Stereo using Monocular Cues within the Tensor Voting Framework

Philippos Mordohai and Gérard Medioni

Institute for Robotics and Intelligent Systems,
University of Southern California,
Los Angeles, CA 90089, USA
{mordohai, medioni}@iris.usc.edu

Abstract. We address the fundamental problem of matching two static images. Significant progress has been made in this area, but the correspondence problem has not been solved. Most of the remaining difficulties are caused by occlusion and lack of texture. We propose an approach that addresses these difficulties within a perceptual organization framework, taking into account both binocular and monocular sources of information. Geometric and color information from the scene is used for grouping, complementing each other's strengths. We begin by generating matching hypotheses for every pixel in such a way that a variety of matching techniques can be integrated, thus allowing us to combine their particular advantages. Correct matches are detected based on the support they receive from their neighboring candidate matches in 3-D, after tensor voting. They are grouped into smooth surfaces, the projections of which on the images serve as the reliable set of matches. The use of segmentation based on geometric cues to infer the color distributions of scene surfaces is arguably the most significant contribution of our research. The inferred reliable set of matches guides the generation of disparity hypotheses for the unmatched pixels. The match for an unmatched pixel is selected among a set of candidates as the one that is a good continuation of the surface, and also compatible with the observed color distribution of the surface in both images. Thus, information is propagated from more to less reliable pixels considering both geometric and color information. We present results on standard stereo pairs.

1 Introduction

The premise of shape from stereo comes from the fact that, in a set of two or more images of a static scene, world points appear on the images at different disparities depending on their distance from the cameras. Establishing pixel correspondences on real images, though, is far from trivial. Projective and photometric distortion, sensor noise, occlusion, lack of texture, and repetitive patterns make matching the most difficult stage of a stereo algorithm. To address mainly occlusion and lack of texture, we propose a stereo algorithm that operates as a perceptual organization process in the 3-D disparity space knowing that false

matches will most likely occur in textureless areas and close to depth discontinuities. Since binocular processing has limitations in these areas, we use monocular information to overcome them. We start by detecting the most reliable matches, which are grouped into layers. Shape and color information from the layers is used to infer matches for the remaining pixels.

The paper is organized as follows: Section 2 reviews related work; Section 3 is an overview of the algorithm; Section 4 describes the initial matching stage; Section 5 the detection of correct matches using tensor voting; Section 6 the segmentation process; Section 7 the disparity computation for unmatched pixels; Section 8 contains experimental results; and Section 9 concludes the paper.

2 Related Work

Published research on stereo with explicit treatment of occlusion includes numerous approaches (see [1] for a comprehensive review of stereo algorithms). They can be categorized into the following categories: local, global and approaches with extended local support, such as the one we propose. Local methods attempt to solve the correspondence problem using local operators in relatively small windows. Kanade and Okutomi [2] use matching windows whose size and shape adapt according to the intensities and disparities that are included in them. In [3] Veksler presents a method that takes into account the average matching error per pixel, the variance of this error and the size of the window.

On the other hand, global methods arrive at disparity assignments by optimizing a global cost function that usually includes penalties for pixel dissimilarities and violation of the smoothness constraint. The latter introduces a bias for constant disparities at neighboring pixels, thus favoring frontoparallel planes. Global stereo methods that explicitly model occlusion include [4][5][6][7] where optimization is performed using dynamic programming. The drawback of dynamic programming is that each epipolar line is processed independently, which results in "streaking" artifacts in the output. Consistency among epipolar lines is ensured by using graph cuts to optimize the objective function. Ishikawa and Geiger [8] explicitly model occlusion in a graph cut framework, but their algorithm is limited to convex energy functions which do not perform well at discontinuities. Kolmogorov and Zabih [9] advance the graph cut matching framework by proposing an optimization technique that is applicable to more general objective functions and obtains very good results.

Between these two extremes are approaches that are neither "winner-take-all" at the local level, nor global. They start from the most reliable matches to estimate the disparities of less reliable ones. Many authors [10][11] use the support and inhibition mechanism of cooperative stereo to ensure the propagation of correct disparities and the uniqueness of matches with respect to both images. Reliable matches without competitors are used to reinforce matches that are compatible with them and eliminate the ones that contradict them, progressively disambiguating more pixels. Zhang and Kambhamettu [12] extend the cooperative framework from single pixels to segmented surfaces, in the form

of small locally planar patches. A different method of aggregating support is nonlinear diffusion, proposed by Scharstein and Szeliski in [13], where disparity estimates are propagated to neighboring pixels until convergence. Sun *et al.* [14] formulate the problem as an MRF with explicit handling of occlusions. In the belief propagation framework, information is passed to adjacent pixels in the form of messages whose weight also takes into account image segmentation. Other progressive approaches include Szeliski and Scharstein [15] and Zhang and Shan [16] who start from the most reliable matches and allow the most certain disparities guide the estimation of less certain ones, while occlusions are explicitly labeled.

