fig. 6. An overall picture of what the new matcher does at each scale.

& Poggio 1976; Medioni & Nevatia 1985; Prazdny 1985]. Given multiscale processing and due allowance for errors in boundary locations, all these constraints have very similar effects and would be difficult to tell apart in practice.

The local constancy constraint in the new matcher will cause performance to degrade on surfaces that slope steeply away from the observer. To assess how quickly this occurs, I built two synthetic stereograms (figure 7) containing ramps with disparity gradients (vertically) of 0.39 and 0.89.[6] Matching succeeds on the first example, but is starting to fall apart on the second. This is slightly worse than human performance, which falls apart at disparity gradient 1 [Burt & Julesz 1980a, b].

2.2 *Uniqueness and Occlusion Detection*

The above algorithm works well when disparities vary smoothly. However, at sharp changes in disparity, a patch of 3D surface may be occluded, that is, visible from only one of the two images. Cells corresponding to such a surface patch are often assigned incorrect matches. Evaluations for these matches may be higher than those of some real matches, so pruning matches with low evaluations is not very reliable. The new algorithm prunes spurious matches[7] using a cautious implementation of a *uniqueness constraint*, that is, a constraint that the underlying, continuous match is bijective. The new algorithm is similar to that proposed by Little and Gillet [1990] but it is simpler and has a tighter physical justification.

Specifically, suppose that location (x, y) in the left-hand image has been assigned disparity (d_x, d_y). The pruning algorithm compares the matching M_L strength at (x, y) to the strength M_R at location $(x + d_x, y + d_y)$ in the right-hand image. Careful reading of sections 2.4–2.5 and section 3 will reveal that the matching process is symmetric in the two images, so the two strengths would have to be identical if the disparity at right-hand location $(x + d_x, y + d_y)$ was exactly $(-d_x, -d_y)$. Because the images are noisy and digitized, the actual disparity at $(x + d_x, y + d_y)$ could be slightly different and M_R slightly higher than M_L.[8] However, on a correctly matched, smoothly varying surface, evaluations change only slowly, so the two evaluations should be similar. If M_L is less than 90% of M_R, the algorithm decides that the matches are incompatible and prunes the disparity at (x, y) (figure 8).

The pruning process occasionally leaves small holes in the disparity maps. Therefore, after pruning, the corresponding location for each match is reexamined. If this location, or any of its eight neighbors, is marked as nonmatching, the match from the first image is copied over. As figures 2, 11, 24, and 26 illustrate, this pruning and filling process seems to remove most matches in occluded regions without damaging other parts of the disparity map.

The uniqueness constraint is one example of a *global consistency* constraint, that is, a constraint restricting disparity values at cells that may not be near one another in either image. Uniqueness is only occasionally implemented (e.g., Ayache and Faverjon [1987]), although it imposes interesting constraints in three-camera systems [Stewart & Dyer 1988]. Most algorithms use the weaker constraint that an image location can contain only one disparity. One stereo matcher [Baker & Binford 1981] and one technique for pruning matches in occluded regions [Little & Gillett 1990] have used the stronger *ordering constraint*, which requires that the left-to-right order of points be preserved. The yet-stronger *disparity gradient bounds* (section 2.1) were originally proposed in the psychophysical literature

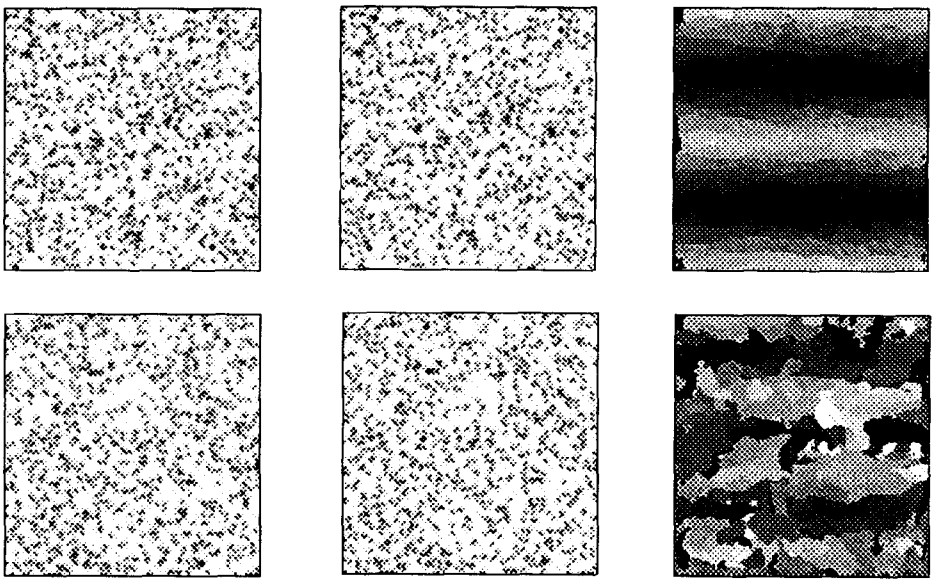

Fig. 7. Two 200 by 200, 10% random-dot stereo pairs (dots 2 cells on a side) depicting triangular ramp patterns. The upper pair, with disparity gradient 0.39, is fused successfully. A roughly correct qualitative pattern is computed for the lower pair (disparity gradient 0.89) but many errors occur.

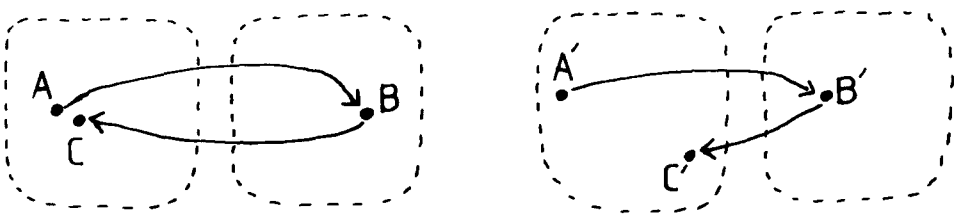

Fig. 8. The left-hand matches are consistent with uniqueness. $A \rightarrow B$ and $B \rightarrow C$ will usually have similar evaluations. The right-hand matches violate uniqueness. The evaluation for $A' \rightarrow B'$ can be no higher than that of $b' \rightarrow C'$. If it is substantially lower, $A' \rightarrow B'$ will be pruned.

[Burt & Julesz 1980a,b] as a global constraint on image disparities and are occasionally [Drumheller & Poggio 1986; Gillett 1988] implemented as such.

When two disparities violate a global constraint, the matcher must decide which one to prune. The obvious tactic, used in the new matcher and in [Drumheller & Poggio 1986], is to prune the match with the lower strength. However, it is essential to include some tolerance for noise in the computed disparities and strengths, such as the 90% threshold in the new matcher. Where no noise tolerance is allowed, as in Drumheller & Poggio's [1986] algorithm, the correspondence may be disrupted in regions of smoothly changing disparity (see the figures in Gillett [1988]) and unstable when the two conflicting matches have similar strength.

Uniqueness is generally accepted as a true constraint on stereo matching, because a patch of surface cannot be at two depths simultaneously. The ordering and disparity gradient constraints, on the other hand, make assumptions about scene geometry that can easily be violated. The psychophysical evidence is somewhat murky. A global disparity gradient constraint seems to be implied by the data in Burt and Julesz [1980a, b], but it would also prohibit reconstruction of sharp changes in disparity, which humans clearly perceive. The thin nail illusion [Krol & van de Grind 1980] could reflect an ordering constraint, but could also be due to coarse scales biasing fine scales in a multiscale algorithm.

Some authors, for example [Grimson 1981a], seem to believe that "transparent" stereograms such as the one in figure 9 violate uniqueness because (a) one dot may match two dots and/or (b) there is one surface in front of another. However, each dot generates a several-cell patch of edge-finder response, which can easily be divided among two disparities. The new matcher gener-

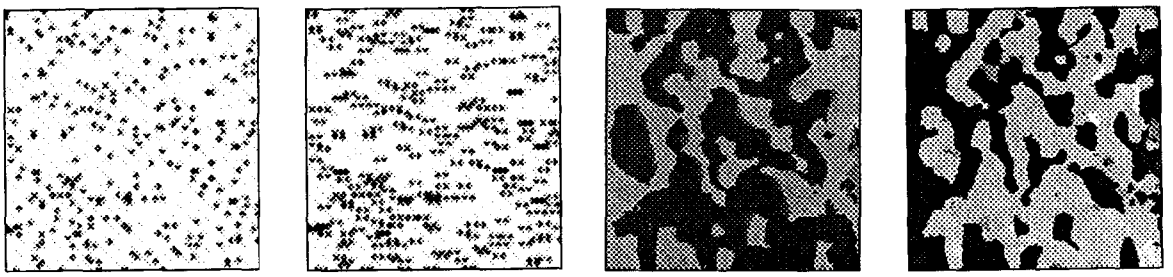

Fig. 9. A 200 by 200, 5% random-dot stereogram (dots 3 cells on a side) depicting Panum's limiting case. Each cell in the left-hand image corresponds to two cells in the right-hand image, at disparities 0 and 6 cells. The uniqueness constraint in the new matcher does not prevent both the left and right disparity maps from detecting matches on both planes.

ates disparities at both correct depths, in both the left-hand and right-hand disparity maps, so long as the patches at each depth are large enough to generate good evaluations. Although two connected surfaces could be interpolated from these patches, I see no evidence that this could not be left to later processing.

2.3 The Edge Finder

The new matching algorithm compares patches of image by comparing the results of applying an edge finder to them. The particular edge finder I use is described by Fleck [1989] and is similar to those in [Fleck 1990b]. From each image, it generates two maps. One (figure 5) labels pixels as *dark* or *light* depending on the sign of a second-difference operator. Pixels right on a boundary or far from any boundary are labeled *zero*. The second map labels image cells as *passing* if their first and third differences have opposite signs [Fleck 1989, 1990b; cf. Clark 1989] and as *failing* otherwise. Sign-test labels, shown in figure 10, are not as stable as the second-difference labels, but they make it possible to eliminate spurious boundaries in staircase intensity patterns and provide useful clues about shading, edge blur, and locations of "roof edges."[9] Boundaries are then marked where,

- the sign test is *passing*, and
- the second-difference labels are *zero* or change from *dark* to *light*.

Requiring a match between edge-finder outputs is a common method of matching images, both in stereo and in other matching tasks [Baker & Binford 1981; Buxton & Buxton 1984; Drumheller & Poggio 1986; Grimson 1981a,b, 1985; Hildreth 1984; Hoff & Ahuja 1989; Little, Bulthoff, & Poggio 1987; Marr & Poggio 1979; Mayhew & Frisby 1980, 1981; Nishihara 1984; Ohta & Kanade 1985; Pollard, Mayhew, & Frisby 1985; Prazdny 1985; Vilnrotter, Nevatia, & Price 1986]. Some matching algorithms use primarily information at boundaries but match higher-level features such as extended straight segment [Ayache & Faverjon 1987; Medioni & Nevatia 1985] or corners [Barnard & Thompson 1980; Gennery 1977; Hannah 1980; Lawton 1983; Moravec

Fig. 10. The edge finder's *sign-test* map (left; light indicates *passing*) and its output boundaries (right), for one of the images in figure 1.

1977, 1981; Nevatia 1976; Spacek 1986] or region descriptions [Boyer & Kak 1988].

Other algorithms compare intensity values rather than using boundaries, either directly [Barnard 1989; Gennert 1987, 1988; Gillett 1988; Levine, O'Handley, & Yagi 1973; Mori, Kidode, & Asada 1973; Quam 1984; Scott 1988; Shahraray & Brown 1988; Witkin, Terzopoulos, & Kass 1987] or after taking finite differences [Kass 1987] or after Fourier transformation [Bajcsy 1973; Matsuyama, Miura, & Nagao 1983; Yeshurun & Schwartz 1989] or after extracting phase information [Fleet & Jepson 1989; Jepson & Jenkin 1989; Thomas 1987]. A few match the image or set of images to models of grey-scale events [Bolles, Baker, & Marimont 1987; Bovik, Clark, & Geisler 1987; Heeger 1987; Kass & Witkin 1985, 1987; Zucker 1985]. See also the survey in [Barnard & Fischler 1982].

2.4 Matching Using the Topological Filter

At each disparity, the algorithm for matching two images is given the two edge-finder outputs, together with the result of the topological filter described in section 3. From these, it computes a matching strength at each cell, based on the size of the match region containing it and the amount of difference between the two edge-finder outputs in this region. Let us assume, first, that edge finder and topological data are only available at a single scale and let us call the two labels in the topological filter output *possible* and *impossible*.

Specifically, to compute matching strength, each *possible* cell is assigned a weight. This weight is 2 if labels match in both the second-difference map and the sign-test map. The weight is 1 if the labels match in only one of the two maps, and 0 if neither matches. These labels were used because they are easy to obtain, reliable, and provide information in smoothly shaded regions as well as at boundaries. Other features with similar properties (e.g., edge contrast, first derivative of intensity) could also be used to generate weights.

The matching strength at each cell (e.g., as in figure 5) is the sum of all cell weights over a support neighborhood of that cell. The details of support neighborhood shapes are given in the appendix. The important features of these neighborhoods are that they are connected and they contain only *possible* cells. Thus, matching strength reflects the size of a connected match region, as well as how much boundaries and other features have moved. *Impossible* cells are assigned strength 0.

When the new algorithm matches images at an incorrect disparity, the topological filter typically labels few cells as *possible*. Thus, only low evaluations are generated and the disparity tends to be rejected in favor of better options. In particular, where there is a sharp change in disparity, intermediate disparities are usually rejected in favor of the correct disparities for the two extended regions. Thus, sharp changes are reconstructed as sharp, though (of course), their 2D shape and location may be blurred.

Comparing the edge finder's two label maps uses (perhaps indirectly) most of the information used by previous edge-based stereo matchers. Contrast magnitude was not incorporated into the new algorithm at all, though it could easily be added and some algorithms [Pollard, Mayhew, & Frisby 1985] use it. Certain previous stereo algorithms (e.g., Grimson [1981a, b, 1985] and Hoff and Ahuja [1989]) exclude near-horizontal boundaries from matching, because they provide little information about horizontal disparities. However, these boundaries are critical to identifying any vertical displacement.

