Inference of Segmented Overlapping Surfaces from Binocular Stereo

Mi-Suen Lee, *Member*, *IEEE Computer Society*, Gérard Medioni, *Senior Member*, *IEEE*, and Philippos Mordohai, *Student Member*, *IEEE*

Abstract—We present an integrated approach to the derivation of scene descriptions from a pair of stereo images, where the steps of feature correspondence and surface reconstruction are addressed within the same framework. Special attention is given to the development of a methodology with general applicability. In order to handle the issues of noise, lack of image features, surface discontinuities, and regions visible in one image only, we adopt a tensor representation for the data and introduce a robust computational technique called tensor voting for information propagation. The key contributions of this paper are twofold: First, we introduce "saliency" instead of correlation scores as the criterion to determine the correctness of matches and the integration of feature matching and structure extraction. Second, our tensor representation and voting as a tool enables us to perform the complex computations associated with the formulation of the stereo problem in three dimensions at a reasonable computational cost. We illustrate the steps on an example, then provide results on both random dot stereograms and real stereo pairs, all processed with the same parameter set.

Index Terms—Binocular stereo, tensor voting, perceptual grouping, surface inference.

1 Introduction

Binocular stereo, a process performed effortlessly by humans with remarkable accuracy has not yet been duplicated at a satisfactory level by computer vision. Four decades ago, Julesz shed new light on binocular stereo vision, introducing random dot stereograms, and demonstrating that depth perception can occur even in the absence of monocular information [17]. An epitome of Julesz's work by the author himself can be found in [18]. The works of Marr and Poggio in 1979 [23] and Burt and Julesz in 1980 [5] are first attempts to define the problem and its fundamental constraints. Since then, progress has been made but the complete stereo problem remains unsolved. The derivation of scene descriptions from a pair of images encompasses two processes: The establishment of feature correspondences and the reconstruction of surfaces based on the depth measurements obtained by the previous process. The completion of these tasks based on exactly two images is encumbered with inherent difficulties. Lack of image features, measurement and quantization noise, surface discontinuities, and half occlusions hinder the perception of the scene by the computer.

Our goal is to address these issues in a general way and to propose an approach capable to deal with a wide variety of scenes. Instead of tackling the correspondence and the surface reconstruction problems sequentially, as most previous methods do, we adopt a unified framework for establishing correct correspondences and reconstructing the surfaces inferred from these correspondences. The integration of these two phases was first proposed by Hoff and Ahuja in 1989 [14] and was also used by Szeliski and Golland in 1998 [34]. The novelty of our approach stems from the use of a robust technique, tensor voting, that allows discontinuities and outliers to be handled properly when inferring surfaces and regions.

As demonstrated in many attempts to derive the "optimal" stereo matcher, local measurements, such as cross-correlation, provide reasonable hypotheses for feature correspondence, among which correct matches cluster to form visual structures, as illustrated in Fig. 1. To determine the correct matches, we apply tensor voting in large three-dimensional neighborhoods of the initial correspondences. Analyzing the results of tensor voting, we are able to handle the tasks of outlier detection, discontinuity localization, and surface interpolation. The method is noniterative, robust to initialization and thresholds, and has one critical-free parameter, the size of the neighborhood of a location in 3D space, also called the scale of the voting field.

The paper is organized as follows: We briefly describe previous work on stereo vision in Section 2, then discuss the issues that need to be addressed by a stereo system in Section 3. Section 4 contains a brief overview of tensor voting, and Section 5 describes its application to stereo. Experimental results are shown in Section 6, the strengths and weaknesses of our approach and its relation with respect to other stereo methods are discussed in Section 7, and conclusions are drawn in Section 8.

2 Previous Work

Previous attempts at solving the stereo problem have taken various paths to reach the desired output, be it a depth map,

- *M.-S. Lee is with Philips Research, Philips Electronics North America Corporation, 345 Scarborough Road, Briarcliff Manor, NY 10510. E-mail: Mi-Suen.Lee@Philips.com.*
- *G. Medioni and P. Mordohai are with the Institute for Robotics and Intelligent Systems, and with the Integrated Media Systems Center, University of Southern California, Los Angeles, CA 90089-0273. E-mail: {medioni, mordohai}@iris.usc.edu.*

Manuscript received 26 June 1999; revised 2 Mar. 2001; accepted 20 Aug. 2001.
Recommended for acceptance by S. Sarkar.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 110144.

0162-8828/02/$17.00 © 2002 IEEE

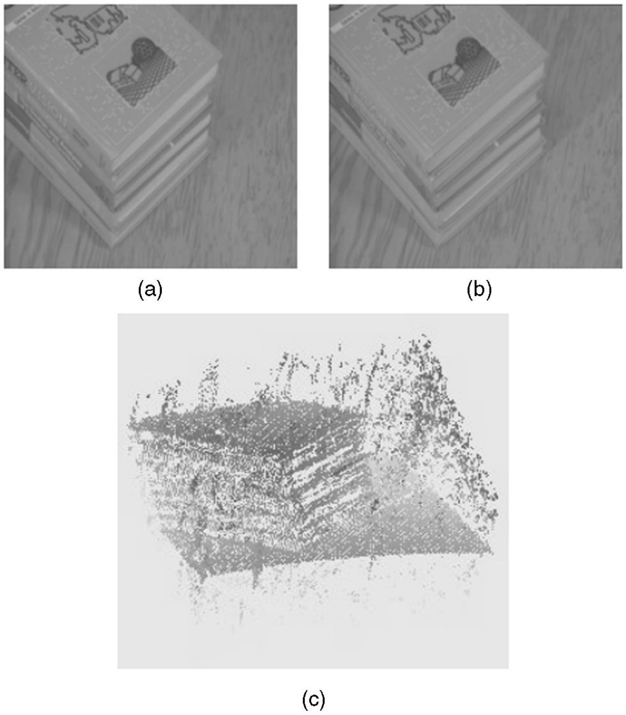

Fig. 1. Stereo pair and correlation-based correspondences. (a) Left image. (b) Right image. (c) Initial correspondences based on cross-correlation.

a set of points in 3D space, or a set of visible surfaces. The general paradigm they adhere to was established by Marr and Poggio [23], the uniqueness and continuity constraints were introduced a little later by Marr [24], the ordering constraint by Yuille and Poggio [39], and the gradient limit constraint by Burt and Julesz [5]. For early works on stereo, we refer readers to the reviews by Barnard and Fischler [1] and Dhond and Aggarwal [9]. This section discusses more recent publications in the field.

A significant class of stereo algorithms takes an approach based on the optimization of a cost function. Some of the most representative work in this category was published by Olsen [27], Okutomi and Kanade [28], Robert and Deriche [29], Fua [10], Wei et al. [37]. The main challenge with these methods is the selection of an objective function that enforces the smoothness constraint and, at the same time, handles discontinuities correctly and also degrades gracefully in the presence of noise. The proper treatment of all the issues that arise in binocular stereo usually results in very high-computational complexity. Roy and Cox [30] and, later, Ishikawa and Geiger [16] formulated the stereo problem as a maximum flow problem where the solution is a minimum cut of an undirected or a directed graph, respectively. This formulation reduces computational cost.

Stochastic formulations have also been attempted. Maximum a posteriori estimators were proposed by Geiger et al. [11], Belhumeur and Mumford [3], and Belhumeur [2], while a maximum-likelihood estimator was proposed by Cox et al. [8] and an approach based on Markov random fields by Boykov et al. [4]. These algorithms either demand explicit modeling of the surfaces, the discontinuities, and noise or assume that features and noise follow a normal distribution. These complex models may be suitable for a number of scenes which do not deviate much from their essential assumptions but may fail on other scenes.

Given the nature of the problem, robust techniques seem a good choice. Stewart proposed MINPRAN [33], a robust estimator that can infer surfaces, compatible with predetermined models, even in severe noise conditions. The shortcoming of MINPRAN lies in the need to know the type and number of surfaces we try to extract a priori and in its computational complexity for nonplanar surface models. Sara and Bajcsy [31], after making a key observation on the cause of the shift of occluding boundaries in disparity maps, propose robust matching operators that can handle considerable amounts of noise and occlusion. The generality of their method remains to be demonstrated.

An interesting approach is to integrate the two phases, namely, feature correspondence and surface reconstruction. It was first proposed by Hoff and Ahuja [14] in 1989. According to their algorithm, we can use matched features to obtain disparity estimations from which we can reconstruct surfaces. This surface information can be used to validate the matches. A similar scheme was introduced by Szeliski and Golland [34] where they simultaneously recover depth, color, and opacity. Initial surface, color, and opacity information is reprojected to the images for verification.

Finally, there is a different class of algorithms that operate in three-dimensional space. It includes the algorithms proposed by Collins [7] and Seitz and Dyer [32]. Their characteristic is the use of the projection rays from the images to the scene and their relations in space to infer possible locations of world points.

A combination of the latter techniques that is related to our approach of the stereo problem was proposed by Chen and Medioni [6]. The processing is performed in the three-dimensional disparity space, where correct matches are identified starting from locations where unique maxima of the cross-correlation of intensity values occur. The coherence principle and the gradient limit constraint are used to guide the extraction of disparity assignments in a volume that contains cross-correlation values for every potential disparity of the image pixels. It produces very good results as long as the input images are of high quality and noise corruption remains low.

3 Design Issues and Constraints

In this section, we will discuss the main challenges associated with stereo vision and the choices that were made to address them. Then, the most common constraints will be presented along with the way they are enforced in our method.