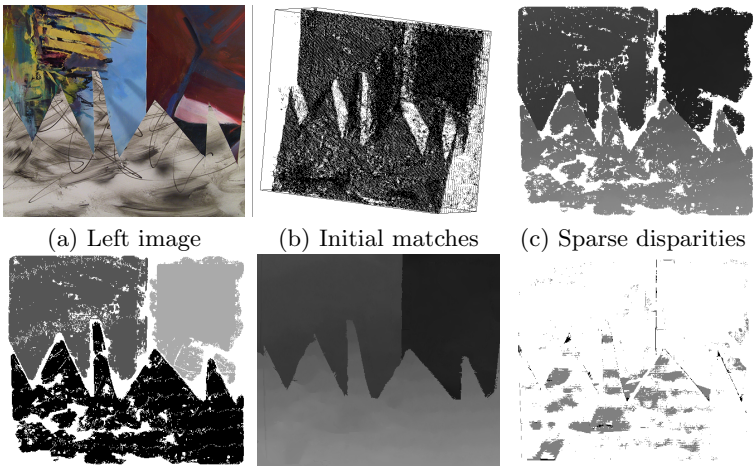
The final class of methods reviewed here are based on image segmentation. Birchfield and Tomasi [17] cast the problem of correspondence as image segmentation followed by the estimation of an affine transformation for each segment between the images. Tao *et al.* [18] introduce a stereo matching technique where the goal is to establish correspondence between image regions rather than pixels. Both these methods are limited to planar surfaces, unlike the one of [12] which was described above. Lin and Tomasi [19] propose a framework where 3-D shape is estimated by fitting splines, while 2-D support is based on image segmentation. Processing alternates between these two steps until convergence. As mentioned above, in [14] image segmentation is a soft constraint, since messages can be passed between different image segments with a lower weight. All of these approaches, however, address color segmentation independently of disparity.

The perceptual organization stage of the approach we propose here is based on the work of Lee *et al.* [20], which was later extended to multiple views in [21]. However, there are significant differences in the way initial matches are generated and, most importantly, in the integration of monocular cues to specifically address occlusion and lack of texture. The approach in [20] has a less sophisticated initial matching scheme, the failures of which cannot always be corrected. In addition, the post-processing mechanism based on edge detection it proposes is not as effective against occlusion as the approach presented here. On the other hand, information propagation in 3-D and the use of surface saliency as the criterion for the selection of pixel correspondences remain cornerstones of our approach.

3 Algorithm Overview

The proposed algorithm has four steps, which are illustrated in Fig. 1, for the "Sawtooth" stereo pair (courtesy of [1]).

- The input to the first stage is a pair of images which we assume have been rectified so that conjugate epipolar lines are parallel and share the same y coordinate. The goal is the generation of matching hypotheses for every pixel and it is accomplished with three different matching techniques. The output is a set of points in 3-D disparity space (Fig. 1(b)).
- Next is the tensor voting stage, during which the unorganized point cloud from the previous stage is encoded in the form of second order symmetric tensors which cast votes to their neighbors. Salient matches can be detected



(a) Left image (b) Initial matches (c) Sparse disparities

(d) Layer labels (e) Final disparities (f) Error map

Fig. 1. Overview of the processing steps for the "Sawtooth" dataset. The initial matches have been rotated so that the multiple candidates for each pixel are visible. Black pixels in the error map indicate errors greater than 1 disparity level, gray pixels correspond to errors between 0.5 and 1 disparity level, while white pixels are correct (or occluded and thus ignored)

based on the amount of support they receive from their neighbors. Uniqueness is also enforced at the end of this stage with respect to surface saliency and not a local measure, such as cross-correlation, which is more susceptible to noise. The output, which we term "sparse disparity map", consists of at most one match for each pixel of the reference image, which has an associated surface saliency value and an estimate of surface orientation. It can be seen in Fig. 1(c). This part of the algorithm is based on our previous work, published in [20].