Like the new algorithm, most previous algorithms average local estimates of match error, to make them more reliable. However, this averaging is often done in the final disparity map, rather than at individual disparities, and tends to blur values across sharp changes in disparity. Interpolation or other post-processing often adds to the blurring. A few stereo and motion algorithms [Ayache & Faverjon 1987; Grimson 1985; Grimson & Pavlidis 1985] try to adapt averaging neighborhoods to local conditions, with some improvement of results. Others [Drumheller & Poggio 1986; Pollard, Mayhew, & Frisby 1985; Schunck 1989] restrict averaging to cells with similar preferred correspondence. In these, however, a good region could artificially inflate the ratings of bad matches near it, but not connected to it.

2.5 Multiscale Match Evaluation

If fine-scale matching of an image uses only the fine-scale match strengths, the choice of the best disparity depends only on information in a very small neighborhood of each location. The new stereo matcher avoids such myopia by combining the fine-scale information with information from coarser scales, which consider a wider context. Versions of this idea have been proposed by Kass [1987] and Mayhew and Frisby [1981].

Specifically, let us call the finest scale 0, the next coarser scale 1, and so forth up to some coarsest scale N. At each displacement (x, y), edge-finder outputs at scale i are matched as in section 2.4 to yield a strength map $S_i(x, y)$.[10] I define the multiscale strength to be

$$M_n(x, y) = \sum_{i=n}^{N} 0.8^{i-n} S_i(x, y)$$

where coarser-scale maps are interpolated to the size of S_n before averaging. This is simply an average of scale n and all coarser scales, with weightings determined by experimentation. For later use, these strengths are rescaled into the range [0, 240] and the values represented to the nearest 20th. Because of the sampling, computing M_n takes only 4/3 the time required to compute S_n.

Although fine-scale information tends to take precedence, coarse-scale information allows the new matcher to interpolate disparities for regions far from boundaries. Figures 11 and 12 illustrate this for stereograms with sparse features. Figure 13 shows a more interesting example, a view of an ellipsoid similar to those presented by Bülthoff and Mallot [1988]. Since all the boundaries are at or near zero disparity, the raised center must be reconstructed from other information in the edge-finder labels: the sign-test values and transitions from zero to nonzero labels.[11] Figure 14 shows disparity profiles for ellipsoids at a range of elongations.

3 Topological matching

The goal of the topological filter is to eliminate areas of image that are impossible matches, at a given approximate disparity. Thus, it requires that the two images contain similar arrangements of regions, but it ignores small differences in shape (figure 3). Without the topological filter, the raw comparison of edge-finder labels, described in section 2.4, is very sensitive to differences in shape, but cannot determine whether they indicate significant qualitative differences. Thus, the topological filter and the label comparison complement one another and combine to yield more effective match evaluations.

For example, consider the two images in figures 1 and 5. Figure 15 (left) shows the result of comparing their second-difference labels, as in the later stages of my algorithm (sections 2.4–2.5) and in Nishihara's [1984] matcher. The raw count of label errors near each pixel does not clearly distinguish matching regions (e.g., left of the central building) from nonmatching ones (e.g., the central building). However, the pattern of the errors is different in the two cases. Most errors in the matching regions come from slight displacements of boundary locations, whereas many errors in nonmatching regions come from boundaries that have no plausible match in the other image.

The topological filter makes this difference visible by adjusting boundaries to eliminate slight differences in shape and location, such as those due to noise and differences in viewpoint, without changing the topological structure or the polarity of boundaries. Figure 15 (right) shows the output of this process. Left of the central building, errors have been reduced to small, isolated faults, whereas the building is still broken up by extended curves. Although isolated faults represent real changes in topology, they can reasonably be discounted as effects of noise (figure 16). A simple application of mathematical morphology (section 3.5) can separate the two classes of regions as in figure 5.

The key to doing this type of boundary adjustment is defining precisely what it means to adjust the position of a boundary without changing the topological structure of the image. In this section, I will first give a precise definition for the topological structure, then motivate why it should be the same for corresponding parts of two stereo images and survey related ideas from

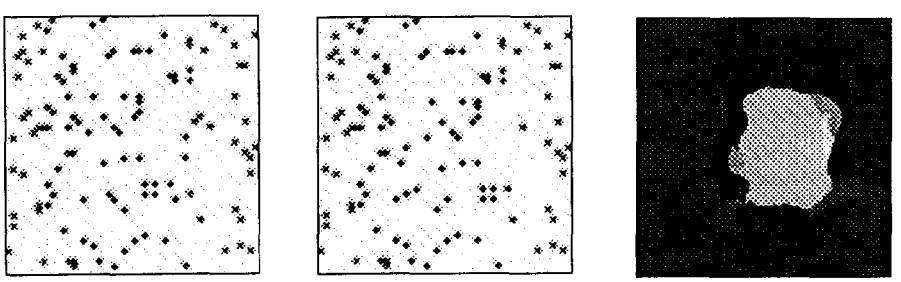

Fig. 11. A 200 by 200, 5% random-dot stereogram (dots 4 cells on a side) depicting a raised square displaced by 20 cells horizontally and 4 cells vertically, relative to the background.

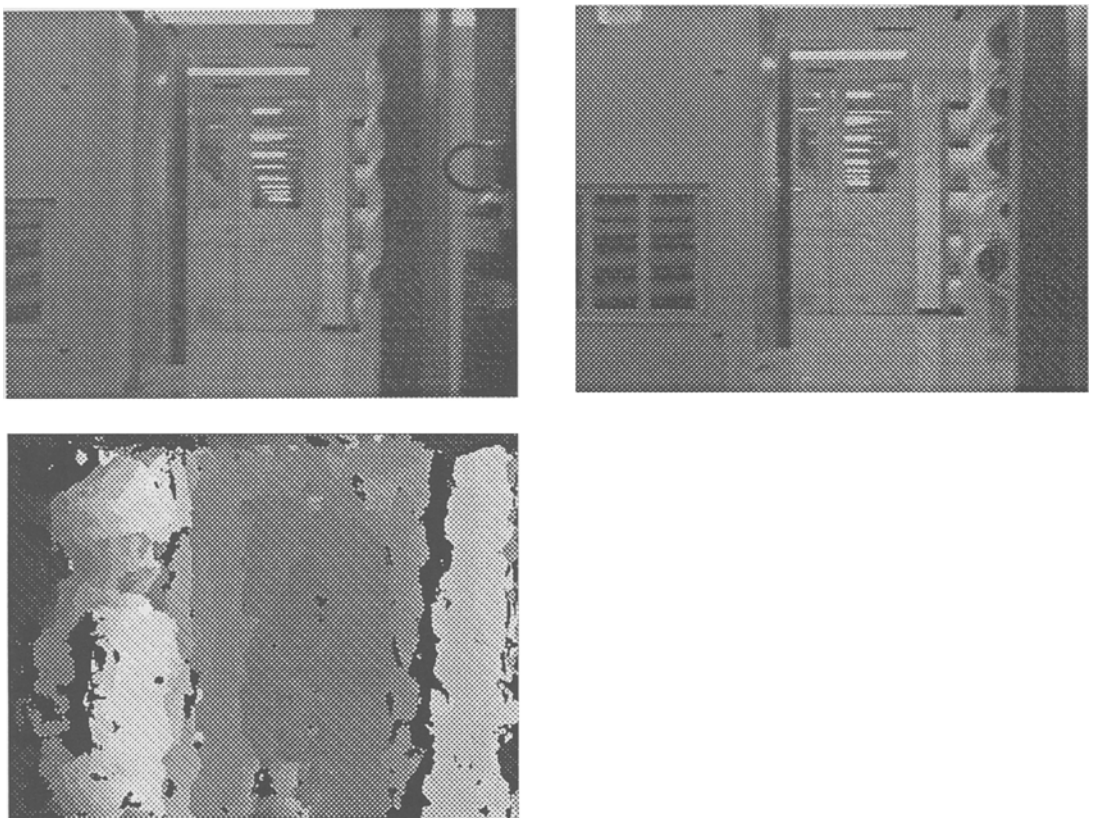

Fig. 12. A 400 by 300 stereogram depicting a corridor. The matcher computed a disparity range of [−71, 1], a 0.57° rotation, and a 8.4 cell vertical translation.

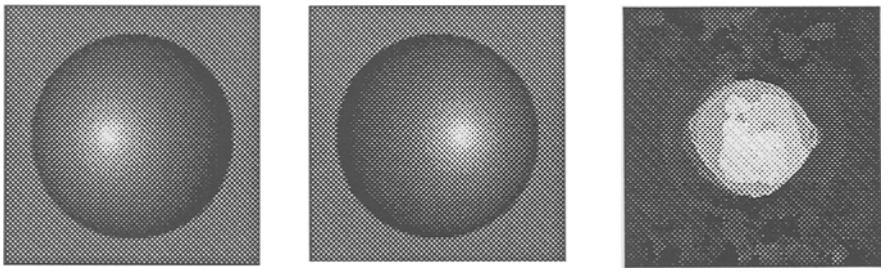

Fig. 13. A 200 by 200 stereogram depicting an ellipsoid with elongation $c = 4$ and disparities ranging from 40 cells at the center of the ellipsoid to 0 on its boundary. The disparity map is presented in cyclopean projection for easier analysis.

previous algorithms. The final two subsections work through the details of the boundary-adjustment algorithm and the final morphological cleaning step.

3.1 Defining Topological Structure Precisely

Standard definitions for the topology of a digitized image [Rosenfeld 1979] do not correspond closely to notions from standard mathematics. This makes it difficult to define and use topological properties more sophisticated than connectivity. I will sketch an alternative model which provides an exact correspondence between continuous and digitized topology. This model is developed more fully and applied to other domains by Fleck [1988a, b, 1990a].

I model digitized images as regular cell complexes. That is, each image is built out of space-filling 2D cells, one per pixel, each cell sharing edges and vertexes with

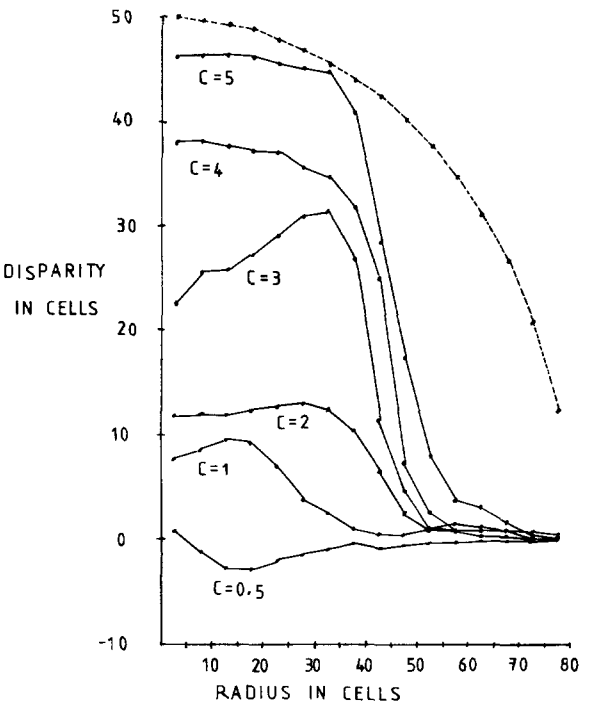

Fig. 14. Average disparities for ranges of distances, from the center of the ellipsoid (radius 0) to its boundary (radius 80). The solid curves show values computed for ellipsoids of 6 different elongations. The dotted curve shows the correct disparities for the $c = 5$ case.

its neighbors. The matcher presented here uses a pseudo-hexagonal arrangement of cells, shown in figure 17 [Fleck 1989]. However, a previous implementation [Fleck 1988b] used a rectangular arrangement, and the same techniques work for any topologically well-behaved tesselation of the image plane, even nonregular ones.[12]

The topological structure of an image only becomes interesting when boundaries are added to it. For the purposes of this article, I assume that these boundaries are supplied by an edge finder, such as the one described in section 2.3. This edge finder places boundaries at two types of locations: between cells and on cells. Most edge finders choose only one of these types of locations for reporting boundaries. However, allowing a mixture of the two types adds little complexity to the formal model and simplifies the description of boundary adjustment (section 3.4).

Thus, the boundaries in a digitized image are a set of vertexes, edges between cells and whole cells. I require that the boundaries be closed. That is, if the boundaries contain an edge, they must also contain its endpoints. If they contain a whole cell, they must also contain its edges and vertexes. The continuous model for the image is then produced by deleting all points in the boundaries, as illustrated in figure 18.[13] Intercell boundaries create gaps one point wide between regions, and on-cell boundaries create wider gaps. However, the topological structure of the remaining points does not depend on the thickness of the boundaries.

Using this model of boundaries, we can then formalize most of the topological constraints on matching by stating that:

> A stereo correspondence is a bijection from a subset of one image to a subset of the other, which is continuous and has a continuous inverse.

I do not require a match between the entire images, because some surfaces in one image may be occluded in the other. Furthermore, modeling sharp changes in

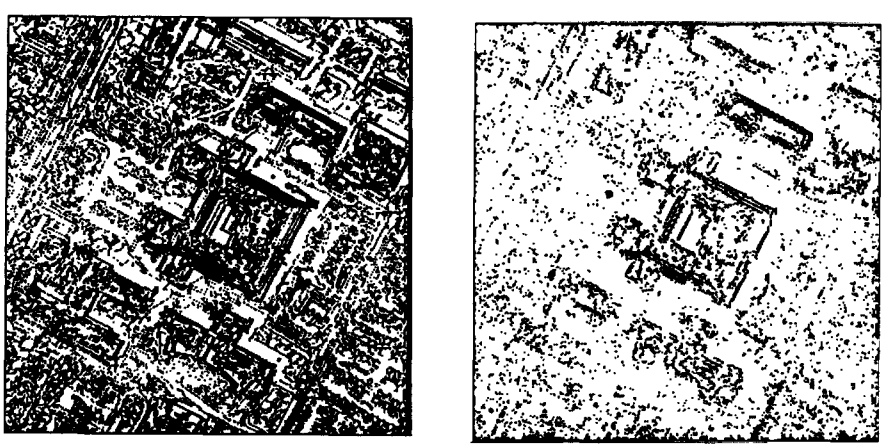

Fig. 15. Before adjustment (left), the two edge-finder maps from figure 5 contain many differences (black stripes). Adjustment removes errors arising from slight fluctuations in boundary location, leaving only real qualitative differences (right).