3.1 Design Issues

The major factors that limit the performance of stereo algorithms include depth and orientation discontinuities, noise, incompatibility of the scenes with the selected models, and inadequate internal representation of the data. To avoid these shortcomings, we opt for a representation that can handle all possible roles of a location in a binocular stereo configuration, and is robust to noise. At the same time, we refrain from making severely restrictive assumptions about the geometry of the scene or the properties of the objects depicted. More specifically, we use the tensor representation, described in the following section, which can efficiently represent surface orientation, discontinuities

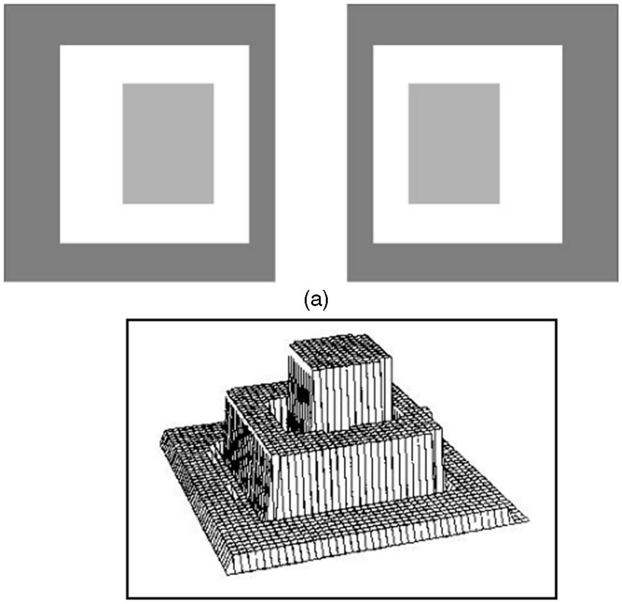

Fig. 2. The "wedding cake" interpretation of overlapping planes. (a) Input images. (b) "Wedding cake" output.

and outliers, and we perform all the processing in 3D instead of 2 1/2D [20].

The 2 1/2D sketch was introduced by Marr [24] and is very popular among stereo researchers. The weakness of this representation lies in the fact that it is view-centered and viewpoint dependent elements are unnecessarily introduced into the framework. Even though this might be acceptable when dealing with a single image, it is detrimental when dealing with a stereo pair and unsuitable for extension to more views. We illustrate the problem using the image pair seen in Fig. 2, which is perceived by humans as three overlapping planes. A common solution from many existing stereo algorithms, shown in Fig. 2, has the form of a "wedding cake" where only one depth value is associated with every pixel in the reference image. However, as asserted by human perception of the stereo pair, this is not what the output should be. Discontinuities in depth do not correspond to any physical property of the 3D objects. Human perception of the scene does not indicate a "wedding cake" structure nor that the lower planes have holes below the upper layers. Instead, humans perceive three overlapping planes without holes or with an uncertainty as to what lies beneath the upper layers. A single depth map, the main form of 2 1/2D representation, is incapable of conveying the correct interpretation of the scene, therefore, a 3D representation in terms of layers is appropriate.

The novelty of our method comes from the fact that we use "saliency" instead of cross-correlation to determine the correctness of matches. Instead of making the decisions at the matching stage based on cross-correlation, we delay them until saliency information is available. From our perspective, high cross-correlation values between image intensities are indications of potential matches, but not very reliable as a criterion for resolving the correctness of matches. We propose the use of saliency for that purpose. By saliency, we mean the likelihood that a location belongs to a perceptual structure, which could be a surface, a curve, or a junction. As demonstrated in the remainder of the paper, our data representation and communication scheme enable us to determine the saliency of data items. In this approach, locations that produce erroneous high cross-correlation values can be detected and removed. Since they are not supported by other data, they will have low saliency and will be identified as outliers. Had we opted to make decisions immediately after the initial computation of cross-correlation values, we would have been misled. This a common failure of correlation-based matching.

A consequence of the use of saliency is the integration of feature matching with surface and curve extraction. The approach is similar to the one proposed in [14]. As we compute the surface or curve saliency of a location, we are able to decide whether the location is a correct match. If it has high local support, in the sense that it is compatible with its neighbors with respect to surface normal or curve orientation, then it is more likely to belong to a surface or curve of the scene. The lack of local support, which results in low saliency, indicates that the location under examination is most likely an outlier and does not belong to some underlying structure.

Since the objective is to obtain a scene description from the stereo pair, the output of a stereo algorithm should be surfaces and curves. A cloud of 3D points could be an intermediate stage of the process but certainly not the goal. Objects do not consist of isolated points but are three-dimensional volumes whose bounding surfaces are visually perceived. Therefore, a description in terms of surfaces, surface boundaries, curves, and curve junctions should be the desired output. Combining tensor voting with a marching process, the proposed framework extracts surfaces in the form of a triangulated mesh in a way similar to [22] and [35].

3.2 Constraints

Taking into consideration that the problem at hand is ill-posed, several constraints that should be imposed on the solution have been proposed. Besides the epipolar constraint, the most widely used among these constraints include the continuity constraint which expresses the fact that "matter is cohesive," the uniqueness constraint, and the ordering constraint [24], [39].

We apply the epipolar constraint in the initial matching phase. Corresponding features are assumed to lie on corresponding epipolar lines, therefore, the search for potential matches can be limited to a one-dimensional problem in image space. After the initial matches have been generated using a simple cross-correlation based technique, all processing is performed in three dimensions.

The continuity constraint states that objects tend to be smooth and continuous, therefore, the scenes they are part of exhibit the same properties. The problem with the continuity constraint is that it applies "almost everywhere," but not at discontinuities. The inherent difficulties in the matching process led researchers to substitute erroneous or missing matches with estimates based on their neighbors, usually along the same epipolar line, a practice that is effective in areas where surfaces are smooth, but has to be avoided close to discontinuities. We impose the smoothness constraint in 3D via tensor voting to overcome predicaments that arise from its imposition in one or two dimensions.

The uniqueness constraint states that a feature on one of the images can match at most one feature on the other image. The novelty of the proposed method is that we do not enforce the uniqueness constraint immediately at the matching stage. Instead, we allow all potential matches as inputs to the tensor voting process, the results of which point out the outliers. We then locally enforce the uniqueness constraint to select the best correspondences. Note that this local constraint does not translate into uniqueness on the reconstructed surfaces, thus allowing transparency and the continuation of occluded surfaces behind the occluding ones. The latter is a desirable phenomenon, as it is consistent with human perception and is often explicitly ignored by many stereo algorithms.

Finally, the ordering constraint states that, along any epipolar line, the order of two or more features should be the same in two views. While it holds for scenes composed of a single surface, the ordering constraint is not applicable to scenes with either small or thin objects and containing transparent or acutely concave surfaces. We implicitly impose this constraint locally only.

4 Tensor Representation and Voting

The core of our framework is tensor representation and tensor voting. The use of tensors in the field of signal processing and computer vision was introduced by Knutsson [19] and Westin [38]. Medioni et al. [25] formulated a methodology for inferring perceptual structures from sparse noisy data under which tensors communicate with their neighborhood sending and receiving information via a voting process. This section presents the basic theory of tensor representation and voting and the reasons they are suitable for the problem at hand. We refer interested readers to the Appendix and to the book [25] for details.

4.1 Tensor Representation

The use of tensors as a means of data representation stems from the observation that discontinuities occur at locations where multiple structures such as curves, surfaces, or regions intersect. For instance, a surface orientation discontinuity, which has the form of a curve, occurs at the intersection between two surfaces. In other words, whereas there is only one normal orientation associated with a location on a smooth surface patch there are multiple orientations associated with any location on a discontinuity. Hence, the desirable data representation is one that can encode multiple orientations and the multiple roles a location in a scene might have, as well as their certainty.

The second-order symmetric tensor possesses precisely this property. It has the form of a 3x3 symmetric, positive semidefinite matrix or, equivalently, of a three-dimensional ellipsoid. The *shape* of the tensor is used to encode certainty of orientation and its *size* to encode feature saliency. The decomposition into its eigenvectors and eigenvalues can provide information as to what the tensor represents (Fig. 3). A tensor that only has one nonzero eigenvalue represents perfect certainty of surface orientation with normal parallel to the corresponding eigenvector, while the magnitude of the eigenvalue indicates the saliency of the surface. A tensor with two equal nonzero eigenvalues represents any possible surface orientation perpendicular to the eigenvector corresponding to the zero eigenvalue or a certain curve orientation parallel to this vector. A tensor with three equal eigenvalues represents perfect uncertainty of all orientations and the magnitude of the eigenvalues indicates whether the location is an intersection of multiple features or an outlier. These three types of tensors are referred to as *stick*, *plate,* and *ball* tensors, respectively. Any second order symmetric tensor can be decomposed into a linear combination of these three basis components.

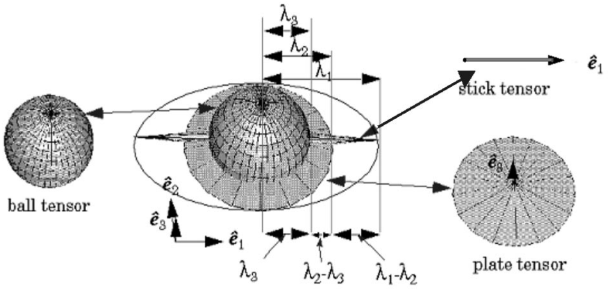

Fig. 3. A second order symmetric tensor.

4.2 Tensor Voting

The strength of our approach resides in the way data points communicate with each other through tensor voting. It is a process similar to the Hough transform [15] in the sense that we let the solution emerge from the data after measurements of compatibility among data items. It differs in the fact that there is no need to specify beforehand the parametric configuration we are looking for and the computational complexity is independent of the dimensionality of the structures to be inferred.

In order to confirm the compatibility of our data or to reach the conclusion that some data items are outliers, we need to communicate local information among neighboring locations. This is accomplished by tensor voting. Every location, whose orientation and saliency information have been encoded in a tensor, can cast votes to neighboring locations. The decomposition of a second-order symmetric tensor into the basis components allows for the execution of the vote in three steps, one for each component, with predefined voting fields. This eliminates the need to calculate a voting field for every tensor we encounter and allows us to perform the voting process using look-up tables, thus significantly reducing the number of operations.

Conveniently, the voting fields are second-order symmetric tensor fields that produce second order symmetric tensors as votes at the receiving locations. Their orientation is determined by the relative position of the vote-casting and the vote-receiving location. Their saliency, that is the strength of their influence, is determined by the voter's saliency attenuated with respect to the distance and curvature between the voting and receiving location. These choices were made because the effect of the vote should obviously decrease with distance and it also decreases with curvature since a planar patch is preferable to a curved surface if both options are possible.