- The outputs of the tensor voting are grouped, using the estimated surface orientations, into smooth layers. These are refined by removing those 3-D points that correspond to pixels that are inconsistent with the layer's color distribution. This addresses the usual problem of surface over-extension that occurs near occlusions. The over-extensions are usually not color-consistent and are removed at this stage. Thus we derive the set of reliable matches. Please note that the term layer throughout this paper is used interchangeably with surface, since by layer we mean a smooth, but not necessarily planar, surface in 3-D disparity space (x, y, d), where d denotes disparity. The label of each pixel can be seen in Fig. 1(d).
- The last module starts from a set of segmented surfaces and computes disparities for unmatched pixels. Disparity candidates are generated from the nearby layers, to which the pixel may belong based on its color. These are also validated in the right image and the final disparity is selected as the one

that is a smooth continuation of the most likely layer. The output of this stage is a dense disparity map with one disparity estimate for every pixel of the reference image including the occluded ones (Fig. 1(e)). Disparity estimation for occluded pixels is possible since the surfaces can be extrapolated using tensor voting even if they are occluded.

The algorithm is applied on the four datasets proposed in [1] and the two proposed in [22], which are also available online at http://www.middlebury.edu/stereo. Quantitative results are presented in Section 8.

4 Initial Matching

A large number of matching techniques have been proposed in the literature [1]. We propose a scheme for combining heterogeneous matching techniques, thus taking advantage of their combined strengths. For the results presented in this paper, three matching techniques are used, but any kind of matching can be integrated in the framework. The techniques used here are:

- A 5×5 normalized cross correlation window, which is small enough to capture details and only assumes constant disparity for small parts of the image.
- A 35×35 normalized cross correlation window, which is applied only at pixels where the standard deviation of the three color channels is less than 20. The use of such a big window over the entire image would be catastrophic, but it is effective when applied only in virtually textureless regions, where smaller windows completely fail to detect correct matches.
- A 7×7 symmetric interval matching window with truncated cost function as in [15]. The images are linearly interpolated along the x-axis so that samples exist in half-pixel intervals. The cost for matching pixel (x_L, y) in the left image with pixel (x_R, y) in the right image is:

$$C(x_L, x_R, y) = \sum_c min\{dist(I_{Lc}(x_i, y), I_{Rc}(x_j, y)) : \\ x_i \in [x_L - \frac{1}{2} \quad x_L + \frac{1}{2}], x_j \in [x_R - \frac{1}{2} \quad x_R + \frac{1}{2}]\} \quad (1)$$

The summation is over the three RGB color channels and $dist()$ is the Euclidean distance between the value of a color channel I_{Lc} in the left image and I_{Rc} in the right image. If the distance for any channel exceeds a preset truncation parameter $trunc$, the total cost is set to $3\times trunc$. This technique is effective near discontinuities due to the robustness of the cost function to pixels from different surfaces. Typical values for $trunc$ are between 3 and 10.

Each matching technique is repeated using the right image as reference and the left as target. This increases the true positive rate especially near discontinuities, where the presence of occluded pixels in the reference window affects the results of matching. When the other image is used as reference, these pixels do not appear in the reference window.

The maximum matching score, or the minimum cost, for every pixel is retained as a matching hypothesis. Matching scores and costs are then discarded and each hypothesis is treated equally in the following stage. A simple parabolic fit [1] is used for subpixel accuracy, mainly because it makes continuous slanted or curved surfaces appear continuous and not staircase-like. Computational complexity is not affected since the number of matching hypotheses is unchanged. Besides the increased number of correct detections, the combination of these matching techniques offers the advantage that the failures of a particular technique are not detrimental to the success of the algorithm. The 35×35 window is typically applied to very small uniform parts of the image and never near discontinuities, where color exhibits some variance. Our experiments have also shown that the errors produced by small windows, such as the 5×5 and 7×7 used here, are randomly spread in space and do not usually align to form non-existent structures. This property is important for our methodology that is based on the perceptual organization, due to "non-accidental alignment", of candidate matches in space.

5 Detection of Correct Matches

This section describes how correct matches can be found among the hypotheses of the previous stage by examining how they can be grouped with their neighboring candidate matches to form smooth 3-D surfaces. This is accomplished by tensor voting, which also allows us to infer the orientation of these surfaces.

5.1 Overview of Tensor Voting

The use of a voting process for structure inference from sparse and noisy data was presented in [23]. The methodology is non-iterative and robust to considerable amounts of outlier noise. It has one free parameter: the scale of voting, which essentially defines the size of the neighborhood of each point. The input data is encoded as second-order symmetric tensors, and constraints, such as proximity, co-linearity and co-curvilinearity are propagated by voting within the neighborhood. The tensors allow the representation of points on smooth surfaces, surface intersections, curves and junctions, without having to keep each type in separate spaces. In 3-D, a second-order tensor has the form of an ellipsoid, or equivalently of a 3×3 matrix. Its shape encodes the type of feature that it represents, while its size the *saliency* or the confidence we have in this information (Fig. 2(a)).