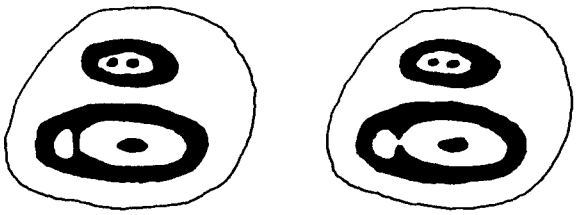

Fig. 16. The difference in topology between these two patches is due to changes in the labels of only a few pixels. This will generate an isolated fault after adjustment, which will be removed by morphological cleaning.

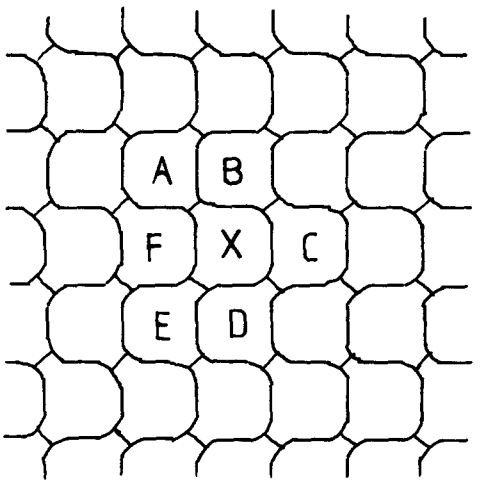

Fig. 17. In a pseudo-hexagonal tesselation, cells are approximately rectangular. However, cell corners are modified so that each cell (e.g., X) has only 6 neighbors (A to F).

disparity requires leaving a gap in the match (perhaps only one point wide). Requiring the correspondence to be continuous in both directions means that it must map regions separated by a boundary onto regions that are similarly separated (informally: boundaries onto boundaries).

These constraints explain most of the examples in figure 3, but would still allow a match between the model and example F. To prevent this, I impose the stronger requirement that the two spaces must be imbedded *isotopically* in the image plane [Rourke & Sanderson 1982]. That is:

DEFINITION: *Two imbeddings* $f: \rightarrow X$ *and* $g: A \rightarrow X$ *are* isotopic *if there is a continuous family of imbeddings* $F_i: A \rightarrow X$, *for* i *in* $[0, 1]$, *such that* $F_0 = f$ *and* $F_1 = g$.

(The related notion of homotopy lacks the requirement that the F_i be imbeddings.) Because it gradually deforms one imbedding into the other, an isotopy cannot remove regions from inside rings (in 2D) or spheres (in 3D). Nor can it change orientation of "handed" regions (e.g., a granny knot in 3D).

These constraints on stereo matches are defined for continuous functions. In general, it is not easy to define precisely when a digitized function corresponds to a particular continuous one. Fortunately, we will not need to do so. Instead, the topological filter will verify that there exists *at least one* continuous function that satisfies these topological conditions and involves only small divergences from the target disparity. Specifically, the boundary-adjustment phase (section 3.4) attempts to construct such a function for the entire image and the morphological cleaning phase (section 3.5) isolates the regions in which this construction was successful (making due allowance for image noise).

3.2 The Stability of Topological Structure

Clearly, the topological constraints defined in the previous section will be useful in matching only if corresponding patches of image typically have the same topological structure. In addition, the implementation of adjustment requires that boundary polarity typically be the same. In general, this seems to be the case, but a full formal explanation of this empirical observation would be quite difficult to construct. For example, it may depend on assumptions about how fast the structure of the world changes across scales. However, it is easy to see why these constraints should hold for certain, suggestive special cases.

Consider first the effects of imaging noise and changes in digitization. Raw edge-finder output is very noisy and not particularly constant between stereo images. However, practical edge-finder algorithms invariably incorporate some type of noise-suppression algorithm. These decrease the variation in boundary locations, topology, and polarity as noise and digitization are varied. In empirical studies of an edge finder closely related to the one used in this work, over 95% of the output for two images of the same scene passed the topological filter, despite varying noise and digitization. (See section 5 and Fleck [1988b] for details.)

Now, suppose that we vary the scene lighting, rather than the noise. Boundary locations and topology are both relatively stable, changing only when low-contrast boundaries become indistinguishable from noise and when shadow boundaries move. The first type of variation can largely be eliminated by taking the log transform of an image before running the edge finder (cf.

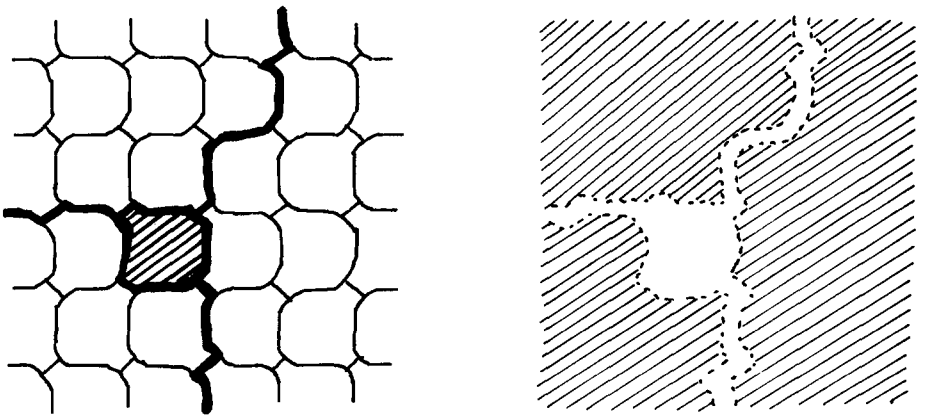

Fig. 18. Left: a set of boundaries in a pseudo-hexagonal tesselation. Inter-cell boundaries are indicated by thickened edges and/or vertexes of cells. On-cell boundaries are indicated by shaded cells. Right: a model for the topological structure of an image with these boundaries.

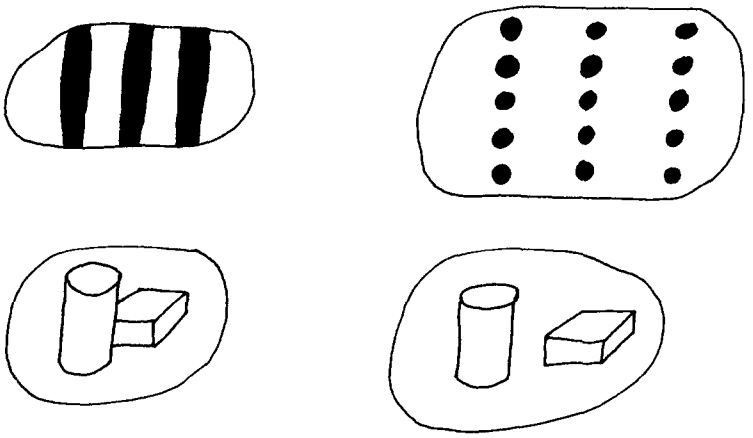

Fig. 19. A small change in viewpoint can change scale of representation (top) and occlusions (bottom).

Voorhees and Poggio [1987]), although this has not yet been incorporated into the algorithms presented here.[14] Similarly, except in rare special cases, changes in lighting do not change the polarity of boundaries.[15]

Finally, consider viewing a constant scene from two slightly different 3D viewpoints. There are two main sources of topological change: changes in (self-)occlusion and changes in scale of representation (figure 19). Both types of events can also change boundary polarity. However, local patches of image seem to show these types of changes only rarely. There are suggestive similarities between this empirical observation and results from singularity theory [Callahan 1974, 1977; Callahan & Weiss 1985; Clark 1988; Koenderink & van Doorn 1976], though these formal analyses have only been developed for smooth surfaces and infinite-precision measurements.

Let us assume, therefore, that we are looking at a local patch of smooth surface, away from any occlusions, whose representation is not affected qualitatively by any changes in scale. This surface patch may have markings due to changes in albedo, shadows, or 3D texture too small (relative to the change in viewing position) to display noticeable changes in self-occlusion. In such cases, projection maps the 3D patch smoothly onto both 2D views and thus induces a smooth deformation of one 2D view onto the other.[16] This is illustrated in figure 20. Clearly, neither the topological structure nor the boundary polarity will be altered, although the boundary shapes are noticeably deformed.

3.3 Previous Related Work

Topological information has not been widely used by previous stereo algorithms. A handful of previous algorithms [Baker & Binford 1981; Mayhew & Frisby 1980, 1981; Mohan, Medioni, & Nevatia 1989; Ohta & Kanada 1985; Grimson 1985] implement some form of *figural*

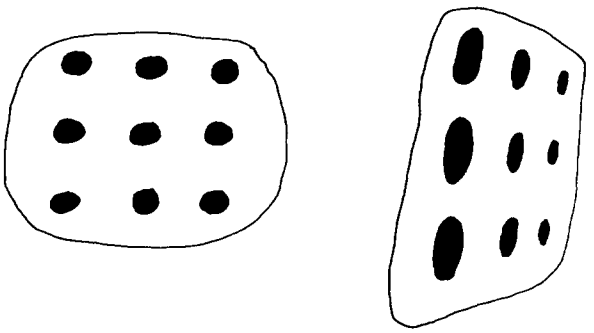

Fig. 20. Surface markings on a 3D surface, seen from two different viewpoints. One view can be smoothly deformed onto the other and they both have the same topological structure and boundary polarity.

continuity constraint, that is, a preference for matching connected boundaries to connected boundaries (figure 4). Such a constraint is implicit in algorithms that match extended boundary segments [Ayache & Faverjon 1987; Medioni & Nevatia 1985]. Finally, Chen [1985] presents evidence that topological structure may be used in human motion perception.

The new algorithm differs from the figural continuity proposals in three major ways. First, figural continuity algorithms consider only connectivity of boundaries, whereas the new constraints preserve the full topological structure of each region including, for example, whether it has any holes (formally: its homology or homotopy groups). Second, previous algorithms only combine information along individual boundary curves, whereas the topological filter considers how all nearby regions are imbedded in the image plane. So, for example, it can distinguish a ring from two filled circles. Finally, some proposals (e.g., see Grimson [1985]) will not split short curves, such as the boundary of a small dot. The new matcher can divide even a small dot among regions of two different disparities, if the context to both sides of the dot requires it.

In general, the new matcher takes a much more local view of topological structure than most previous work in computer vision. Topological examples presented in mathematics courses typically compare the entire topological structure of two very simple objects, for example, a doughnut and a coffee mug. Most applications in computer vision have followed this lead, for example, using Euler numbers for object recognition. By contrast, the new matcher is given a fine pattern of regions, out of which it must extract patches in which the match is well-behaved. A patch may, and often does, contain only part of some connected region in the original image. It may, and often does, contain small holes reflecting isolated faults.

Topological information is implicit in the locations of boundaries in the two images. Thus, placing tight constraints on the local shape of the disparity field, for example, a disparity-gradient bound, could theoretically impose constraints on matching similar to the topological ones. For robust performance on real images, however, implementations of these constraints (see sections 2.1 and 2.2) must make allowance for noise in boundary locations and match evaluations, as well as for sharp changes in disparity. Those that make such allowances (e.g., Pollard, Mayhew, and Grisby [1985]) do not even enforce the figural continuity constraint, whereas strict enforcement of constraints like the disparity gradient bound (e.g., see Drumheller and Poggio [1986]) causes some good matches to be rejected.

Boundary polarity is used by most stereo matching algorithms and some texture analyzers [Vilnrotter, Nevatia, & Price 1986]. In some cases, polarity is encoded as part of an "orientation" reported for each boundary segment. Psychophysical examples suggest that people are unable to fuse stereograms with reversed contrast, though large changes in contrast magnitude can be tolerated. Boundary polarity also plays a crucial role in parsing regions as shadows [Cavanagh 1987] and recognizing faces [Pearson & Robinson 1985].

3.4 Adjustment

The first step in implementing the topological filter is to construct the boundary-adjustment algorithm. This algorithm is given two images and a specified translation between them (i.e., the one determined by the (d_x, d_y) disparity pair currently under consideration). It then aligns the two images at that disparity and attempts to reconcile differences between their boundaries without altering the topological structure of either image. In the terms of section 3.1, it attempts to build an isotopy between the two images which never deviates very much from the translation.

Because adjustment is imbedded in an algorithm that searches all integer disparities, it can make some simplifying assumptions about its input:

1. Each boundary has moved at most a few cells relative to the specified translation.
2. Corresponding regions overlap.
3. Noncorresponding regions bearing the same dark/light label do not overlap.

Condition 1 requires boundary displacements to be small in an absolute sense; conditions 2 and 3 require them to be small relative to the size of regions. Since regions in edge-finder output are essentially always at least two pixels wide, a typical pixel has a nontrivial neighborhood in which these conditions hold, at the correct nearest-pixel disparity. Failures occur when the disparity field slopes rapidly (see section 2.1). These assumptions are essential to building an efficient algorithm: searching for isotopies under more general conditions could involve extensive search.

Adjustment starts by ensuring that the given translation between the two images (clearly continuous in both directions) preserves the cell structure of the two images (ignoring the edge-finder boundaries). For a rectangular cell arrangement, this is trivial, because the tesselations match exactly after an integer translation. For the pseudohexagonal tesselations used in this paper, it is sometimes necessary to swap the roles of even and odd rows in one image. This may induce topological changes in the image, but they are small enough to be ignored. In more general situations, it might be necessary to subdivide cells in one or both images to make the tesselations match [Fleck 1988b, 1990a].[17]

The algorithm then tries to adjust the boundaries in the two images, so as to make them identical. This is done by pushing cells into the boundaries, widening them in both images (figure 21). Each adjustment operation corresponds to any isotopy. Thus, the result of many adjustments is isotopic to the original image. And, so, where adjustment succeeds in making the boundaries identical, there must exist an isotopy between (subsets of) the two original images. If the edge finder's dark/light labels can also be made identical, this isotopy preserves boundary polarity.

Specifically, the adjustment algorithm first identifies *conflicts* between corresponding cells in the two images. Two cells conflict if one is in the boundaries and the other is not, or if they bear different dark/light labels. Consider two corresponding boundaries that are slightly misaligned, as in figure 21. If the correspondence meets conditions 2 and 3, cells to the same side of both boundaries will have the same label, whereas cells between them will conflict. Thus, the locations of conflict regions identify corresponding pairs of boundaries.