The only parameter in the entire tensor voting process is the rate of attenuation of the voting field's strength. It is implemented as a Gaussian function, whose spread is the free parameter, and is equivalent to the size of the voting neighborhood, since the latter is essentially the area around a location, where the votes cast by it are not negligible. The framework has low sensitivity to this parameter, and slight alterations of its value result in negligible changes of the outputs.

4.3 Vote Interpretation

We have developed a unified framework for data representation and communication. The next step is to define a procedure to analyze the results of this communication. This task is assisted by another convenient property of the tensor representation. The accumulation of votes at a location is implemented as a tensor addition, or a summation of 3x3 matrices. After voting is completed, the resulting matrix, which also incorporates any prior information of the location, can be decomposed into its eigensystem. We can determine a location's role in the scene by analyzing the eigensystem after the vote.

The total saliency of the location as an inlier is determined from the magnitude of its largest eigenvalue. A very small value indicates an isolated point without any support from its neighbors. On the contrary, a large value indicates a location that received a large number of votes and, therefore, is likely to belong to an underlying structure. *Surface saliency* is encoded by the difference of the two largest eigenvalues (see Appendix) and its orientation is the eigenvector corresponding to the largest eigenvalue. Similarly, *curve saliency* is encoded in the difference between the second and third eigenvalues. The tangent to this curve is the eigenvector corresponding to the smallest eigenvalue. Finally, curve junction saliency is encoded in the smallest eigenvalue. It is not accompanied by any preference in orientation.

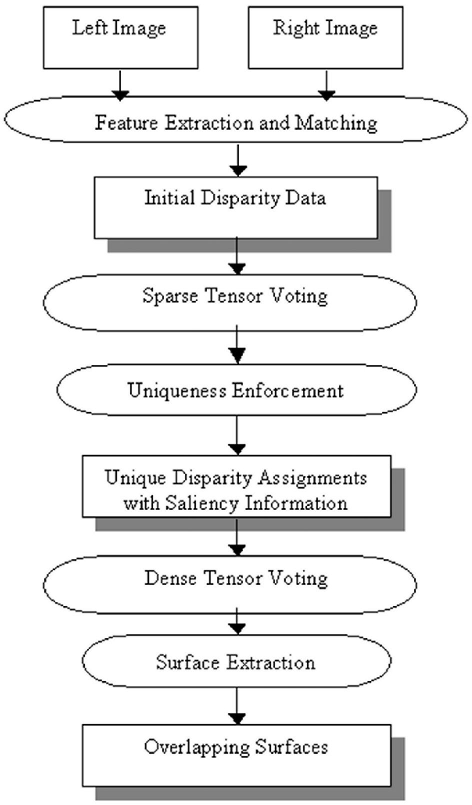

Fig. 4. Flowchart of our approach.

5 Tensor Voting and Stereo

After reviewing the design choices that were made and the theoretical background of our method, this section describes the implementation of the tensor voting framework to stereo. The various phases of the procedure are outlined in the flowchart of Fig. 4. We begin by extracting features on each image of the stereo pair and then establishing potential correspondences based on cross-correlation. We proceed by estimating the saliency of these correspondences using tensor voting and removing outliers from the data set. The next step is to compute saliency in the entire disparity space and extract salient structures. Optionally, in the case of real images, we can use monocular edge information to aid the process.

5.1 Initial Feature Correspondence

The algorithm accepts as input a pair of stereo images of a static scene. If calibration information is not available, we use a method such as the one proposed by Zhang et al. [40] to obtain the epipolar geometry. In the remainder of this paper, we will assume, without loss of generality, that the images have already been rectified. The first preprocessing step is the extraction of points of interest or features that can be matched. Intensity variance is used as a criterion for the selection of a pixel as a feature suitable for matching. The reason behind this is to avoid totally textureless areas where all matches are ambiguous. In practice, for most typical scenes, a large percentage of the pixels, possibly exceeding 90 percent, are retained for the next step. In case of random dot stereograms, all the dots are retained. Intensity variance is defined as

$$\sigma(u,v) = \sum_{(i,j)\in([u-win,u+win],[v-win,v+win])} (I(i,j) - \mu_I(u,v))^2, \tag{1}$$

where $2 * win + 1$ is the size of the search window, and $\mu_I(u,v)$ is the mean intensity in that window. We retain all pixels whose intensity variance is not so close to zero as to be considered insignificant.

Feature extraction is the only monocular processing step in the entire algorithm. The next step is feature matching. We use normalized cross-correlation of intensity values to match features across the two views. To overcome the inherent problems of matching, instead of devising a sophisticated matching scheme, which we feel is not the remedy to the problem, we allow *all* potential matches to be used as input to the next stage. We define potential matches as the ones with cross-correlation values close to the maximum value for the feature under examination. Since cross-correlation can be misleading (see [31] for a very interesting analysis), we let the disambiguation of matches take place at a later stage. It must be pointed out that stricter thresholds on feature selection and matching, resulting in fewer, but possibly better matches, do not affect the results significantly since our framework works equally well for sparse and dense initial data even in the presence of significant outlier noise.

5.2 Correspondence Saliency Estimation

After the initial potential matches have been recorded, they are encoded in the tensor representation described earlier. The encoding and all subsequent processing are performed in a three-dimensional space, either disparity space or real-world space, in case calibration information is available. Our

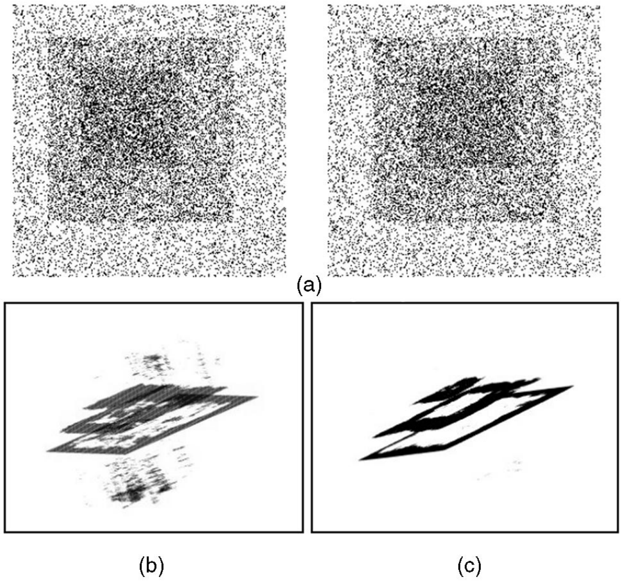

Fig. 5. Random dot stereogram with three overlapping layers. (a) Random dot stereogram input. (b) Initial correspondences. (c) Correspondences after tensor voting and uniqueness enforcement.

goal at this point is to determine which are the correct matches and to discard erroneous matches and outliers. We accomplish this with a sparse tensor vote where every location casts votes to all active sites in its neighborhood. The resulting tensors at each location are decomposed into the three basis components and the role of each point is determined. Points with high surface saliency are more likely to belong to smooth surface patches, points with low surface saliency but high curve saliency are more likely to belong to surface discontinuities, while points with no orientation preference are probably outliers. More specifically, if:

$$\lambda_{i,1} - \lambda_{i,2} > thres * \max_{j}\{\lambda_{j,1} - \lambda_{j,2}\} \tag{2}$$

location i is considered a good candidate to belong to a surface and is retained. The difference between the largest and second largest eigenvalue of the tensor at location i, $\lambda_{\mathrm{i},1}$, and $\lambda_{\mathrm{i},2}$, must be within a percentage of the maximum such difference among all locations. The threshold *thres* has no significant effect on the output if kept within a reasonable range. The insensitivity to this parameter is due to the fact that locations that are marginally labeled inliers or outliers do not contribute much to structure extraction. Outliers mistakenly kept do not have strong surface saliency to affect the extracted surfaces, while missing points are filled in when information is propagated.

Points that are labeled outliers on account of the fact that they do not display any feature saliency are removed from the data set and their influence is removed by performing another vote. Each data item about to be removed casts votes in its neighborhood that are subtracted from the tensor of the receiving locations instead of being added.

After the collection of local support information at each location, the local uniqueness constraint can be enforced with respect to saliency. Each line of sight is examined and only the most salient data item is retained. Locations that had originally higher cross-correlation values in error are easily detected at this stage since their presence is not supported by the neighboring locations and their saliency is low. Correct matches that seemed not to be the first choice for a feature in the beginning may survive due to their compatibility with their neighbors in the formation of a salient structure. Note that erroneous removal of a correct point is not catastrophic since it will most likely be interpolated at the next stage. The input pair of a random dot stereogram is shown in Fig. 5a, the initial matches produced by cross-correlation in Fig. 5b, and the matches retained after tensor voting and uniqueness enforcement in Fig. 5c.

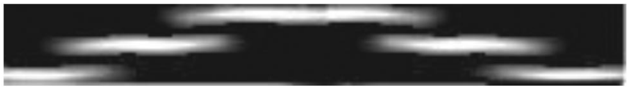

Fig. 6. Cut of the surface saliency volume.

5.3 Salient Structure Extraction

The desired output is not a cloud of points with associated depth measurements but a description of the scene in terms of surfaces and surface discontinuities. Therefore, starting from the sparse tensors, we must derive a dense representation of the space from which to extract structure. We perform a dense tensor vote in which the ball components do not contribute and compute a saliency tensor at every voxel in the three-dimensional space. Once we have the necessary saliency information, salient surfaces, surface junctions, and curve junctions are extracted by a nonmaximal suppression process [35] based on the original Marching Cubes algorithm proposed by Lorensen and Cline [22].

In order to reduce computational cost, the calculation of saliency tensors at locations with no prior information and structure extraction are integrated and performed as a marching process. Beginning from seeds, locations with highest saliency, we perform a dense vote only towards the directions dictated by the orientation of the features. Surfaces are extracted with subvoxel accuracy, as the zero-crossings of the first derivative of surface saliency. A slice of the inferred surface saliency in the volume can be seen in Fig. 6. Brighter gray-scale values indicate higher saliency. Locations with high surface saliency are selected as seeds for surface extraction, while locations with high curve saliency are selected as seeds for curve extraction. The marching direction in the former case is perpendicular to the surface normal, the eigenvector of the saliency tensor corresponding to the largest eigenvalue, while, in the latter case, the marching direction is along the tangent to the curve, the eigenvector corresponding to the smallest eigenvalue. Curve junction saliency is a strictly local property and is not communicated in the extraction process. The surfaces extracted from the random dot stereogram of Fig. 5 are shown in Fig. 7. They consist of overlapping layers parallel to the image plane.