The tensors are initialized as unitary matrices, since no information about their preferred orientation is known. During the voting process, each input site casts votes to its neighboring input sites that contain tokens. The votes are also second-order symmetric tensors. Their shape corresponds to the orientation the receiver would have, if the voter and receiver were in the same structure. The saliency (strength) of a vote cast by a unitary stick tensor decays with respect to the length of the smooth circular path connecting the voter and receiver,

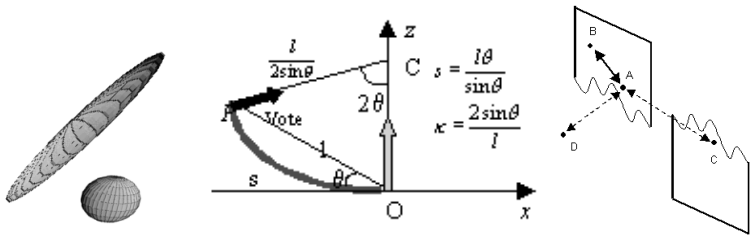
according to the following equation:

$$S(s, \kappa, \sigma) = e^{-(\frac{s^2 + c\kappa^2}{\sigma^2})} \tag{2}$$

Where s is the length of the arc between the voter and receiver, and κ is its curvature (see Fig. 2(b)), σ is the scale of voting, and c is a constant. The votes cast by un-oriented voters can be derived from the above equation, but this is beyond the scope of this paper. Vote accumulation is performed by tensor addition, which is equivalent to the addition of 3×3 matrices. After voting is completed, the eigensystem of each tensor is analyzed and the tensor is decomposed as in:

$$\begin{aligned} T &= \lambda_1 \hat{e}_1 \hat{e}_1^T + \lambda_2 \hat{e}_2 \hat{e}_2^T + \lambda_3 \hat{e}_3 \hat{e}_3^T = \\ &= (\lambda_1 - \lambda_2)\hat{e}_1 \hat{e}_1^T + (\lambda_2 - \lambda_3)(\hat{e}_1 \hat{e}_1^T + \hat{e}_2 \hat{e}_2^T) + \lambda_3(\hat{e}_1 \hat{e}_1^T + \hat{e}_2 \hat{e}_2^T + \hat{e}_3 \hat{e}_3^T) \end{aligned} \tag{3}$$

where λ_i are the eigenvalues in decreasing order and $\hat{e}_i$ are the corresponding eigenvectors. The likelihood that a point belongs to a smooth perceptual structure is determined as follows. The difference between the two largest eigenvalues encodes surface saliency, with a surface normal given by $\boldsymbol{e_1}$. The difference between the second and third eigenvalue encodes curve saliency, with a curve tangent parallel to $\boldsymbol{e_3}$. Finally, the smallest eigenvalue encodes junction saliency. If surface saliency is high, the point most likely belongs on a surface and $\boldsymbol{e_1}$ is its normal. Outliers that receive no or inconsistent support from their neighborhood can be identified by their low saliency and the lack of a dominant orientation. In the case of stereo, we assume that that all inliers lie on surfaces that reflect light towards the cameras, and therefore we do not consider curves and junctions.



(a) Example tensors (b) Vote generation (c) Voting in 3-D

Fig. 2. Tensor Voting. (a) The shape of the tensor indicates if there is a preferred orientation, while its size the confidence of this information. The top tensor has a strong preference of orientation and is more salient than the bottom tensor, which is smaller and un-oriented. (b) Vote generation as a function of the distance and curvature of the arc and the orientation of the voter. (c) Voting in 3-D neighborhoods eliminates interference between adjacent pixels from different layers

5.2 Detection of Matches as Surface Inliers

The goal of this stage is to address stereo as a perceptual organization problem in 3-D, based on the premise that the correct matches should form coherent surfaces in the 3-D disparity space. This is the only part of our approach that is based on [20]. The input is a cloud of points in a 3-D space $(x, y, zscale \times d)$, where $zscale$ is a constant used to make the input less flat with respect to the d-axis, since is disparity space is usually a lot flatter than actual (x, y, z). Its typical value is 8 and the sensitivity is extremely low for a reasonable range such as 4 to 20. The quantitative matching scores are disregarded and all candidate matches are initialized as un-oriented tensors with saliency (confidence) 1. If two or more matches fall within the same $(x, y, zscale \times d)$ voxel their initial saliencies are added, thus increasing the confidence of candidate matches confirmed by multiple matching techniques.