Conflicts are eliminated in two ways. First, cells neither in nor adjacent to any boundaries are relabeled *zero* if the corresponding cell in the other image is *zero*.

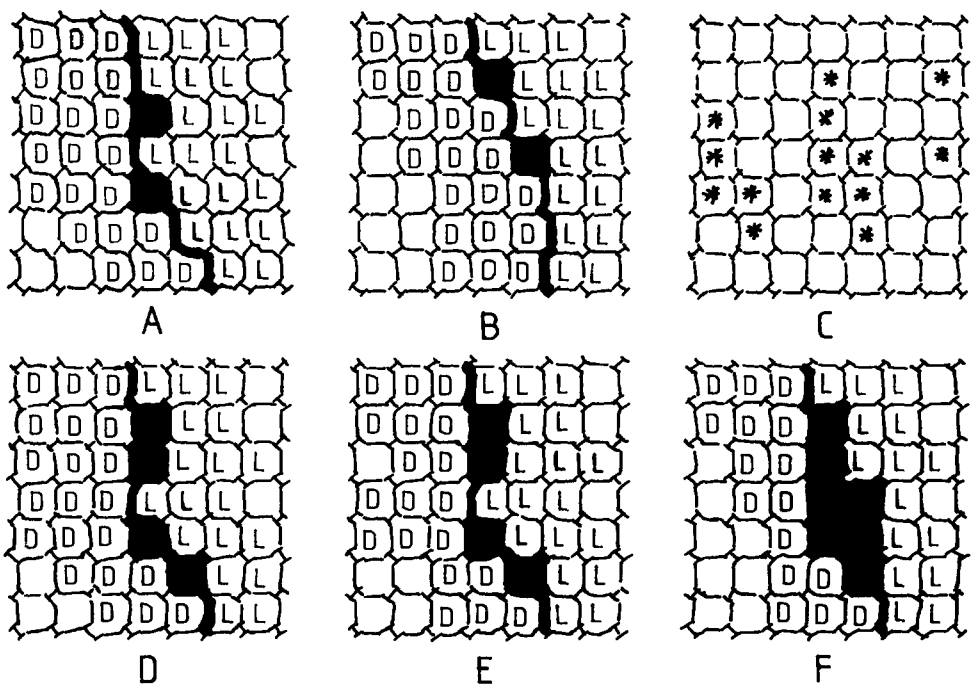

Fig. 21. A and B show the edge finder's boundaries and dark/light labels for two image patches. Adjustment identifies cells with conflicts (*C*) and moves them into the boundaries (*D*) or zeros their labels (*E*). By a sequence of such changes, A and B are reduced to a common form (*F*) without altering their topological structures.

Other conflicts are eliminated by moving conflicting cells into the boundaries, whenever this can be done without changing the topological structure of the image (figure 21).[18] In both cases, boundary polarity and topological structure are preserved. If these operations cannot successfully resolve a conflict, the images either differ in topological structure, differ in boundary polarity, or do not satisfy one of the three conditions.

To simplify analysis, cells are moved into the boundaries one at a time. In a pseudohexagonal tesselation, a nonboundary cell C can be reassigned to the boundaries if its edges consist of exactly two connected sections, one in the boundaries and one not in the boundaries.[19] Ignoring reflections, rotations, and minor changes of cell shape, there are six cases, shown in figure 22, in which a cell can be moved into the boundaries. I have included the case of an isolated-point boundary for completeness, but it cannot occur in the edge-finder output used in this article.

For each case, it is straightforward to show that, if the middle cell is moved into the boundary, the new space is isotopic to the old one. Consider the case shown in figure 23. A small neighborhood (N) is constructed along the nonboundary side of the cell (C), tapering so that its width limits to zero at the corners of the cell. We can ensure that N contains no boundaries by confining it to the four cells immediately adjacent to this side of C. The isotopy then gradually shrinks the region $C \cup N$ onto the region $N - (N \cap C)$, while keeping the rest of the space fixed.[20]

Figure 23 actually corresponds only to a special case of the third operation from figure 22, because the edges that could be either boundaries or nonboundaries are shown as nonboundaries. However, the construction will work even if some or all of these edges are boundaries. Furthermore, it can easily be modified for the other five cases. In fact, the techniques are relatively general and can be used to develop operations for other tesselations.

These local adjustment operations can be applied across the image in parallel, so long as each pass does not attempt to reassign two adjacent cells at the same time. To cover all cells, the current implementation uses four passes, each considering one fourth of the cells. Since these operations can reassign only cells immediately adjacent to boundaries, the four-pass process is repeated three times. Depending on details of boundary shape, misaligned boundaries can be adjusted if the location errors are at most 3–6 cells.

3.5 Cleaning Up the Match Evaluation

Unless two images happen to be exactly contrast reversals, some parts of the images will match after adjustment (see figure 15). A good match, however, requires correspondence between extended patches of the images. Thus, regions that are heavily fragmented by topological differences should be relabeled as nonmatching. Conversely, two similar patches of image may have slight differences in topological structure due to noise, which should be ignored in matching. The output should be a clean classification of the image, as in figure 5.

The current implementation achieves both of these goals using an adaptation of the *close* and *open* operations from mathematical morphology [Serra 1982]. Initially, each cell is classified as either *possible* or *impossible*, depending on whether it has a label conflict remaining after adjustment. The operation *propagate (label)* spreads one of the two labels from each cell bearing that label to its six immediate neighbors. Small patches of match are pruned by 3 iterations of *propagate (impossible)* followed by 3 iterations of *propagate (possible)*. Small holes in match regions are then filled by 3 iterations of *propagate (possible)* followed by 3 iterations of *propagate (impossible)*.

4 Searching for Stereo Matches

At each scale, the new stereo matcher compares the two images at a variety of possible disparities. These candidates include disparities found at the previous (next coarser) scale, plus a range of similar values. This section presents the exact search strategy. Its parameters are based on psychophysical measurements of human abilities and, in particular, it can find correspondences despite vertical displacements. However, it prevents the search space from exploding using the fact that the vertical disparity field has only a few degrees of freedom.

4.1 Camera Alignment

The new matcher assumes that its input images are in approximate, but not necessarily perfect, vertical alignment. Perfect alignment would enforce the so-called *epipolar constraint* and would mean that all disparities would be horizontal.[21] However, exact alignment is difficult to achieve in real stereo systems, particularly those in which the cameras move. Although there exist numerous systems for calibrating the initial alignment, camera positions may slip slowly from their commanded

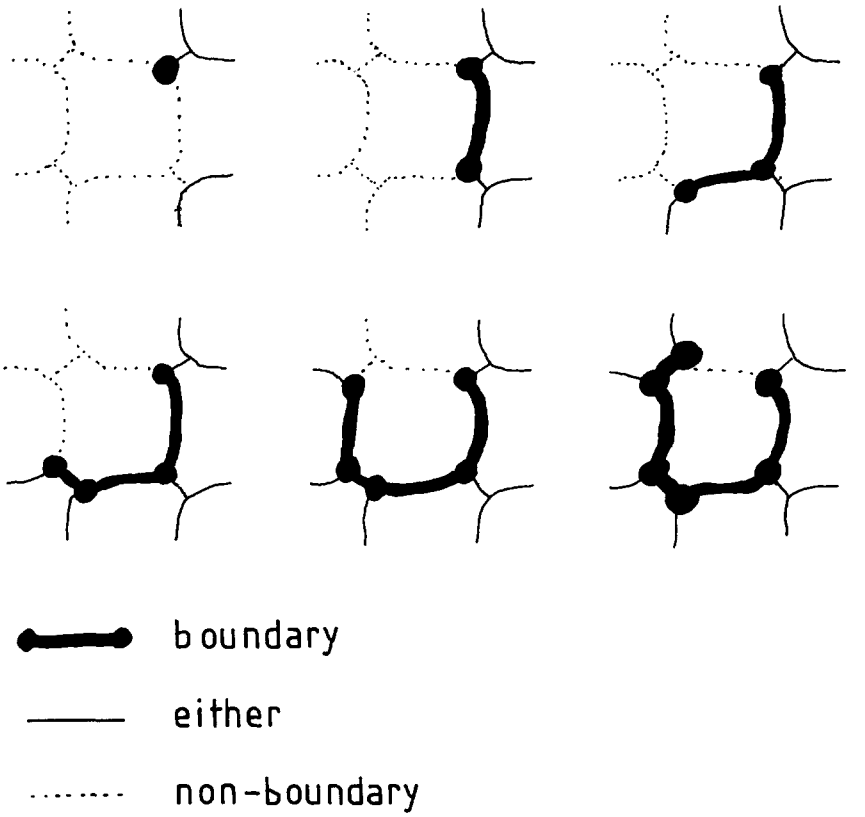

Fig. 22. In all of these boundary configurations, the middle cell can be moved into the boundaries without affecting the topological structure of the regions.

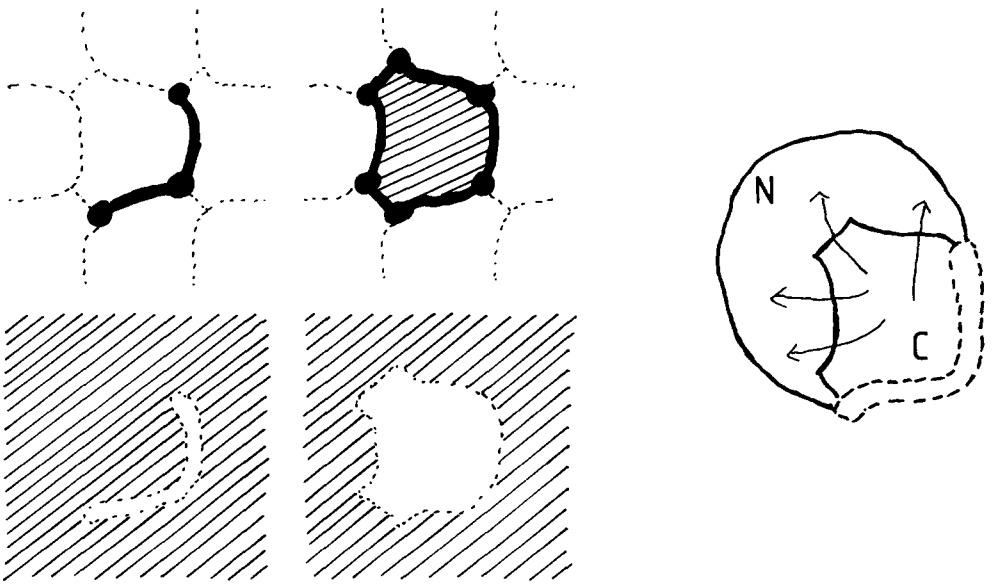

Fig. 23. Proving that one of the adjustment operations preserves topological structure. Top: the boundary configurations. Bottom: their underlying spaces. Right: an isotopy relating the two underlying space.

positions after repeated motions and eventually accumulate large errors. Objects or people may also collide with the cameras. Thus, it is an advantage for a stereo matcher to be able to tolerate small errors in vertical alignment and provide the information required to correct the alignment.[22]

In the introduction, we saw that searching for vertical, as well as horizontal, disparities opens up a much wider space of possible matches. All stereo algorithms produce some incorrect matches, even assuming vertical disparities are zero. Allowing even ± 2 cells vertical displacement multiplies the error rate at each scale by 5. Worse, in a multiscale matcher, each incorrect disparity spawns a whole colony of possibilities, some increasingly far from the correct vertical disparity. The only unrestricted 2D search I am aware of [Nishihara 1984] used few scales and small search neighborhoods. Another implementation [Gennert 1988] allows only small, smoothly varying vertical disparities and a third [Day & Muller 1989; Otto & Chau 1989] requires both components of disparity to vary slowly.

The possibilities can be reduced considerably by observing that the vertical disparity field has only a few degrees of freedom. Because the cameras are close to vertical alignment, differences in depth across the scene create only negligible differences in vertical disparity. Thus, the vertical disparity values depend only on the relative positions of the two cameras. Describing this requires only 6 parameters, one of which is used up modeling the horizontal separation of the cameras. In the current implementation, I assume that only two of these parameters have significant effects on the vertical disparity field: vertical translation of one image relative to the other and relative rotation about the centers of the images.

The new matcher uses this constraint to correct the vertical disparity field estimated at one scale before passing it on to the next scale. Specifically, it uses a least-squares method, weighted by matching strength, to model vertical disparity values as a linear function of horizontal distance from the image center. The slope and y-intercept of this line then determine a rotation about the image center and a vertical translation.[23] These two parameters are supplied to the computation at the next finer scale. In retrospect, least-squares is an unreliable way to fit a model to measured disparities, because they are subject to occasional large errors. Robust regression techniques [Rousseeuw & Leroy 1987; Hoaglin, Mosteller, & Tukey 1983] might yield more accurate results.

4.2 The Search for Possible Disparities

The search for possible stereo matches at each scale is supplied with a map of coarse-scale horizontal disparities from the previous scale, plus a two-parameter description of the vertical disparities. Because the matcher considers only local translations, any relative rotation of the images may undermine the quality of match evaluations. Therefore, the matcher begins by adjusting the positions of the two images, using the vertical disparity parameters, so as to make all vertical disparities zero.[24] The coarse-scale horizontal disparity field is also adjusted to compensate for the effects of any rotation. The edge finder is now run, at this and all coarser scales, on the adjusted images.[25]

The search algorithm then explores disparities similar to those in the coarse-scale horizontal disparity field. The algorithm allows substantial deviations from the coarse-scale values (± 17 cells horizontally; ± 4 vertically) to mimic human abilities (section 4.4). However, it uses a flexible exploration algorithm to avoid searching the full range when this is not necessary. The search grows outward from a smaller neighborhood of the coarse-scale values. It is continued only so long as it yields disparities that are *promising*, that is, that account for a nontrivial number of cells in the images.

The current algorithm defines a promising disparity to be one whose total strength over the two images is at least $560\sqrt{s}$, where s is the area of either the left or the right image.[26,27] At a scale where the image is 100 by 100, this corresponds to a 14 by 14 patch of moderately good (strength 143) match in both images. This heuristic seems to select a plausible set of candidates to explore, but it should not be regarded as a final (or even principled) solution to the problem of controlling flexible search.