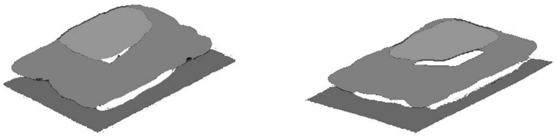

Fig. 7. Rotated views of the extracted surfaces.

As mentioned in the previous section, the extraction process degrades very gracefully in the presence of noise. Since outliers are unlikely to accidentally form structured arrangements, their tensors are dominated by the ball component and given that the ball component does not participate in the dense tensor vote their effects are negligible. Conversely, correct points that may have not been detected by the initial matching process or have been mistakenly removed may be filled in during the dense vote.

5.4 Computational Complexity

The establishment of initial correspondences is of linear complexity, more specifically, $O(dn)$, where n is the number of features and d is the disparity range we examine. The sparse vote is of $O(Cn)$ complexity, where C is the average number of neighbors of a data item. In the worst case, this can lead to $O(n^2)$, but that is a clear indication of an incorrect setting of the size of the neighborhood. For most practical cases, C is a small fraction of the data set. The enforcement of the uniqueness constraint is a linear process since it is done along the lines of sight which, in general, are less than the data items and at most are equal to them.

The most time consuming stage is the extraction process. The inherent output sensitivity of the Marching Cubes algorithm [22], [35] causes our method to be output sensitive as well. The marching process examines voxels that contain some structure to be extracted with high accuracy, in the order of 98 percent. The cost associated with the calculation of the saliency tensor at these locations is reduced by storing previously computed votes in a balanced binary tree, which is purged when its size becomes a hindrance, rather than an acceleration to the algorithm. Correct use of the balanced binary tree can ensure that only a negligible number of tensors have to be computed twice, saving a significant amount of floating point operations. Given that each voxel is affected by a constant number of its neighbors, which will be denoted by K and is a function of the attenuation factor of the voting field, and that the binary tree contains at most M entries, an estimate of the computational complexity of this stage is $O(sKlogM)$, where s is the number of voxels containing features. Since we cannot alter the number of voxels containing features without loss of accuracy in the output and we need a large enough neighborhood for correct results, a more efficient way of selecting the size of the binary tree should be devised.

Space complexity is $O(n)$ in the first stages, and $O(n + s + M)$ in the extraction stage. The relationship between the number of input locations and the number of voxels containing features depend on the scene and the quantization of the 3D volume where the marching process takes place.

Since all processing is local, a parallel implementation is feasible. Feature matching can be done in parallel along epipolar lines, sparse tensor voting can be performed locally in segments of the data set, uniqueness enforcement can be carried out in parallel along lines of sight, and the extraction process can also be performed in parallel, starting from various seeds simultaneously.

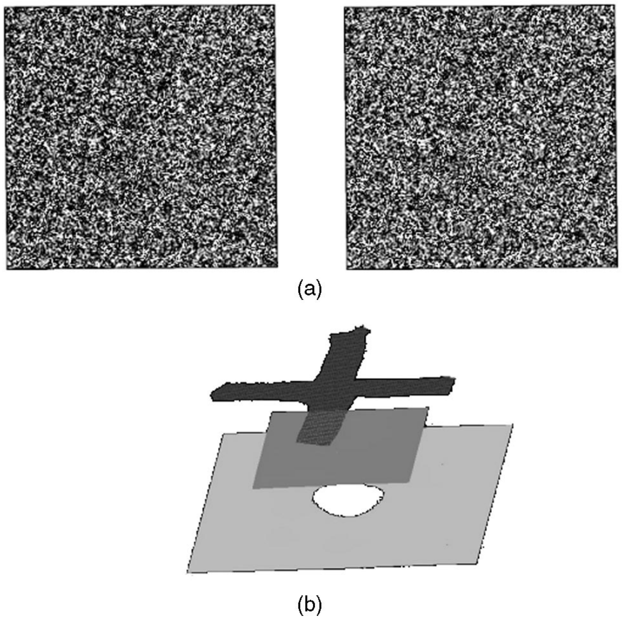

Fig. 8. Random dot stereogram of a cross and two overlapping planes.(a) Input images. (b) Output surfaces.

6 Experimental Results

In this section, we present experimental results from random dot stereograms, as well as real stereo pairs. We applied our algorithm to diverse scenes, keeping the size of the voting neighborhood constant at 51 x 51 x 35.

6.1 Synthetic Data

The example used to illustrate the steps of our method (Figs. 5, 6, and 7) demonstrates the capability of our framework to handle multiple layers. Fig. 8 shows another example of a random dot stereogram depicting a cross floating over two overlapping planes. This a classical example, adapted from Nalwa's book [26]. It is also properly handled without layer initialization or alterations of the parameters. Note that the boundaries of the cross are accurate and we even detect the corners explicitly.

The random dot stereogram shown in Fig. 9 is a special case, as it involves transparency. Every dot on the right image has exactly two matches in the left image. This configuration was introduced by Julesz [18] and gives rise to two layers at different disparity levels. Due to the symmetry between the two layers, they are equivalent in terms of saliency and they are both extracted as seen in Fig. 9d. This example illustrates our point that even though we enforce uniqueness at the local level, the output still contains overlapping layers.

The final example on synthetic data is shown in Fig. 10. The input is two views of a synthetic cube and the output surfaces can be seen in Fig. 10b. This example illustrates the capability of our approach to explicitly handle sharp orientation discontinuities in synthetic datasets and extract curves along these discontinuities.

6.2 Real Data

In this section, we present results on stereo pairs of real scenes. Fig. 11 depicts the steps of processing a stereo pair of the famous "Renault" part, used in many stereo experiments.

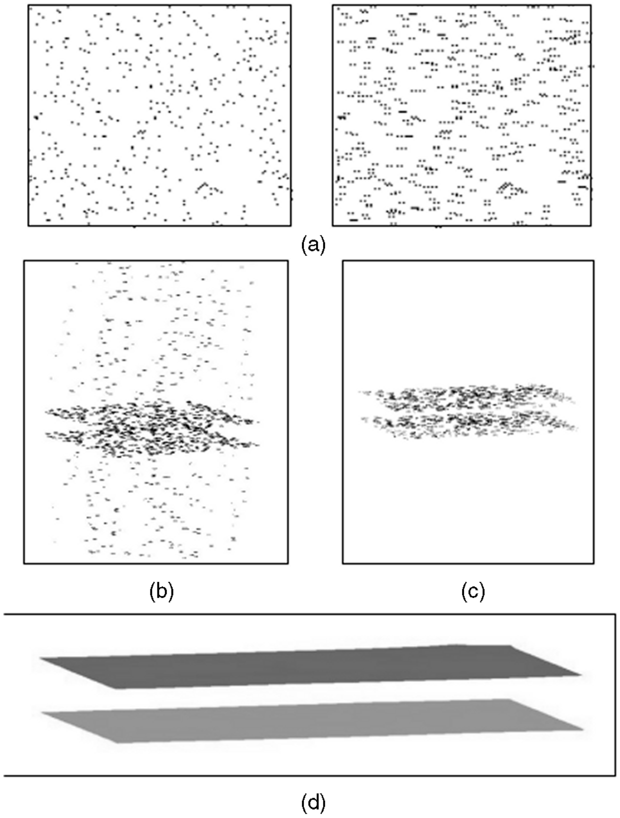

Fig. 9. Transparent planes random dot stereogram. (a) Random dot stereogram input. (b) Initial correspondences. (c) Unique disparity assignments. (d) Extracted surfaces.

The input images can be seen in Fig. 11a, the initial correspondences in Fig. 11b, the output of the sparse tensor vote and uniqueness enforcement in Fig. 11c, the extracted surfaces in Fig. 11d, and texture-mapped views of the inferred surfaces in Fig. 11e. Note that parts of the occluded background have been filled in near the occluded edges.

Fig. 12 is an example based on aerial images of a sports arena. The major difficulty in this case is the fact that the arena itself and the surrounding environment do not contain significant texture information. Nevertheless, we are able to extract surfaces based on the sparse initial matches. Note that calibration information was not available and, as a result, the extracted surfaces are displayed in scaled disparity space.

Fig. 13 depicts a stereo pair from a different domain. Surfaces are extracted from an uncalibrated stereo pair of a face. We are able to represent the complex surface of the human face without imposing additional constraints such as locally planar or spherical patches. The background plane, that has disparity zero, has been removed for clarity.

Finally, Fig. 14 is also a classical stereo pair. It consists of a number of approximately fronto-parallel layers. A depth map of the reconstructed surfaces is shown in Fig. 14b. A total of six layers are extracted. Lighter shades of gray indicate layers that are further from the viewer.

6.3 Region Trimming

A final, optional step after surface extraction, still at an experimental stage, is region trimming. It can be applied to correct region boundaries in case monocular edge information is available. Obviously, it is not applicable to random dot stereograms. As pointed out in [31], surface boundaries are often shifted by intensity-based stereo matching. This affects our results only in the case of depth discontinuities since orientation discontinuities are properly handled by tensor voting. When depth discontinuities occur there is a noticeable overextension of the surfaces. To illustrate the symptoms in the book scene, we compute the product of the input images and the corresponding disparity maps (Fig. 15). One can identify background regions that appear brighter than the rest of the background and as bright as the foreground and which have been assigned erroneous disparity values.