To start the search, the algorithm extracts promising disparities from the coarse-scale horizontal disparity field.[28] For each promising disparity d_x, the first pass of the search matches the images at all disparities $(d_x + \epsilon_x, 0)$ where $\epsilon_x \in [-13, 13]$. As matching proceeds, the disparity map is updated, so that each cell always contains the best disparity found so far, plus its strength. Passes 2–5 of the search identify all promising disparities (d_x, d_y) at the current scale and search for matches at the 8 neighboring disparities $(d_x + \epsilon_x, d_y + \epsilon_y)$ where $\epsilon_x, \epsilon_y \in [-1, 1]$, $(\epsilon_x, \epsilon_y) \neq (0, 0)$.

4.3 Search Results

The matcher has been run on 30 real and synthetic stereo pairs, chosen to test the limits of its performance.

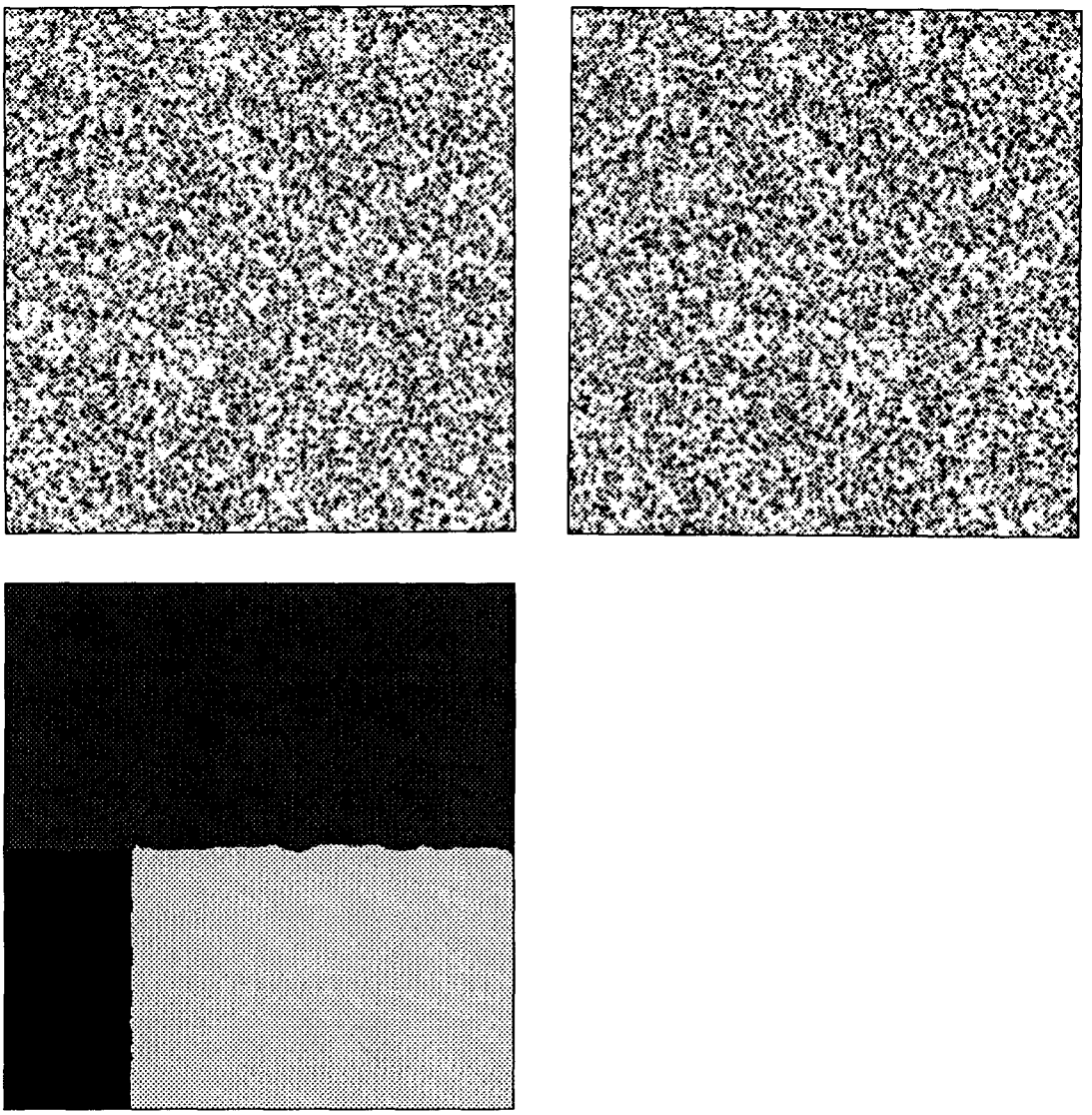

Fig. 24. A 400 by 400, 20% random-dot stereogram (dots 2 cell on a side) depicting two panels, one at disparity 0 and one at disparity 100.

Figure 24 shows that it can successfully fuse large ranges of horizontal disparities. Figures 25–26 show that it can fuse moderately large rotations and vertical translations between the two images. Figure 11 shows that it can fuse patches of image with small, but nonzero, vertical displacements relative to the rest of the image. Figures 2 and 12 show rotations and vertical translations in natural stereo pairs. Some stereo pairs can even be fused when edge maps at one (medium-resolution) scale are zeroed.

4.4 Comparison to Previous Work and Psychophysics

Compared to previous algorithms, the new stereo matcher explores quite wide ranges of disparities at each scale, because it was designed to model the full range of human abilities. Measuring search bounds for humans is complicated by multiscale processing. Horizontally, ranges of at least ± 20 cells of disparity (at the finest scale) can be fused at a single fixation [Poggio & Poggio 1984].[29] Experiments with high-pass filtered images [Mowforth, Mayhew, & Frisby 1981] suggest horizontal bounds of ± 13 cells at each scale.[30]

Vertically, measured bounds vary from ± 20 [Poggio & Poggio 1984] to ± 7 cells [Nielsen & Poggio 1983], for observation times too short to permit eye movement. However, poor form discrimination [Nielsen & Poggio 1983] suggests that these measurements may reflect only coarse-scale fusion and the (scale-relative) bounds

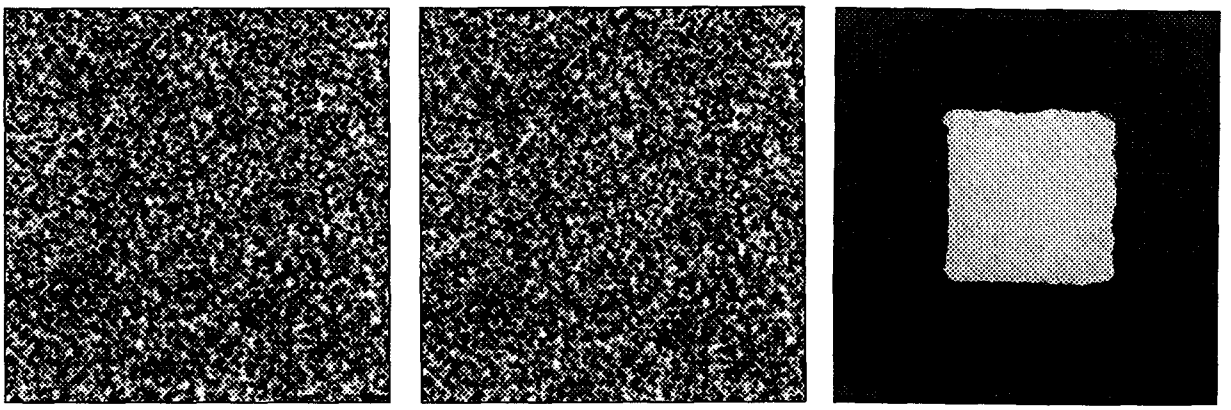

Fig. 25. A 300 by 300, 50% random-dot stereogram (dots 2 cells on a side) depicting a raised square at disparity 4 cells different from the background, the entire image having a 16 cell vertical translation.

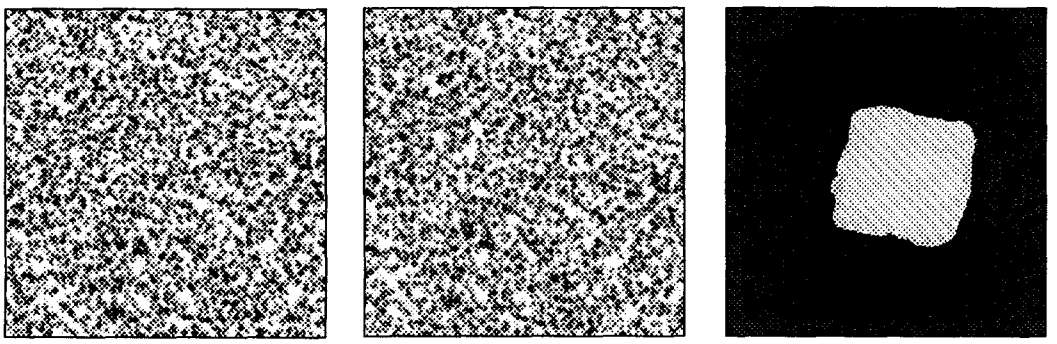

Fig. 26. A 250 by 250, 20% random-dot stereogram (dots 2 cells on a side) depicting a raised square at disparity 15 cells different from the background, the entire image having a 10° rotation. The square appears rotated in the output disparity map because the relative rotation was synthesized by rotating one image and the program corrected it by rotating the other.

could be much smaller. Although the new matcher can fuse images with large vertical disparities, its method of modeling vertical disparities prevents it from fusing a small patch of image displaced from the background by more than $\pm 4 - 7$ cells (depending on image contents). Such a difference in performance between overall vertical disparity and vertical disparity of patches also occurs psychophysically [Nielsen & Poggio 1983; Duwaer & van den Brink 1981].

Previous computer algorithms typically examine all fine-scale disparities within a fixed-size neighborhood of each disparity at the next-coarser scale. Table 2 lists neighborhood sizes for a number of recent multiscale algorithms:[31] Marr–Poggio–Grimson (Marr and Poggio [1985]; Nishihara [1984]; and Hoff and Ahuja [1989]. The table also shows search areas for two single-scale algorithms: PMF (Pollard, Mayhew, and Frisby [1985] and Drumheller and Poggio [1986]).[32] These algorithms, however, search a neighborhood of only one candidate (zero disparity) at the finest scale, whereas multiscale algorithms search neighborhoods of many candidates in stereograms with large disparity ranges. Larger bounds are found only in algorithms matching very sparse features (e.g., see Barnard & Thompson [1980]).[33]

Comparing these search bounds to the parameters of the new topological algorithm is slightly tricky, because the size of its search area depends on how many promising locations are found. The minimum search area is that of the first iteration, that is, 27 cells for a single coarse-scale suggestion. In my tests, the search at all iterations has covered 35–334 disparities, 1.2–4.5 times the number of disparities searched at the first iteration, equivalent to 33–120 candidates from a single coarse-scale disparity. This is noticeably larger than those of previous algorithms. Furthermore, the new matcher can handle a larger maximum displacement (17 cells horizontally, 4 cells vertically) from the coarse-scale match.

Table 2. Search bounds for stereo algorithms.

	Horizontal Tolerance	Vertical Tolerance	Search Area	Multiscale?
Topological	±13–15 cells	±0–4 cells	27–315 cells	yes
Marr–Poggio–Grimson	±4 cells	slight	≤27 cells	yes
Nishihara	±2 cells	±2 cells	25 cells	yes
Hoff and Ahuja	±5 cells	none	11 cells	yes
PMF	±30 cells	none	61 cells	no
Drumheller and Poggio	±14 cells	none	29 cells	no

Published results do not indicate clearly whether small search neighborhoods were chosen for efficiency or because the methods for evaluating matches started to break down. Only Grimson [1981a, b] and Kass [1983, 1987] have analyzed their algorithms statistically. Grimson's algorithm clearly cannot tolerate search neighborhoods much wider than those used. Kass's analysis suggests his algorithm may tolerate fairly wide bounds, but his assumption that images resemble Gaussian noise undermines the validity of the formal analysis and any quantitative bounds computed from it. Furthermore, his published results apparently used only small search areas. My new matcher has been tested with a range of search strategies (cf. Fleck [1988b]) and seems to be relatively insensitive to the number of alternative disparities considered.

5 Extensions, Applications, and Other Frills

5.1 Using the Matcher in Other Domains

Not surprisingly, the new stereo matching algorithm can also be used to match a pair of images from a motion sequence, as illustrated in figure 27. However, the search strategy must be modified. For motion matching, modeling of the vertical disparity field is deleted and vertical disparities from each scale passed on, unaltered, to the next. Search at each scale is started from promising alignments at the previous scale, together with their 8 immediate neighbors. Exploration of alignments that prove promising at the current scale than proceeds as in the stereo matcher. Although some of the interframe motions are large (up to 10 cells) and the hand is moving nonrigidly, the reconstructed motion field is plausible.

The matcher can also be used to find the period of textured patterns. For example, in figure 28, patches of image in the top left-hand corner and in the bottom half are periodic, for example, they approximately match themselves if shifted some distance horizontally. The required distance is different for the two regions. The upper right-hand corner has no such self-similarity. For formal definitions related to periodicity see Grünbaum and Shephard [1987].

To compute the horizontal period of a texture, the stereo algorithm is modified so that only a 1D set of disparities is considered and negative disparities are forbidden. To avoid the peak in matching strengths near zero disparity, disparities of less than 4 cells and those whose strength is less than that of the next smaller displacement are suppressed. The strength at each disparity is reduced by 0.2 times the magnitude of the disparity, to favor the smallest period explaining the data. Finally, only one output map is produced: the strength for disparity (dx, dy) at cell (x, y) is used to update both cell (x, y) and cell $(x + dx, y + dy)$ in this map.

For the example, in figure 28, approximately correct periods are computed for the two periodic regions whereas the aperiodic region generates only small patches of random values. A full periodicity analysis algorithm would require searching for matches in several directions and is beyond the scope of this article. However, it is clear that this method has the potential to detect changes in texture period across the image. Previous algorithms for detecting periodicity [Bajcsy 1973; Matsuyama, Miura & Nagao 1983; Vilnrotter, Nevatia, & Price 1986] combine data from the entire image, making analysis of spatial variation impossible.