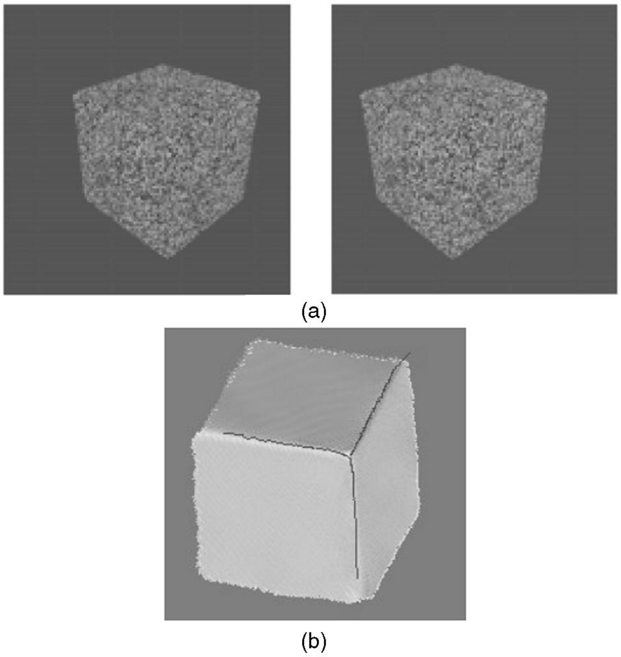

Fig. 10. Stereo pair of a synthetic cube. (a) Input images. (b) Output surfaces.

We propose to treat surface overextension by inferring the correct boundaries and removing the overextensions. For instance, in [16], other researchers have used monocular edge information to treat discontinuities. Our approach is entirely different. We do not use edges to reduce discontinuity penalties, a characteristically two-dimensional process, but, instead, project those edges in the three-dimensional space and use them to guide the true surface boundary detection. Since edge detectors often fail to extract the entire occluding boundary, the detected edgels need to be linked. This is a curve inference problem addressed in [21]. Tensor voting and analysis of the results with respect to curve saliency allows the extraction of complete boundaries.

The last step is the identification of which of the two segments of the surface, as segmented by the boundary, should be retained and which should be removed. A vector voting scheme can be employed to measure the "polarity" of boundary points [21], that is, the side on which the majority of their neighbors lies. It can be easily implemented under the tensor voting philosophy, with the sole

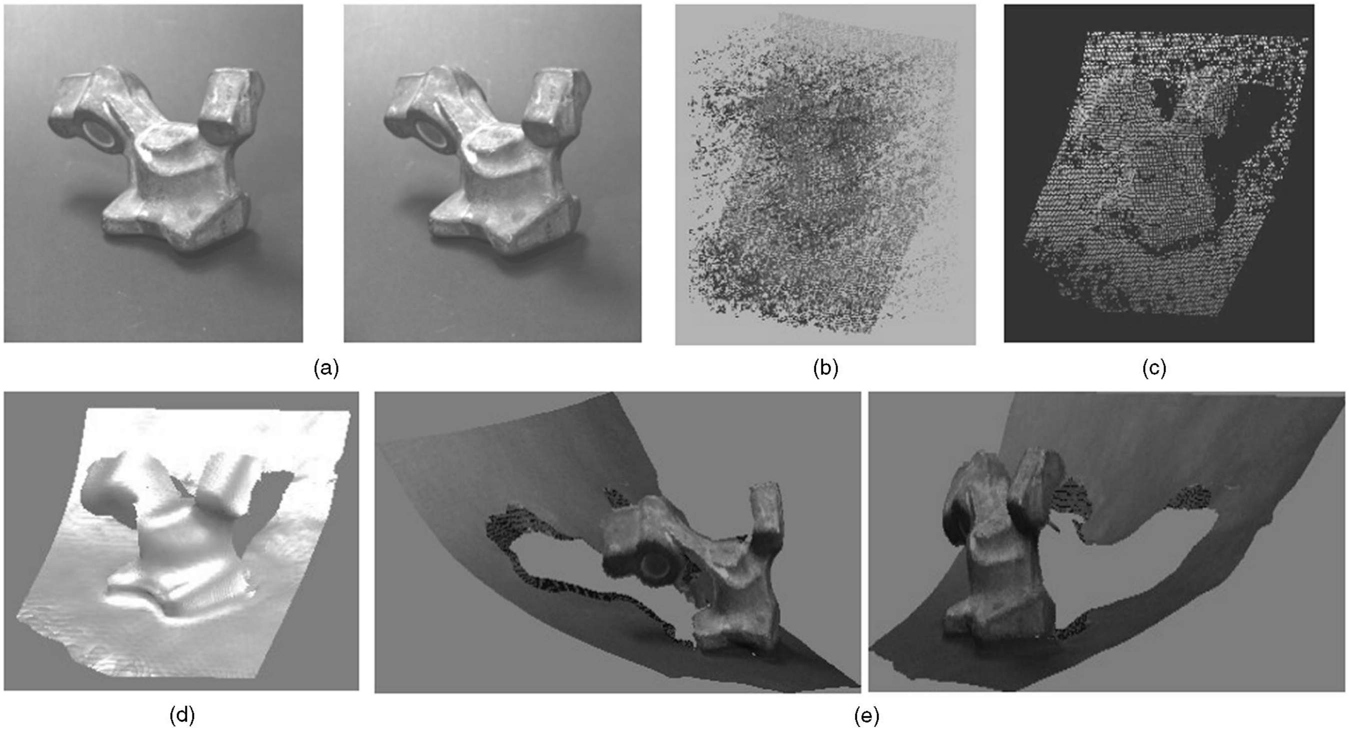

Fig. 11. Renault part. (a) Input images. (b) Initial correspondences. (c) Unique disparity assignments. (d) Inferred surface in disparity space. (e) Texture-mapped views.

difference being that instead of performing tensor additions to accumulate the votes, vector additions are performed, thus maintaining direction as well as orientation. A simple illustration can be seen in Fig. 16. After we have established on which side the inliers lie, the overextended surfaces are removed, as seen in Fig. 17.

7 Discussion and Future Work

We have presented a framework that is able to deal with the binocular stereo problem 3D space. The necessity of processing in three dimensions, as opposed to one or two, was justified in Section 3. This is difficult for stereo techniques based on optimization ([2], [3], [8], [11], [27], [28], [29], [31], [37]) due to the increase in computational cost associated with the increase in dimensionality. Even the maximum flow formulation ([16], [30]) is essentially two-dimensional since there can only be one depth value associated with an image pixel. These methods, by design, are incapable of handling transparency, and their outputs

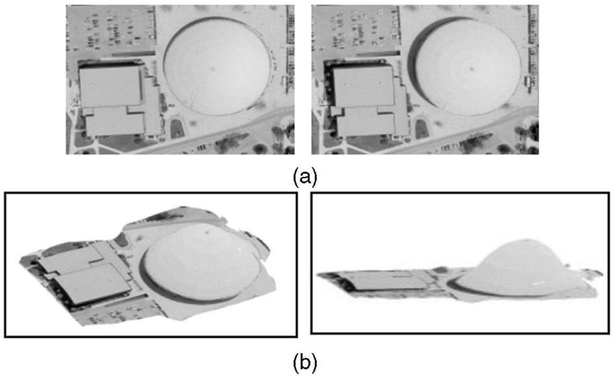

Fig. 12. The arena. (a) Input images. (b) Views of extracted surfaces.

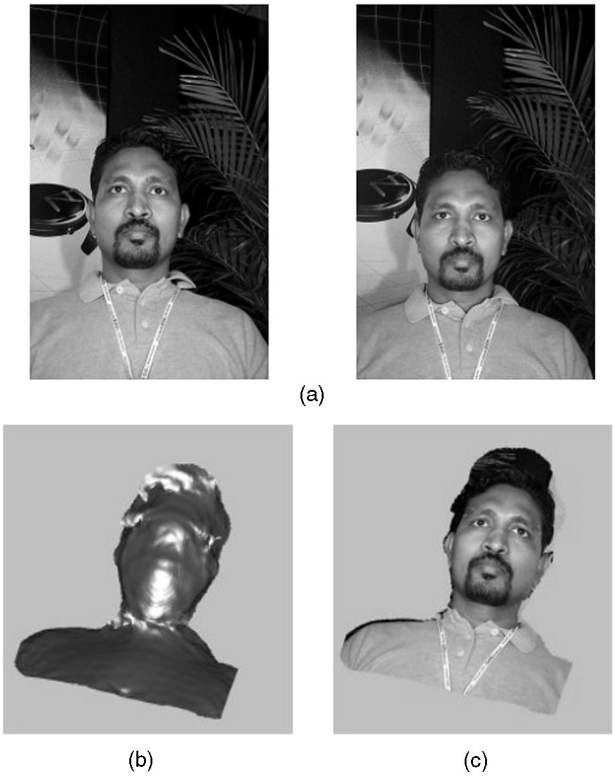

Fig. 13. A face. (a) Input images. (b) Extracted surface. (c) Texture-mapped surface.

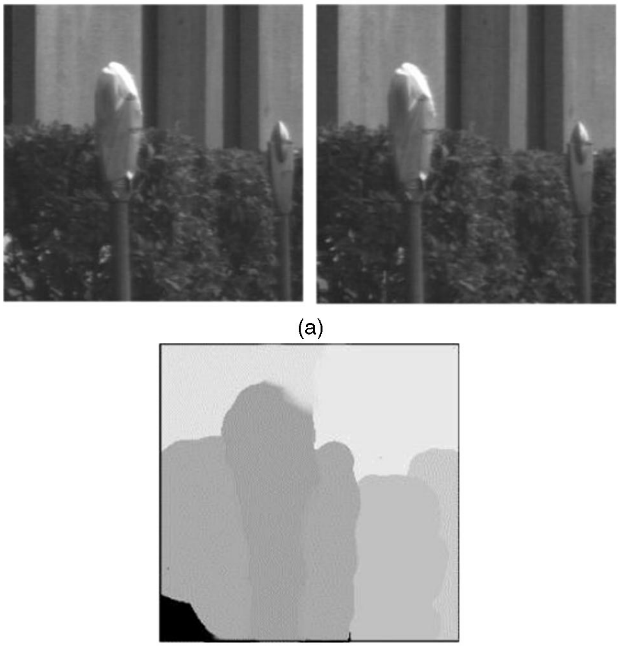

Fig. 14. The "parking meters" stereo pair. (a) Input images. (b) Depth map.

are view-dependent since they allocate a single depth value to each ray of sight of a reference camera, either one of the two real cameras, or a virtual camera.

In addition to the capability to handle transparency, three-dimensional processing plays an important role in the enforcement of the continuity constraint. Unless we operate in three dimensions, there is the unwanted effect that two locations adjacent in one of the images but very distant in the three-dimensional world interact significantly with each other. This occurs at depth discontinuities and is definitely an undesirable phenomenon since there should not be any propagation of information between locations that belong to different objects or distinct parts of the same object. The tensor voting framework enables the proper enforcement of the continuity constraint locally, in the actual neighborhood of the locations in real or disparity space.

A key contribution is the use of saliency instead of cross-correlation as the criterion for determining correct matches. The shortcomings of cross-correlation based matching are well known and are the cause of failures in stereo systems. We claim that surface saliency is a more relevant quantity when one is interested in extracting surfaces from a stereo pair. Similarly, curve saliency is more relevant during curve extraction. The delay in the enforcement of the uniqueness constraint allows us to examine the local support a location receives, before deciding whether it is an inlier or an outlier. Calculation of saliency can disambiguate matches, remove outliers and interpolate surfaces and curves in case of missing features.

Related to the above is the integration of feature matching with structure extraction. Unlike conventional methods, where the two tasks are performed sequentially, that is surface extraction is performed after the "correct" matches have been established, we use saliency to establish the correct matches, and use these matches to reconstruct the underlying structure of the scene. The matches that are considered inliers, are used to guide and constrain the extraction process and, hence, their correctness can be judged by the type of surface they produce.

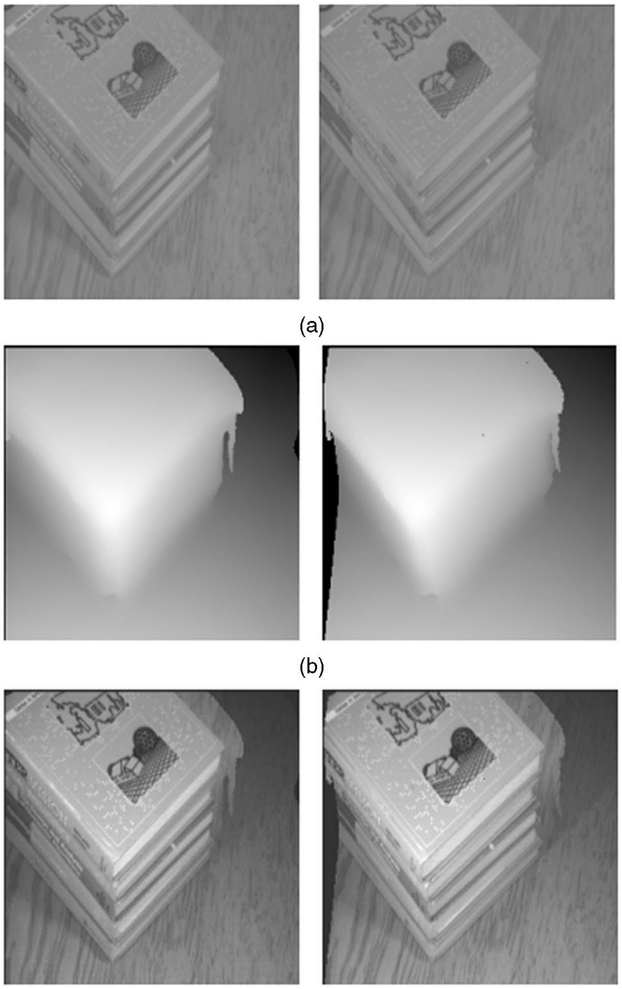

Fig. 15. Illustration of overextended surfaces. (a) Input images. (b) Disparity after densification. (c) Intensity-disparity product.

The presence of noise is something that cannot be neglected in any stereo system. Noise is introduced by the imaging devices, imperfect calibration, and image coordinate and disparity quantization. The tensor voting framework has proven to be extremely robust to noise [21], [25], [35]. It is able to survive corruption of noise up to a few times the order of the inlier data. Corruption by noisy data five times larger in number than the correct data was shown not to be catastrophic in [13]. Even if similar corruption is unlikely in the case of binocular stereo, this example demonstrates the noise tolerance of the tensor voting framework. In fact, the only assumption we make about noise is that even if it accidentally forms artifacts, these should be less salient than the actual features of the scene. If that is not the case, these artifacts are extracted, a fact not inconsistent with human perception.

All the examples shown in previous sections were carried out with the same value of the scale of the voting

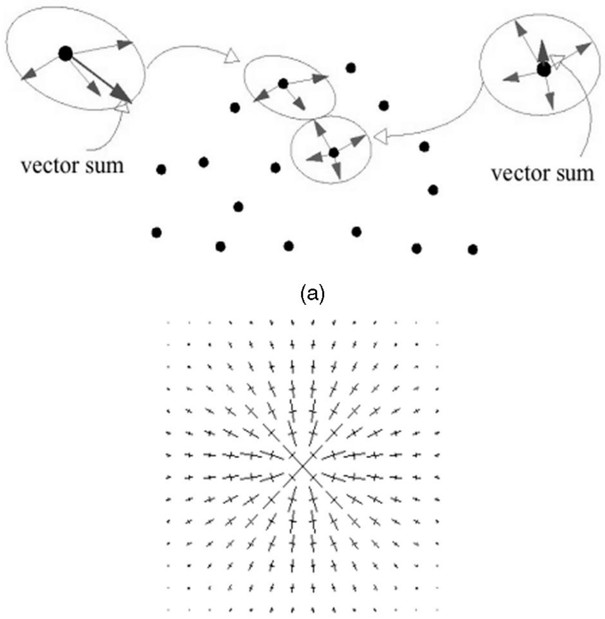

Fig. 16. Voting for boundary extraction. (a) Vector sums for boundary detection. (b) Voting field.

field. While the robustness of the algorithm with respect to this parameter is an advantage, it is also a weakness. A single scale for the entire data set is most effective when the distribution of the data does not display significant variations. If that is not the case, different scales should be applied on regions with different data densities. A large scale enables the communication of more distant locations and is less vulnerable to noise, but may cause excessive smoothing of the results. On the other hand, a smaller scale is more suitable for dense areas, as it allows local influence only and preserves details. A scheme for automatic scale adaptation is one of the major axes of our future research.

Of equal importance is the continuation of research on more accurate boundary detection. The results demonstrated in Section 6.3 are promising but the work on boundary detection that would inhibit surface overextension is far from finished. We intend to further develop our boundary detection technique and make it independent of edge detectors. Polarity information is sufficient for discontinuity detection as indicated by our experiments on synthetic data. Boundaries can be extracted as curves in 3D that display locally maximum polarity.

We intend to augment our framework by incorporating more information that currently is not utilized to the proper

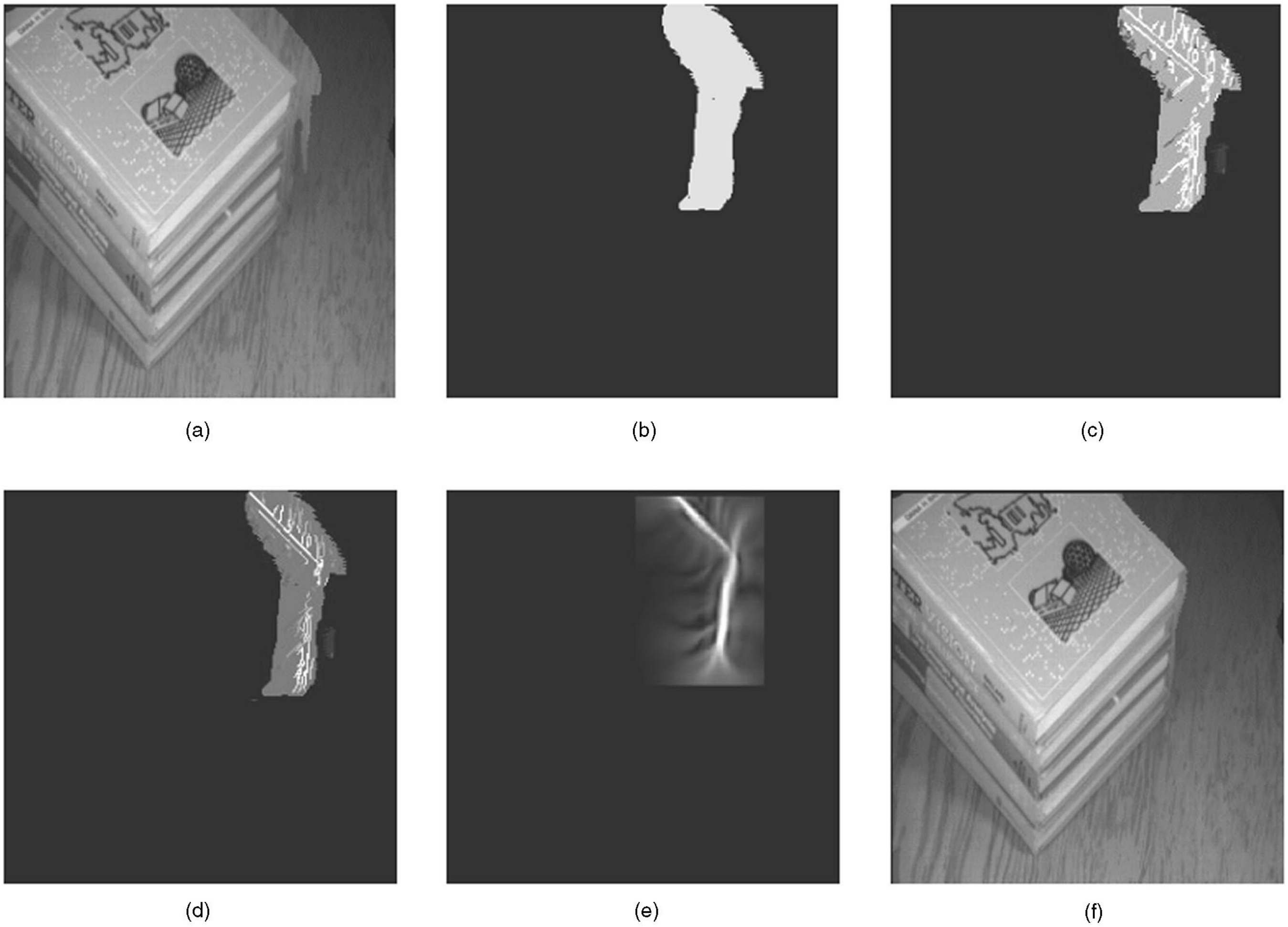

Fig. 17. Surface trimming. (a) Intensity-disparity product. (b) Overlapping regions. (c) Edge segments in the region. (d) Boundary saliency of the edgels. (e) Curve saliency. (f) Trimmed surfaces.

extent. Our naïve initialization of the tensors as balls does not take into account the fact that these locations are not points but are surfaces visible by the cameras. Therefore, the tensors could be initialized as stick tensors parallel to the lines of sight. Furthermore, curvature information that can be extracted from the data should be used to guide surface and curve extraction. So far, we have conducted experiments on curvature extraction from synthetic data with satisfactory results [36].