Finally, the matcher can also be used for evaluating the performance of different edge finders, or different parameter settings for a fixed algorithm. In this application, the correspondence between the images is fixed and the matcher must simply verify which patches match successfully. Figure 29 shows edge-finder output for two images of the same scene, but with differences in noise and digitization.[34] When noise suppression is not used, the images only match where there are strong features. When appropriate noise suppression is used, most of the images match successfully, confirming that noise suppression algorithm has made the edge finder's output more stable in the presence of camera noise. The

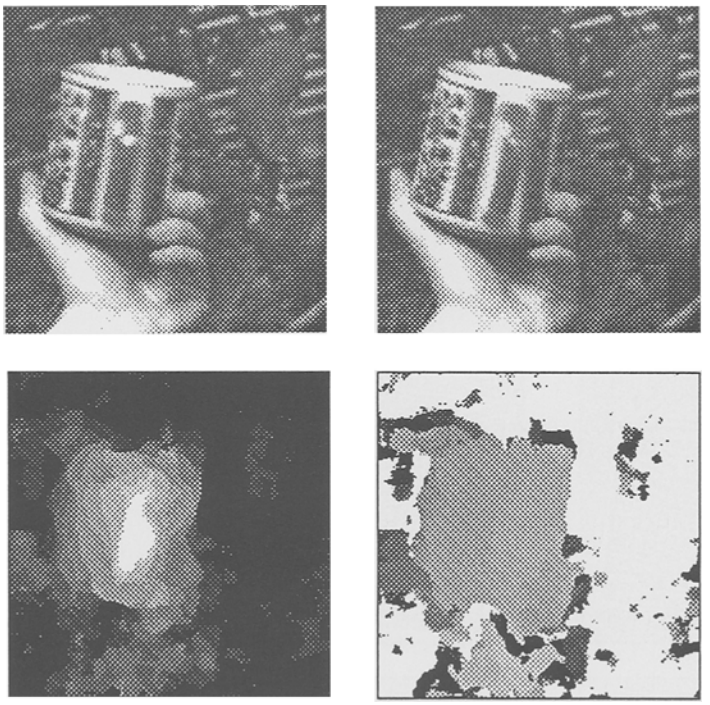

Fig. 27. Matching results for two 252 by 252 frames from a motion sequence, depicting a hand rotating a cup about a vertical axis. Apparent motion in the image plane was computed using a modified version of the stereo matcher. The left-hand image shows the computed motion magnitudes and the right-hand image the computed motion directions. Because the true space of directions is circular, both very light and very dark grey represent motion to the right. Areas with zero motion (and thus undefined direction) are shown in white.

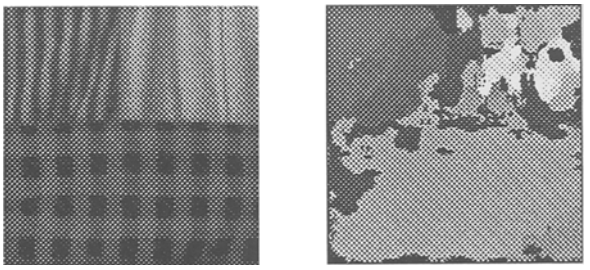

Fig. 28. Left: a 200 by 200 image of cloth textures. Right: computed horizontal periods: about 15 cells in the upper left-hand corner and about 26–27 cells in the lower half.

amount of fluctuation in boundary locations can also be computed, as the number of cells relabeled by adjustment divided by a measure of the total length of boundary in the region [Fleck 1988b].

Most edge finders show a trade-off between stability and number of reported boundaries, as parameters are varied. Comparative assessments of performance on a pair of images can be produced by plotting the percentage of the images successfully matched against the number of boundaries in the matched regions.[35] Figure 30 shows an example of this type of evaluation from a study [Fleck 1988b] of the performance of Canny's [1983, 1986] edge finder and the Phantom edge finder [Fleck 1988a, b]. Such plots can be used both for comparing the performance of competing edge finders and for choosing good settings of parameters for each individual algorithm. The matcher enabled this study to use complex real images, making its results more useful than those of previous studies [Fram & Deutsch 1975; Haralick 1984; Nalwa & Binford 1986; Pratt 1978; Sher 1987a, b] based on simple, synthetic images.

5.2 Subpixel Interpolation

The above stereo algorithm produces horizontal disparities only to the nearest pixel, but a postprocessor refines them to subpixel precision.[36] This is vital for capturing the full detail in low-disparity images such as figure 1 (see table 1). Suppose that the images have a constant disparity and are aligned exactly. If we shift one image horizontally against the other, the matching strength at each cell should decrease linearly with the amount of

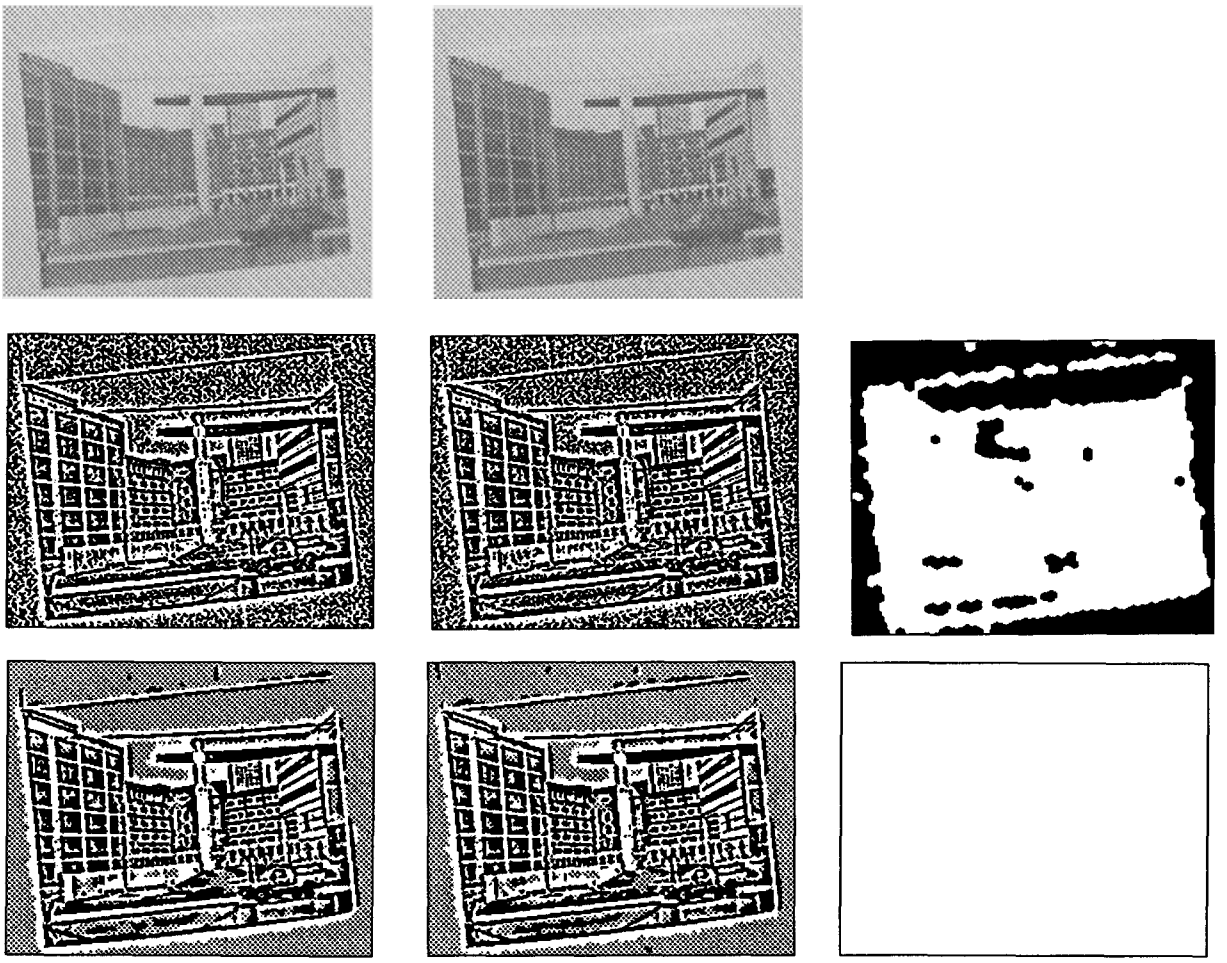

Fig. 29. Two 288 by 224 images of the same scene with different noise and digitization (top). Without noise suppression, the edge-finder outputs differ in many areas (middle). After noise suppression (bottom), the matcher detects no qualitative differences between the two outputs.

motion. This approximation will hold until significant numbers of boundaries start to cross noncorresponding boundaries, which requires at least two pixels of motion even for dense textures. Since the rate of decrease should be the same for equivalent motion in the opposite direction, matching strengths should form a symmetrical peak pattern, as in figure 31. Thus, if the nearest-pixel disparity is (d_x, d_y), the matcher can estimate the peak location from the strengths at $(d_x - 1, d_y)$, (d_x, d_y), and $(d_x + 1, d_y)$.

This method of estimating subpixel disparities is similar to the sign-correlation technique used by Nishihara [1984] and related to psychophysical models in which data is averaged along boundaries [Krotkov 1986; Watt & Morgan 1983]. It works because a curve in a digitized image typically hits the digitization in a variety of different ways along its length. Since a patch of image typically contains one or more lengths of boundary, averaging disparities or matching strengths over the patch increases location accuracy.

The output of many edge finders can be interpolated to sub-pixel precision [Boie & Cox 1987; Boie, Cox, & Rehak 1986; Hildreth 1983, 1984; Huertas & Medioni 1986; Nalwa 1987; Young 1986] and such interpolation is often motivated by the need to provide subpixel stereo disparities. However, manipulating subpixel boundary locations seems to require either expanding images to a cumbersome size or adding other complexity to the

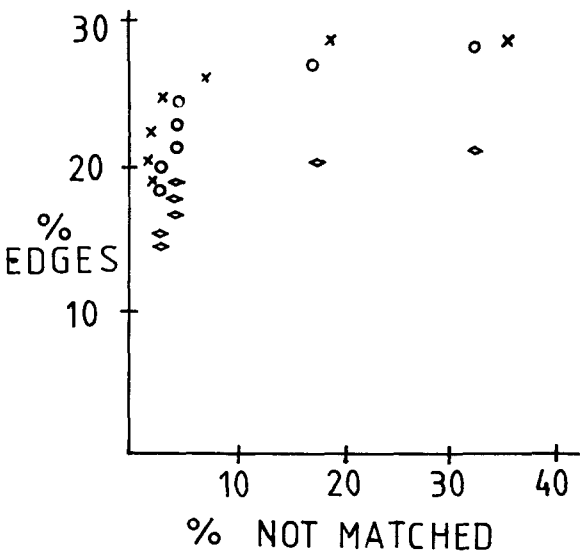

Fig. 30. An example of a graph comparing the performance of Canny's edge finder (circles and lozanges show data for two alternative methods of counting boundaries) to that of the Phantom edge finder (X marks). This graph shows the percentage of the image that matched, plotted against a measure of how many boundaries were detected, for a number of settings of the edge finders' noise thresholds. Values that are higher and to the left represent better compromises between stability and returning as many boundaries as possible.

representation. It is not clear that the extra information provided by subpixel boundaries is useful except in high-precision industrial applications.

The implemented interpolation algorithm is complicated by the fact that neither disparity field is constant. It is not symmetrical in the two images: without loss of generality, I will describe refinement of the left-hand half of the disparity map. The algorithm considers each horizontal disparity d_x individually. It masks off the area of the left-hand image at this disparity, finds the corresponding areas of the right-hand images at disparities $d_x - 1$, d_x, and $d_x + 1$, evaluates the match at $d_x - 1$, d_x, and $d_x + 1$, and then interpolates the subpixel disparity values for cells at disparity d_x. Evaluations are computed by treating the masked-off areas just like the *possible* areas generated by the topological filter and applying the single-scale evaluation algorithm described in section 2.4.

There are two tricks to making this work. First, although refined values will only be computed for cells at disparity d_x, the masked-off area must also include cells at disparities $d_x - 1$ and $d_x + 1$ to avoid artifacts when disparities change smoothly. Second, selection of corresponding cells in the right-hand image must take account of the vertical disparity values. Specifically, suppose that we are evaluating the match at disparity d_x and that cell (x, y) in the left-hand image has disparity $(d_{x'}, d_{y'})$. In computing the evaluation, the edge-finder labels at cell (x, y) are compared to the labels at cell $(x + d_x, y + d_{y'})$ in the right-hand image.

The current implementation computes disparities to the nearest 20th of a cell. To determine how much of this precision is reliable, I produced the synthetic image shown in figure 32. Although the square is only half a cell different in disparity from the background, it is clearly visible in the stereo output. Figure 33 shows the distributions of disparity values for the square and background, taken from both left and right disparity maps and ignoring locations within 2 cells of the square boundary. The distributions are well separated, with means of −0.005 and −0.491 cells and standard deviations of 0.046 and 0.103 cells. Thus, under good conditions, computed disparities could be accurate to perhaps 1/5 of a cell, similar to human performance [Poggio & Poggio 1984]. Of course, under bad conditions (e.g., sparse image features) disparities might not even be accurate to the nearest cell.

6 Conclusions

This paper has presented a stereo algorithm incorporating two new features:

- A search strategy that takes advantage of the fact that correct vertical disparity fields have only a few degrees of freedom, and
- A filter that ensures corresponding patches of image share the same topological structure and boundary polarity.

The new algorithm can fuse the usual range of stereo examples, producing clean subpixel disparity maps. Unlike previous algorithms, however, it has been shown capable of fusing stereo pairs involving significant vertical misalignment. Furthermore, it correctly reconstructs most sharp changes in disparity as sharp.

The main disadvantage of the current implementation is that it is slow. One of the larger and more complicated examples in this article (figure 2) took 11.6 hours to run on a Sun 4, plus 4.3 hours for subpixel interpolation. However, the algorithm uses only simple, local operations and would run much faster on appropriate parallel hardware. Furthermore, in a real-time system or model of human perception, the work involved in correcting large vertical misalignments might plausibly be spread over several timesteps.

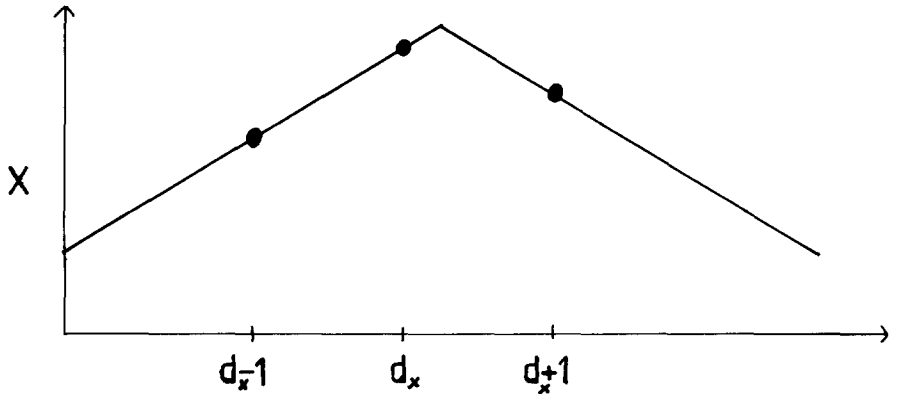

Fig. 31. In subpixel interpolation, the three measurements should ideally be samples along a peak pattern, with the slopes of both lines equal.

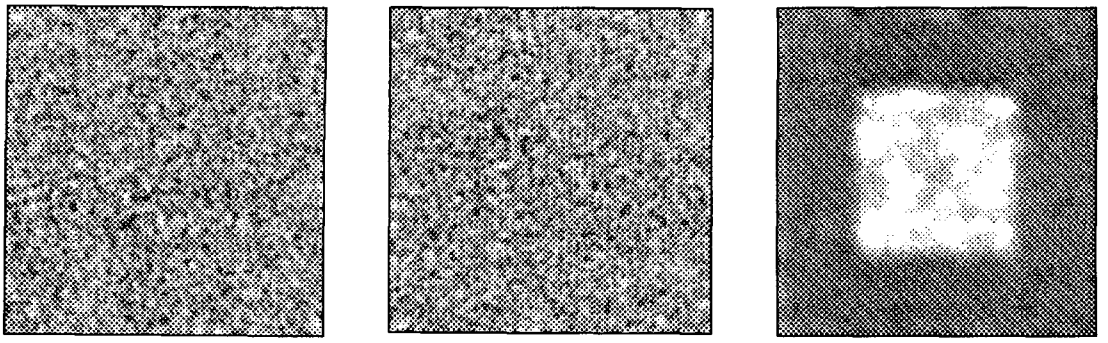

Fig. 32. A 250 by 250 stereogram depicting a raised square at disparity 0.5 cell on a background at zero disparity. This was produced by subsampling a 500 by 500, 20% random-dot stereo pair (dots 2 cells on a side) depicting a raised square at a 1-cell disparity.

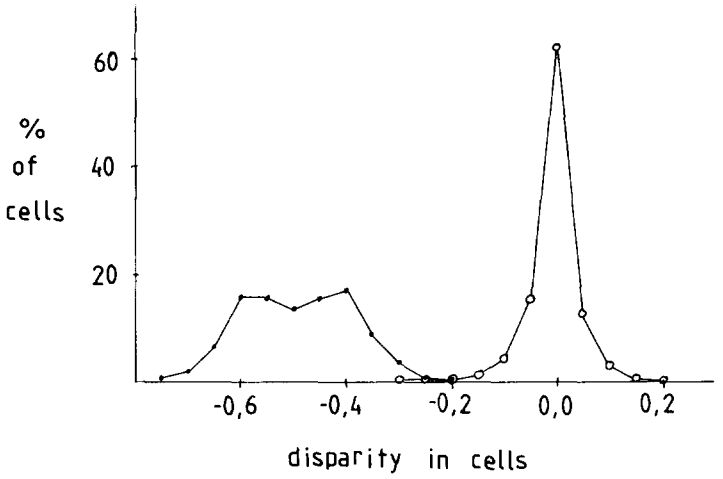

Fig. 33. Histogram of disparity values for the square (solid dots) and background (open dots) in figure 32. Counts are expressed as percentages of the total image area occupied by the square or background.

Acknowledgments

Comments on this work and/or useful pointers were provided by Hal Abelson, Mike Brady, Rod Brooks, Heinrich Bülthoff, James Callahan, David Forsyth, Eric Grimson, Ellen Hildreth, Alison Noble, and an anonymous reviewer.

This research was started at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology and continued at the Department of Engineering Science, Oxford. Support for the MIT AI Laboratory's artificial intelligence is provided in part by the Advanced Research Projects Agency of the Department of Defense under Office of Naval Research contract N00014-85-K-0124. The author was also supported by the Fannie and John Hertz Foundation, by the AT&T Bell Laboratories Graduate Research Program for Women, and by a postdoctoral fellowship sponsored by British Petroleum.

References

Ayache, N., and Faverjon, B. 1987. Efficient registration of stereo images by matching graph descriptions of edge segments. *Intern. J. Comput. Vision* 1: 107-131.

Bajcsy, R. 1973. Computer identification of visual surfaces. *Comput. Graph. Image Proc.* 2: 118-130

Baker, H.H., and Binford, T.O. 1981. Depth from edge and intensity based stereo. *Proc. 7th Intern. Joint Conf. Artif. Intell., Vancouver*, pp. 631-636.

Barnard, S.T. 1989. Stochastic stereo matching over scale. *Intern. J. Comput. Vision* 3(1): 17-32.

Barnard, S.T., and Fischler, M.A. 1982. Computational stereo. *Computing Surveys* 14: 553-572.

Barnard, S.T., and Thompson, W.B. 1980. Disparity analysis of images. *IEEE Trans. Patt. Anal. Mach. Intell.* 2: 333-340.

Boie, R.A., and Cox, I.J. 1987. Two dimensional optimal edge recognition using matched and Wiener filters for machine vision. *Proc. Intern. Conf. Comput Vision*, London, pp. 450-456.

Boie, R.A., Cox, I.J., and Rehak, P. 1986. On optimum edge recognition using matched filters. *Proc. Conf. Comput. Vision Patt. Recog.* Miami Beach, pp. 100-108.

Bolles, R.C., Baker, H.H., and Marimont, D.H. 1987. Epipolar-plane image analysis: An approach to determining structure from motion. *Intern. J. Comput. Vision* 1: 7-55.

Bovik, A.C., Clark, M., and Geisler, W.S. 1987. Computational texture analysis using localized spatial filtering. *Proc. IEEE Comput. Soc. Work. Comput. Vision*, pp. 201-206.

Boyer, K.L., and Kak, A.C. 1988. Structural stereopsis for 3-D vision. *IEEE Trans. Patt. Anal. Mach. Intell.* 10: 144-166.

Bülthoff, H.H., and Mallot, H.A. 1988. Interaction of depth modules: Stereo and shading. *J. Opt. Soc. Amer.* 5: 1749-1758.

Burt, P., and Julesz, B. 1980a. Modifications of the classical notion of Panum's fusional area. *Perception* 9: 671-682.

Burt, P., and Julesz, B. 1980b. A disparity gradient limit for binocular fusion. *Science* 208: 615-617.

Buxton, B.F., and Buxton, H. 1984. Computation of optic flow from the motion of edge features in image sequences. *Image and Vision Comput.* 2: 59-75.

Callahan, J. 1974. Singularities and plane maps. *Amer. Math. Monthly* 81: 211-240.

Callahan, J. 1977. Singularities and plane maps II: Sketching catastrophes. *Amer. Math. Monthly* 84: 765-803.

Callahan, J., and Weiss, R. 1985. A model for describing surface shape. *Proc. Conf. Comput. Vision Patt. Recog.* San Francisco, pp. 240-245.

Canny, J.F. 1983. Finding edges and lines in images. M.S. thesis, Mass. Inst. of Technol., also *Artif. Intell. Lab. Techn. Reprt.* 720.

Canny, J.F. 1986. A computational approach to edge detection. *IEEE Trans. Patt. Anal. Mach. Intell.* 8: 679-698.

Cavanagh, P. 1987. Reconstructing the third dimension: Interactions between color, texture, motion, binocular disparity, and shape. *Comput. Vision Graph., Image Process.* 37: 171-195.

Chen, L. 1985. Topological structure in the perception of apparent motion. *Perception* 14: 197-208.

Clark, J.J. 1988. Singularity theory and phantom edges in scale space. *IEEE Trans. Patt. Anal. Mach. Intell.* 10: 720-727.

Clark, J.J. 1989. Authenticating edges produced by zero-crossing algorithms. *IEEE Trans. Patt. Anal. Mach. Intell.* 11: 43-57.

Day, T., and Muller, J-P. 1989. Digital elevation model production by stereo-matching spot image-pairs: A comparison of algorithms. *Image Vision Comput.* 7(2): 95-101.

Drumheller, M., and Poggio, T. 1986. On parallel stereo. *Proc. IEEE Intern. Conf. Robot. Autom.*, San Francisco, pp. 1439-1448.

Duwaer, A.L., and van den Brink, G. 1981. Diplopia thresholds and the initiation of vergence eye movements. *Vision Research* 21: 1727-1737.

Fleck, M.M. 1988a. Representing space for practical reasoning. *Image Vision Comput.* 6: 75-86.

Fleck, M.M. 1988b. Boundaries and topological algorithms. Ph.D. thesis, Mass. Inst. of Technol., also *Artif. Intell. Lab. Techn. Reprt.* 1065.

Fleck, M.M. 1989. Spectre: An improved phantom edge finder. *Proc. Alvey Vision Conf.*, pp. 127-132.

Fleck, M.M. 1990a. Topological models for space and time. OUEL Report No. 1860/90, Oxford Univ., Dept. of Eng. Science.

Fleck, M.M. 1990b. Multiple widths yield reliable finite differences. *Proc. 3rd Intern. Conf. Comput. Vision*, December, Osaka, Japan, pp. 58-61.

Fleet, D.J., and Jepson, A.D. 1989. Computation of normal velocity from local phase information. *Proc. IEEE Conf. Comput. Vision Patt. Recog.*, San Diego, pp. 379-386.

Fram, J.R., and Deutsch, E.S. 1975. On the quantitative evaluation of edge detection schemes and their comparison with human performance. *IEEE Trans. Comput.* C-24(6): 616-628.

Gennert, M.A. 1987. A computational framework for understanding problems in stereo vision. Sc.D. thesis, MIT, Cambridge, MA.

Gennert, M.A. 1988. Brightness-based stereo matching. *Proc. 2nd Intern. Conf. Comput. Vision*, Tampa, FL, pp. 139-143.

Gennery, D.B. 1977. A stereo vision system for an autonomous vehicle. *Proc. 5th Intern. Joint Conf. Artif. Intell.*, Cambridge, MA, pp. 576-582.

Gillett, W.E. 1988. Issues in parallel stereo matching. M.S. thesis, Mass. Inst. of Technol., Cambridge, MA.

Grimson, W.E.L. 1981a. *From Images to Surfaces: A Computation Study of the Human Early Visual System*, MIT Press: Cambridge, MA.

Grimson, W.E.L. 1981b. A computer implementation of a theory of human stereo vision. *Phil. Trans. Roy. Soc. London B* 292: 217–253.

Grimson, W.E.L. 1985. Computational experiments with a feature based stereo algorithm. *IEEE Trans. Patt. Anal. Mach. Intell.* 7: 17–34.

Grimson, W.E.L., and Pavlidis, T. 1985. Discontinuity detection for visual surface reconstruction. *Comput. Vision, Graph., Image Process.* 30: 316–330.

Grünbaum, B., and Shephard, G.C. 1987. *Tilings and Patterns*. W.H. Freeman: New York.

Hannah, M.J. 1980. Bootstrap stereo. *Proc. DARPA Image Underst. Work.* College Park, MD, pp. 201–208.

Haralick, R.M. 1984. Digital step edges from zero crossings of second directional differences. *IEEE Trans. Patt. Anal. Mach. Intell.* 6: 58–68.

Heeger, D.J. 1987. Optical flow using spatiotemporal filters. *Intern. J. Comput. Vision* 1: 279–301.

Hildreth, E. 1983. The detection of intensity changes by computer and biological vision system. *Comut. Vision, Graph., Image Process.* 22: 1–27.

Hildreth, E., 1984. *The Measurement of Visual Motion*. MIT Press: Cambridge, MA.

Hoaglin, D.C., Mosteller, F., and Tukey, J.W., Eds., 1983. *Understanding Robust and Exploratory Data Analysis*. John Wiley: New York.

Hoff, W., and Ahuja, N. 1989. Surfaces from stereo: Integrating feature matching, disparity estimation, and contour detection. *IEEE Trans. Patt. Anal. Mach. Intell.* 11: 121–136.

Huertas, A., and Medioni, G. 1986. Detection of intensity changes with subpixel accuracy using Laplacian-Gaussian masks. *IEEE Trans. Patt. Anal. Mach. Intell.* 8: 651–664.

Jepson, A.D., and Jenkin, M.R.M. 1989. The fast computation of disparity from phase differences. *Proc. IEEE Conf. Comput. Vision Patt. Recog.*, San Diego, pp. 398–403.

Kass, M. 1983. Computing visual correspondence. *Proc DARPA Image Underst. Work.*, Arlington, VA, pp. 54–60.

Kass, M. 1987. Linear image features in stereopsis. *Intern. J. Comput. Vision* 1: 347–368.

Kass, M., and Witkin, A. 1985. Analyzing oriented patterns. *Proc. 9th Intern. Joint Conf. Artif. Intell.*, Los Angeles, pp. 944–952.

Kass, M., and Witkin, A. 1987. Analyzing oriented patterns. *Comput. Vision Graph. Image Process.* 37: 362–385.

Koenderink, J.J., and van Doorn, A.J. 1976. The singularities of the visual mapping. *Biological Cybernetics* 24: 51–59.

Krol, J.D., and van de Grind, W.A. 1980. The double-nail illusion: Experiments on binocular vision with nails, needles, and pins. *Perception* 9: 651–669.

Krotkov, E.P. 1986. Visual hyperacuity: Representation and computation of high precision position information. *Comput. Vision, Graph., Image Process.* 33: 99–115.

Lawton, D. 1983. Processing translational motion sequences. *Comput. Vision Graph., Image Process.* 22: 116–144.

Levine, M.D., O'Handley, D.A., and Yagi, G.M. 1973. Computer determination of depth maps. *Comput. Graph. Image Process.* 2: 131–150.