Finally, the efficiency of the extraction process is an area that can be considerably improved. The complexity of the Marching Cubes algorithm depends on the voxel size rather than the complexity of the extracted surfaces. The need to capture detail in some regions forces us to generate triangulated meshes of the surfaces containing large numbers of triangles even in flat regions. Postprocessing to reduce the number of triangles is an option, but a more efficient extraction technique is clearly superior.

8 Conclusion

Undeniably, there is a plethora of stereo methodologies, each encompassing interesting ideas and features. They succeed in many of the tasks involved in the process of extracting information from a binocular stereo pair of images. Unfortunately, none of them can be described as a complete approach with general applicability regardless of the scene depicted. The reasons for this are twofold: on one side, one can attribute the shortcomings to unsuitable representation of the data and incorrect implementation of the constraints and, on the other, to the lack of a framework capable of handling all the major and minor details of the stereo problem with a reasonable computational cost.

Tensor representation is sufficient and general enough for encoding all the necessary information for the problem at hand. Tensor voting is a framework that enables the communication and interpretation of this information in a complete three-dimensional space with a reasonable computational complexity. This ratio of sophisticated processing to complexity has not been attained by other methods. Equipped with these tools, we had to make some critical choices in the selection of the constraints and in the setting of our goals in order to develop a robust solution to the stereo problem. We believe that the use of saliency to determine the correctness of matches and the integration of feature matching and structure extraction to be the major contributions of this paper.

Appendix

Overview of Tensor Voting Formalism

A.1 Representation

Points can simply be represented by their coordinates. A local description of a curve is given by the point coordinates, and its associated tangent or normal plane. A local description of a surface patch is given by the point coordinates and its associated tangent or normal plane. Here, however, we do not know in advance what type of entity (point, curve, surface) a token may belong to. Furthermore, because features may overlap, a location may actually correspond to multiple feature types at the same time.

To capture first order differential geometry information and its singularities, a second order symmetric tensor is used. It captures both the orientation information and its confidence, or saliency. Such a tensor can be visualized as an ellipse in 2D or an ellipsoid in 3D. Intuitively, the shape of the tensor defines the type of geometric entity represented (point, curve, or surface element) and its size represents the saliency.

To express a second order symmetric tensor S, we choose to take the associated quadratic form and to decompose it into its eigensystem, leading to a representation based on the eigenvalues $\lambda_1, \lambda_2, \lambda_3$ and the eigenvectors $\hat{e}_1, \hat{e}_2, \hat{e}_3$. In a more compact form,

$$S = \lambda_1 \hat{e}_1 \hat{e}_1{}^T + \lambda_2 \hat{e}_2 \hat{e}_2{}^T + \lambda_3 \hat{e}_3 \hat{e}_3{}^T,$$

where $\lambda;_1 \geq \lambda_2 \geq \lambda_3 \geq 0$ are the eigenvalues, and $\hat{e}_1, \hat{e}_2, \hat{e}_3$ are the eigenvectors corresponding to $\lambda_1, \lambda_2, \lambda_3$, respectively. The eigenvectors represent the principal directions of the ellipsoid and the eigenvalues encode the size and shape of the ellipsoid.

A.2 Tensor Decomposition

As a result of the voting procedure, we produce *arbitrary* second-order, symmetric tensors; therefore, we need to handle any generic tensor. The spectrum theorem [12] states that any tensor can be expressed as a *linear* combination of three basis tensors, i.e.,

$$\begin{aligned} S = {} & (\lambda_1 - \lambda_2)\hat{e}_1 \hat{e}_1{}^T \\ & + (\lambda_2 - \lambda_3)(\hat{e}_1 \hat{e}_1{}^T + \hat{e}_2 \hat{e}_2{}^T) + \lambda_3(\hat{e}_1 \hat{e}_1{}^T + \hat{e}_2 \hat{e}_2{}^T + \hat{e}_3 \hat{e}_3{}^T), \end{aligned}$$

where $\hat{e}_1 \hat{e}_1{}^T$ describes a stick tensor with one non-zero eigenvalue, $(\hat{e}_1 \hat{e}_1{}^T + \hat{e}_2 \hat{e}_2{}^T)$ describes a plate tensor with two equal nonzero eigenvalues and $(\hat{e}_1 \hat{e}_1{}^T + \hat{e}_2 \hat{e}_2{}^T + \hat{e}_3 \hat{e}_3{}^T)$ describes a ball tensor with three equal eigenvalues. Fig. 18 illustrates the decomposition of a general second-order symmetric tensor into these components.

A dominant stick component indicates a location that most likely belongs on a smooth surface, a dominant plate component indicates preference for a smooth curve, and a dominant ball component indicates a possible curve junction. At each location, the saliency of each of the three types of information is captured as follows: ***Point-ness*** is defined by no orientation and the saliency is given by λ_3. ***Curve-ness*** is defined by a tangent orientation given by $\hat{e}_3$ and saliency by $\lambda_2 - \lambda_3$. ***Surface-ness*** is defined by a normal parallel to $\hat{e}_1$ and saliency $\lambda_1 - \lambda_2$.

A.3 Tensor Communication

We now turn to our communication and computation scheme which allows a site to exchange information with its neighbors and infer new information.

Token refinement and dense extrapolation. The input tokens are first encoded as second-order tensors. In 3D, a point token is encoded as a 3D ball. A curve element is encoded as a 3D plate. A surface element is encoded as a

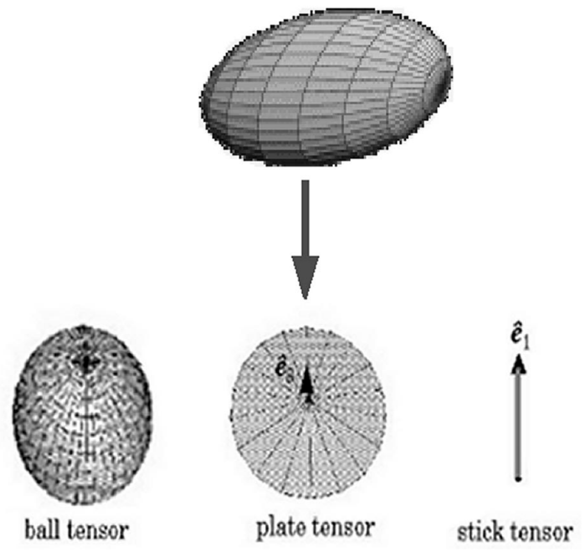

Fig. 18. Tensor decomposition.

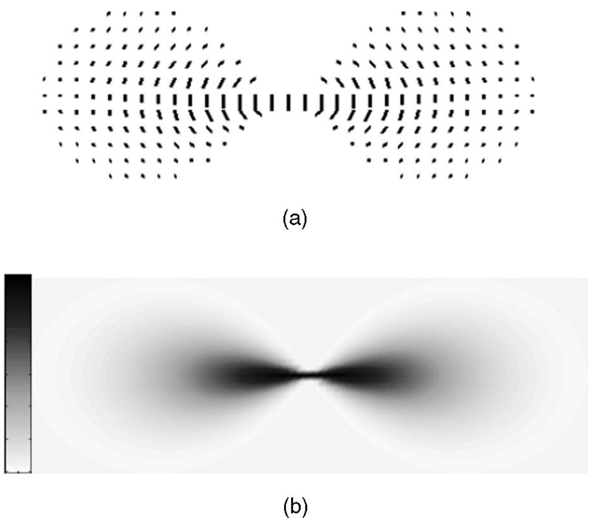

Fig. 19. The fundamental 2D stick field. (a) Normal voting field. (b) Intensity-coded strength (saliency).

3D stick. These initial tensors communicate with each other in order to derive the most preferred orientation information (or refine the initial orientation if given) for each of the input tokens (*token refinement, sparse vote)* and extrapolate the inferred information at every location in the domain for the purpose of coherent feature extraction (*dense extrapolation, dense vote*). Note that, each token is first decomposed into the basis elements, before broadcasting this information. While they may be implemented differently for efficiency, these two operations are equivalent and can be regarded as *tensor convolution* of the data set with *voting kernels*.

Derivation of the 3D voting kernels. All voting kernels can be derived from the *fundamental 2D stick kernel*, by rotation and integration. Fig. 19 shows this 2D stick kernel. In [25], we explain in mathematical terms that this voting kernel, in fact, encodes the proximity and the smoothness constraints. Denote the fundamental 2D stick kernel by V_F. The 3D stick kernel is obtained by revolving the normal version of V_F 90 degrees about the z-axis, then integrating the contributions of the rotating V_F field by rotating about the x-axis by tensor addition. To obtain the plate kernel, we rotate the 3D stick kernel obtained above about the z-axis, integrating the contributions by tensor addition. To obtain the ball kernel, we rotate the 3D stick kernel about the y-axis and z-axis, integrating the contributions by tensor addition.

Saliency decay function. We use the Gaussian function as the saliency decay function that determines the strength of the voting field with respect to distance and curvature. The voting function, in polar coordinates, for a stick tensor parallel to the x-axis with unit magnitude is:

$$\begin{aligned} \mathrm{V}(\mathrm{s},\theta,\varphi) &= \mathrm{e}^{-\left(\frac{\mathrm{s}^2+c\rho^2}{\sigma^2}\right)} \\ \rho &= \frac{2\cos\theta}{l} \\ s &= \frac{l\theta}{\sin\theta}. \end{aligned} \tag{3}$$

The distance between the vote-casting and receiving locations is denoted by l, while s is the length of the circular arc that goes through the receiving location and whose normal is the stick tensor. ϑ is the angle of the circular arc, while c is a constant. The only free parameter is σ which is used to control essentially the size of the neighborhood for data communication.