Little, J., Bültoff, and Poggio, T. 1987. Parallel optical flow computation. *Proc. DARPA Image Underst. Work.*, Los Angeles, pp. 915–920.

Little, J., and Gillett, W. 1990. Direct evidence for occlusion in stereo and motion. *1st Europ. Conf. Comput. Vision*, Antibes, France, pp. 336–340.

Marr, D., and Hildreth, E. 1980. Theory of edge detection. *Phil. Trans. Roy. Soc. London B* 207: 187–217.

Marr, D., Palm, G., and Poggio, T. 1978. Analysis of a cooperative stereo algorithm. *Biological Cybernetics* 28: 223–239.

Marr, D., and Poggio, T. 1976. Cooperative computation of stereo disparity. *Science* 194: 283–287.

Marr, D., and Poggio, T. 1979. A computational theory of human stereo vision. *Phil. Trans. Roy. Soc. London B* 204: 301–328.

Matsuyama, T., Miura, S., and Nagao, M. 1983. Structural analysis of natural textures by Fourier transformation. *Comput. Vision Graph., Image Process.* 24: 347–362.

Mayhew, J.E.W., and Frisby, J.P. 1980. The computation of binocular edges. *Perception* 9: 69–86.

Mayhew, J.E.W., and Frisby, J.P. 1981. Psychophysical and computational studies towards a theory of human stereopsis. *Artificial Intelligence* 17: 349–385.

Medioni, G., and Nevatia, R. 1985. Segment-based stereo matching. *Comput. Vision, Graph., Image Process.* 31: 2–18.

Mohan, R., Medioni, G., and Nevatia, R. 1989. Stereo error detection, correction, and evaluation. *IEEE Trans. Patt. Anal. Mach. Intell.* 11: 113–120.

Moravec, H.P. 1977. Towards automatic visual obstacle avoidance. *Proc. Intern. Joint Conf. Artif. Intell.* p. 584.

Moravec, H.P. 1981. Rover visual obstacle avoidance. *Proc. 7th Intern. Joint Conf. Artif. Intell.*, Vancouver, pp. 785–790.

Mori, K., Kidode, M., and Asada, H. 1973. *Comput. Graph. Image Process.* 2: 393–401.

Mowforth, P., Mayhew, J.E.W., and Frisby, J.P. 1981. Vergence eye movements made in response to spatial-frequency-filtered random-dot stereograms. *Perception* 10: 299–304.

Nalwa, V.S. 1987. Edge-detector resolution improvement by image interpolation. *IEEE Trans. Patt. Anal. Mach. Intell.* 9(3): 446–451.

Nalwa, V.S., and Binford, T.O. 1986. On detecting edges. *IEEE Trans. Patt. Anal. Mach. Intell.* 8: 699–714.

Nevatia, R. 1976. Depth measurement by motion stereo. *Comput. Graph. Image Process.* 5: 203–214.

Nielsen, K.R.K., and Poggio, T. 1983. Vertical image registration in stereopsis. Mass. Inst. of Technol., *Artif. Intell. Lab. Memo 743.*

Nishihara, H.K. 1984. Practical real-time imaging stereo matcher. *Optical Engineering* 23: 536–545.

Ohta, Y., and Kanade, T. 1985. Stereo by intra- and inter-line scanline search using dynamic programming. *IEEE Trans. Patt. Anal. Mach. Intell.* 7: 139–154.

Otto, G.P., and Chau, T.K.W. 1989. "Region-growing" algorithm for matching of terrain images. *Image Vision Comput.* 7(2): 83–94.

Pearson, D.E., and Robinson, J.A. 1985. Visual communication at very low data rates. *Proc. IEEE* 73: 795–812.

Poggio, G.F., and Poggio, T. 1984. The analysis of stereopsis. *Ann. Rev. Neuroscience* 7: 379–412.

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

Pratt, W.K. 1978. *Digital Image Processing*. John Wiley: New York.

Prazdny, K. 1985. The detection of binocular disparities. *Biological Cybernetics* 52: 93–99.

Quam, L.H. 1984. Hierarchical warp stereo. *Proc. DARPA Image Underst. Work.*, New Orleans, pp. 149–155.

Rosenfeld, A. 1979. Digital topology. *Amer. Math. Monthly* 86: 621–630.

Rourke, C.P., and Sanderson, B.J. 1982. *Introduction to Piecewise-Linear Topology.* Springer-Verlag: Berlin.

Rousseeuw, P.J., and Leroy, A.M. 1987. *Robust Regression and Outlier Detection.* John Wiley: New York.

Schunck, B.G. 1989. Image flow segmentation and estimation by constraint line clustering. *IEEE Trans. Patt. Anal. Mach. Intell.* 11: 1010–1027.

Scott, G.L. 1988. *Local and Global Interpretation of Moving Images.* Pitman: London.

Serra, J. 1982. *Image Analysis and Mathematical Morphology.* Academic Press: New York.

Shahraray, B., and Brown, M.K. 1988. Robust depth estimation from optical flow. *Proc. 2nd Intern. Conf. Comput. Vision*, Tampa, FL, pp. 641–650.

Sher, D.B. 1987a. A probabilistic approach to low-level vision. Ph.D. thesis, University of Rochester, Dept. Comp. Sci. Techn. Rep. 232.

Sher, D.B. 1987b. Tunable facet model likelihood generators for boundary pixel detection. *Proc. IEEE Comput. Soc. Work. Comput. Vision*, pp. 35–40.

Spacek, L.A. 1986. Edge detection and motion detection. *Image Vision Comput.* 4: 43–56.

Stewart, C.V., and Dyer, C.R. 1988. The trinocular general support algorithm: A three-camera stereo algorithm for overcoming binocular matching errors. *Proc. 2nd Intern. Conf. Comjput. Vision*, Tampa, FL, pp. 134–138.

Thomas, G.A. 1987. Television motion measurement for DATV and other applications. BBC Res. Dept. Report BBC RD 1987/11.

Trivedi, H.P., and Lloyd, S.A. 1985. The role of disparity gradient in stereo vision. *Perception* 14: 685–690.

Vilnrotter, F.M., Nevatia, R., and Price, K.E. 1986. Structural analysis of natural textures. *IEEE Trans. Patt. Anal. Mach. Intell.* 8: 76–89.

Voorhees, H., and Poggio, T. 1987. Detecting textons and texture boundaries in natural images. *Proc. 1st Intern. Conf. Comput. Vision*, London, pp. 250–258.

Watt, R.J., and Morgan, M.J. 1983. Mechanisms responsible for the assessment of visual location: Theory and evidence. *Vision Research* 23: 97–109.

Witkin, A., Terzopoulos, D., and Kass, M. 1987. Signal matching through scale space. *Intern. Comput. Vision* 1(2): 133–144.

Williams, D.R. 1988. Topography of the foveal cone mosaic in the living human eye. *Vision Research* 28(3): 433–454.

Yeshurun, Y., and Schwartz, E.L. 1989. Cepstral filtering on a columnar image architecture: A fast algorithm for binocular stereo segmentation. *IEEE Trans. Patt. Anal. Mach. Intell.* 11(7): 750–767.

Young, R.A. 1986. Locating industrial parts with subpixel accuracies. *Proc. Soc. Photo-Optical Instr. Eng.* 728: 2–9.

Zucker, S.W. 1985. Early orientation selection: Tangent fields and the dimensionality of their support. *Comput. Vision, Graph., Image Process.* 32: 74–103.

Appendix: Building Support Neighborhoods

As discussed in sections 2.4 and 5.2, support neighborhoods for match evaluations are required to be connected and to contain only *possible* cells. These requirements leave some flexibility in support-neighborhood shape. In an earlier matcher implementation [Fleck 1988b], support neighborhoods were chosen so as to be *star-convex*:

DEFINITION: *A neighborhood N is* star-convex *about a cell x if every cell in N can be joined to x by a connected, straight path lying entirely in N.*

For digitized images, any approximately straight path was considered acceptable. These neighborhoods were limited to a radius of 3 cells (paths 4 cells long) and the largest neighborhood meeting all these conditions was used.

The current implementation uses a simpler, separable sum, introduced by Fleck [1989]. The basic 1D sum at cell x adds weights along a connected, approximately straight path, extending at most 7 cells from x in each direction (i.e., total length at most 15 cells). The path used is the maximum-length path meeting these conditions. These sums are taken for all cells in the image, using paths in a consistent direction d.

The 2D sum in direction d is produced by cascading two 1D sums, taken in perpendicular directions. If all cells in the image could belong to support neighborhoods, the 2D sum at each cell would be the sum of weights in a square neighborhood of that cell. When some cells are excluded, the shape of the neighborhood can vary as d is changed. To produce an isotropic sum for the current implementation, 2D sums are taken in each of six directions and the maximum value used.

Near the edges of matching regions, 2D sums for cells in odd and even rows may differ slightly due to the pseudohexagonal tesselation (figure 17). In the full multiscale matcher, these slight differences can blossom into serious instabilities in regions where there are few features. To reduce this, the matching strengths in each row are averaged with those in adjacent rows, using weights [1, 2, 1].

Notes

[1]Except for a change to cyclopean output in figure 13, all results presented in this article were produced by the same algorithm, with the same settings of search and other parameters. In particular, nonzero vertical disparities were explored even for pairs that happened to be in near-perfect vertical alignment.

[2]The discussion in this article assumes near-parallel viewing geometry, so that changes in depth generate exclusively horizontal disparities when cameras are properly aligned. However, everything should apply, *mutatis mutandis*, to other viewing geometries.

[3]E.g., let depth be constant horizontally and have a slow slope vertically.

[4]The quotes emphasize the fact that such constraints do not refer to smoothness or continuity in the mathematical sense, which cannot be verified using finite-precision measurements, but to bounds on various derivatives of disparity. By contrast, the analysis in section 3 does use the usual mathematical definition of continuity.

[5]Roughly speaking, a bound on first differences of disparities. See [Burt & Julesz 1980a, b; Fleck 1988b; Pollard, Mayhew, & Frisby 1985; Trivedi & Lloyd 1985] for details.

[6]All synthetic stereo pairs shown in this article are generated with the full range of intensity values ([0, 255]) and then degraded to simulate real image conditions. A Gaussian (standard deviation 1 cell) was used to smooth each and then Gaussian noise (standard deviation 2 intensity units) was added.

[7]I.e., the match evaluations for the cells involved are set to zero.

[8]Since M_R is the evaluation for the best disparity at $(x + d_x, y + d_y)$, it must be at least as good as that of disparity $(-d_x, -d_y)$, i.e., M_L.

[9]Compare the psychophysical data reported by Bülthoff and Mallot [1988] and Mayhew and Frisby [1981].

[10]Note that the magnitude of the displacement must be reduced at coarser scales, to compensate for the sampling used in creating them.

[11]Bülthoff and Mallot's theoretical analysis is slightly flawed, as they have not accounted for smoothing in human or camera systems, which smears intensity values near the figure/background boundary. Empirically, although extra boundaries are sometimes present, they are quite close to the edges of the figure.

[12]Such as that found in the human eye, which is only roughly hexagonal [Williams 1988].

[13]This space is given the topological structure, metric, differential structure that it inherits as a subspace of the image plane.

[14]Replacing each intensity value I with $179(\log_{10}(I + 10) - 1)$ works well.

[15]Such as a shadow boundary falling exactly along a physical boundary.

[16]Either perspective or orthographic projection.

[17]In full technical generality, some conditions on cell shape are required to avoid pathological situations. See Fleck [1990a].

[18]The more complicated procedure described by Fleck [1988b] is equivalent.

[19]This is implemented by counting the ends of boundary segments on the edges of C. A vertex counts as one end if exactly one of the adjacent edges of C is in the boundaries. It counts as two ends if it is in the boundaries, but neither adjacent edge of C is.

[20]In fact, this isotopy can be constructed so as to be *smooth*, that is, having continuous derivatives of all orders. For these spaces, however, smoothness seems to convey little or no additional constraint.

[21]See note 1 regarding fancy models of camera geometry.

[22]Systematic deviations from the expected patterns of vertical disparities could be also used to update the model of camera misalignments.

[23]If either image dimension is smaller than 20 cells, or if the total strength in the two images is less than 35 times the combined area of both images, this model fit is considered unreliable and the program assumes zero translation and zero rotation.

[24]In an on-line system, this would clearly be accomplished by moving the eyes or cameras. In the current off-line implementation, one image is warped into the coordinate system of the other. Areas in this warped image that were not visible in the original image are filled in from the second (stationary) image.

[25]If fine disparity information is important (cf. section 5.2), rotation must interpolate values. Images are much easier to interpolate than edge maps.

[26]By "total strength" is meant the sum of matching strengths for all cells currently listed at this disparity in both halves of the disparity map.

[27]As the scale is made finer, the apparent size of a patch of surface increases in proportion to the area of the image. However, the expected number of distinct horizontal disparities in this patch increases in proportion to the square root of the image area. The expected number of cells per disparity (for a fixed theory of what range of disparities to expect in the scene) grows as the ratio of these two rates.

[28]If there is no previous scale or if the cells assigned to these promising disparities fail to cover at least 50% of the combined area of the two coarse-scale images, the disparity (0, 0) is also treated as promising.

[29]The measurements are for foveal vision and were originally expressed in minutes of arc. I converted them for clarity, assuming a center-to-center cell spacing of 0.5 minutes of arc [Williams 1988].

[30]To convert their results, I assumed that striped patterns become reliably visible when they have a period of 4 cells.

[31]All reported bounds have been normalized by edge-finder scale, so that they are appropriate to the density of edges reported by the edge finder used in the new matcher [Fleck 1989], a Canny [Canny 1983, 1986] operator with $e = 1$ cell and/or a Marr-Hildreth operator [Hildreth 1983; Marr & Hildreth 1980] with $w = 4$ cells.

[32]The edge-finder scale for the Drumheller and Poggio algorithm, Canny $\sigma = 1.5$, was supplied by Walter Gillett.

[33]Only very few algorithms are listed here because published descriptions of many algorithms do not contain enough information to calculate these search bounds.

[34]The digitization difference was produced by translating the scene before taking the second picture and realigning the images by hand.

[35]Compare the stereo evaluation technique used by Nishihara [1984].

[36]The method described below could be extended to refine vertical disparities as well, but I cannot think of any practical reason for wanting to do so.