Acknowledgments

This research was funded in part by the US National Science Foundation under grant No. IRI-9811883. Funding support was also provided by Geometrix, Inc.

References

[1] S.T. Barnard and M.A. Fischler, "Computational Stereo," *Computing Survey,* vol. 14, pp. 553-572, 1982.

[2] P. Belhumeur, "A Bayesian Approach to Binocular Stereopsis," *Int'l J. Computer Vision,* vol. 19, no. 3, pp. 237-260, 1996.

[3] P. Belhumeur and D. Mumford, "A Bayesian Treatment of the Stereo Correspondence Problem Using Half Occluded Regions," *Proc. Computer Vision and Pattern Recognition,* pp. 506-512, 1992.

[4] Y. Boykov, O. Veksler, and R. Zabih, "Markov Random Fields with Efficient Approximations," *Proc. Computer Vision and Pattern Recognition,* pp. 648-655, 1998.

[5] P. Burt and B. Julesz, "A Disparity Gradient Limit for Binocular Fusion," *Perception,* vol. 9, pp. 671-682, 1980.

[6] Q. Chen and G. Medioni, "A Volumetric Stereo Matching Method: Application to Image-Based Modeling," *Proc. Computer Vision and Pattern Recognition,* vol. 1, pp. 29-34, 1999.

[7] R.T. Collins, "A Space-Sweep Approach to True Multi-Image Matching," *Proc. Computer Vision and Pattern Recognition,* pp. 358-363, 1996.

[8] I.J. Cox, S.L. Hingorani, S.B. Rao, and B.M. Maggs, "A Maximum Likelihood Stereo Algorithm," *Computer Vision and Image Understanding,* vol. 63 no. 3, pp. 542-567, 1996.

[9] U.R. Dhond and J.K. Aggarwal, "Structure from Stereo—A Review," *IEEE Systems, Man, and Cybernetics,* vol. 19, pp. 1489-1510, 1989.

[10] P. Fua, "From Multiple Stereo Views to Multiple 3-D Surfaces," *Int'l J. Computer Vision,* vol. 24, no. 1, pp. 19-35, 1997.

[11] D. Geiger, B. Ladendorf, and A. Yuille, "Occlusions and Binocular Stereo," *Int'l J. Computer Vision,* vol. 14, pp. 211-226, 1995.

[12] G.H. Granlund and H. Knutsson, *Signal Processing for Computer Vision.* Kluwer Academic, 1995.

[13] G. Guy and G. Medioni, "Inference of Surfaces, Curves and Junctions from Sparse Noisy 3D Data," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 19, no. 11, pp. 1265-1277, Nov. 1997.

[14] W. Hoff and N. Ahuja, "Surfaces from Stereo: Integrating Feature Matching, Disparity Estimation, and Contour Detection," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 11, no. 2, pp. 121-136, Feb. 1989.

[15] P.V.C. Hough, "Methods and Means for Recognising Complex Patterns," US Patent 3 069 654, 1962.

[16] H. Ishikawa and D. Geiger, "Occlusion, Discontinuities, and Epipolar Lines in Stereo," *Proc. European Conf. Computer Vision,* pp. 232-248, 1998.

[17] B. Julesz, "Binocular Depth Perception of Computer-Generated Patterns," *Bell System Technical J.,* vol. 39, pp. 1125-1162, 1960.

[18] B. Julesz, *Dialogues on Perception.* MIT Press, 1995.

[19] H. Knutsson, "Representing Local Structure Using Tensors," *Proc. Sixth Scandinavian Conf. Image Analysis,* pp. 244-251, 1989.

[20] M.S. Lee and G. Medioni, "Inferring Segmented Surface Description from Stereo Data," *Proc. Computer Vision and Pattern Recognition,* pp. 346-352, 1998.

[21] M.S. Lee and G. Medioni, "Grouping ., -, ->, O-, into Regions, Curves, and Junctions," *Computer Vision and Image Understanding,* vol. 76, no. 1, pp. 54-69, 1999.

[22] W.E. Lorensen and H.E. Cline, "Marching Cubes: A High Resolution 3-D Surface Reconstruction Algorithm," *Computer Graphics,* vol. 21, no. 4, pp. 163-169, 1987.

[23] D. Marr and T. Poggio, "A Theory of Human Stereo Vision," *Proc. Royal Soc. London,* vol. B204, pp. 301-328, 1979.

[24] D. Marr, *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information.* W.H. Freeman and Co., 1982.

[25] G. Medioni, M.S. Lee, and C.K. Tang, *A Computational Framework for Segmentation and Grouping.* Elsevier Science, 2000.

[26] S. Nalwa, *A Guided Tour of Computer Vision.* Addison-Wesley, 1993.

[27] S.I. Olsen, "Stereo Correspondence by Surface Reconstruction," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 12, no. 3, pp. 309-314, Mar. 1990.

[28] M. Okutomi and T. Kanade, "A Multiple-Baseline Stereo," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 15, no. 4, pp. 353-363, Apr. 1993.

[29] L. Robert and R. Deriche, "Dense Depth Map Reconstruction: A Minimization and Regularization Approach which Preserves Discontinuities," *Proc. Fourth European Conf. Computer Vision,* pp. 439-451, 1996.

[30] S. Roy and I.J. Cox, "A Maximum-Flow Formulation of the N-Camera Correspondence Problem," *Proc. Int'l Conf. Computer Vision,* pp. 492-499, 1998.

[31] R. Sara and R. Bajcsy, "On Occluding Contour Artifacts in Stereo Vision," *Proc. Computer Vision and Pattern Recognition,* pp. 852-857, 1997.

[32] S.M. Seitz and C.R. Dyer, "Photorealistic Scene Reconstruction by Voxel Coloring," *Proc. Computer Vision and Pattern Recognition,* pp. 1067-1073, 1997.

[33] C.V. Stewart, "MINPRAN: A New Robust Estimator for Computer Vision," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 17, no. 10, pp. 925-938, Oct. 1995.

[34] R. Szeliski and P. Golland, "Stereo Matching with Transparency and Matting," *Int'l J. Computer Vision,* vol. 32, no. 1, pp. 45-61, 1999.

[35] C.K. Tang and G. Medioni, "Inference of Integrated Surface, Curve and Junction Descriptions from Sparse 3-D Data Sets," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 20, no. 11, pp. 1206-1223, Nov. 1998.

[36] C.K. Tang and G. Medioni, "Curvature-Augmented Tensorial Framework for Integrated Shape Inference from Noisy, 3D Data," *IEEE Trans. Pattern Analysis and Machine Intelligence,* to be published.

[37] G.Q. Wei, W. Brauner, and G. Hirzinger, "Intensity- and Gradient-Based Stereo Matching Using Hierarchical Gaussian Basis Functions," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 20, no. 11, pp. 1143-1160, Nov. 1998.

[38] C.F. Westin, "A Tensor Framework for Multidimensional Signal Processing," PhD thesis, Linkoeping Univ., Sweden, 1994.

[39] A.L. Yuille and T. Poggio, "A Generalized Ordering Constraint for Stereo Correspondence," AI Memo 777, AI Lab, MIT, 1984.

[40] Z. Zhang, R. Deriche, L.T. Luong, and O. Faugeras, "A Robust Approach to Image Matching: Recovery of the Epipolar Geometry," *Artificial Intelligence J.,* vol. 78, pp. 87-119, 1995.

Mi-Suen Lee received the BSc degree from the Chinese University of Hong Kong in 1989, the MPhil degree from the University of Hong Kong in 1992, and the PhD degree from the University of Southern California in 1998, all in computer science. Since 1998, she has been a senior member of the research staff at Philips Research, Briarcliff Manor, New York. Her current research interests include perceptual grouping, robust techniques, shape analysis, shape from stereo, motion or shading, and image-based rendering. She is a member of the IEEE Computer Society.

Gérard Medioni received the Diplôme d' Ingénieur Civil from the Ecole Nationale Supérieure des Télécommunications, Paris, France, in 1977, and the MS and PhD degrees in computer science from the University of Southern California, Los Angeles, in 1980 and 1983, respectively. He has been with the University of Southern California (USC) in Los Angeles since 1983, where he is currently a professor of computer science and electrical engineering, codirector of the Computer Vision Laboratory, and chairman of the Computer Science Department. He was a visiting scientist at INRIA Sophia Antipolis in 1993 and chief technical officer of Geometrix, Inc. during his sabbatical leave in 2000. His research interests cover a broad spectrum of the computer vision field and he has studied techniques for edge detection, perceptual grouping, shape description, stereo analysis, range image understanding, image to map correspondence, object recognition, and image sequence analysis. He has published more than 100 papers in conference proceedings and journals. Dr. Medioni is a senior member of the IEEE. He has served on the program committees of many major vision conferences and was program chairman of the 1991 IEEE Computer Vision and Pattern Recognition conference in Maui, program cochairman of the 1995 IEEE Symposium on Computer Vision held in Coral Gables, Florida, general cochair of the 1997 IEEE Computer Vision and Pattern Recognition conference in Puerto Rico, program cochair of the 1998 International Conference on Pattern Recognition held in Brisbane, Australia, and general cochairman of the upcoming 2001 IEEE Computer Vision and Pattern Recognition Conference in Kauai. Professor Medioni is an associate editor of the *Pattern Recognition and Image Analysis* journal and one of the North American editors for the *Image and Vision Computing* journal.

Philippos Mordohai received his Diploma in electrical and computer engineering from the Aristotle University of Thessaloniki, Greece, in 1993, and the MS degree in electrical engineering from the University of Southern California (USC), Los Angeles, in 2000. He is pursuing the PhD degree in electrical engineering at USC. He is a graduate research assistant at the Computer Vision Laboratory of the Institute of Robotics and Intelligent Systems, and the Integrated Media Systems Center at USC. His research interests include computer vision, perceptual grouping, and integrated media systems. He is a student member of the IEEE.