After the inputs have been encoded as tensors, they cast votes to their neighbors. The voting neighborhood includes all locations at which the strength of the votes is at least 2.5% of the voter's saliency. Therefore, its size is a function of σ from Eq. 2. What should be pointed out here is the fact that since information propagation is performed in 3-D there is very little interference between candidate matches for pixels that are adjacent in the image but come from different surfaces (see Fig. 2(c)). This is a big advantage over information propagation between adjacent pixels, even if it is mitigated by some dissimilarity measure.

Once voting is completed, the results can be analyzed and the surface saliency of every candidate match can be computed as in Eq. 3. Uniqueness is enforced with respect to the left image by retaining the candidate with the highest surface saliency for every pixel. We do not enforce uniqueness with respect to the right image since it is violated by slanted surfaces which project to a different number of pixels on each image. Since the objective is disparity estimation for every pixel in the reference image, uniqueness applies to that image only. The fact that a candidate match has no competition for a given pixel does not necessarily indicate that it is correct, since the correct match could have been missed at the first stage. Therefore, candidate matches with low surface saliancy are rejected even if they satisfy uniqueness. Surface saliency is a more reliable criterion for the selection of correct matches than the score of a local matching operator, because it requires that candidate matches, identified as such by local operators, should also form coherent surfaces in 3-D. This scheme is capable of rejecting false positive responses of the local operators, which is not possible at the local level. Based on the datasets we use, good results are achieved when the least salient candidates are gradually rejected until disparity estimates remain for about 70-80% of the pixels. In the data set, which we call the "sparse disparity map", remain matches with high surface saliency, which also satisfy uniqueness.

6 Segmentation into Layers

Surface inliers are segmented into layers using a simple growing scheme. By layers we mean surfaces with smooth variation of surface normal. Therefore, the

layers do not have to be planar and the points that belong to them do not have to form one connected component. Labeling starts from seed matches that have maximum surface saliency by examining matches within a certain distance in 3-D for compatibility in terms of surface normals as in Fig. 3(a). If a smooth surface that goes through the seed and the match under consideration exists, then the point is added to the layer. Further comparisons for the addition of more points to a layer are made between unlabeled points and the points from the layer that are closer to them. For all the experiments presented in this paper the grouping criteria are: $cos(\theta_1) < 0.95$ and $max\{cos(\theta_2), cos(\theta_3)\} < 0.08$. The search region, which is a non-critical parameter, is set equal to the voting neighborhood size. Since we do not attempt to fit global surface models, our grouping scheme performs equally well when the scene surfaces deviate from planar or quadric models.

To derive the reliable set of matches, one additional step is necessary to remove possible contamination from the layers due to surface over-extension from the initial matching stage. The colors of all points assigned to a layer are examined for consistency with the layer's local color distribution and the outliers are removed from the layer. Color consistency of a pixel is checked by computing the ratio of pixels of the same layer with similar color to the current pixel over the total number of pixels of the layer within the neighborhood. This is repeated for every layer on both images and if the current assignment does not correspond to the maximum ratio *in both images*, then the pixel is removed from the layer. The color similarity ratio for pixel (x_0, y_0) in the left image with layer i can be computed according to the following equation:

$$R_i(x_0, y_0) = \frac{\sum_{(x,y)\in N} T(lab(x,y) = i \, AND \, dist(I_L(x,y), I_l(x_0,y_0) < c_{thr}))}{\sum_{(x,y)\in N} T(lab(x,y) = i))} \quad (4)$$

Where $T()$ is a test function that is 1 if its argument is true, $lab()$ is the label of a pixel and c_{thr} is a color distance threshold in RGB space, typically 10. The same is applied for the right image for pixel $(x_0 - d_0, y_0)$. Rejected pixels are not added to the layer with the maximum color similarity since they are not geometrically consistent with that layer. Layers with a very small number of points, such as 0.5% of the number of pixels, are also rejected. This addresses the usual problem of surface over-extension that occurs near occlusions, since occluded pixels can be erroneously assigned the disparity of the foreground, due to the absence of a visible correspondence in the other image. The over-extensions, however, are usually not color-consistent and are removed at this stage.

Our reliable set of matches is in the form of these layers which consist of matches that are unique with respect to the left image, have high surface saliency, and are both geometrically and photometrically consistent with their neighbors. Quantitative evaluation for the reliable sets of matches is presented in Table 1. The error metric used is the one proposed in [1], where matches are considered erroneous if they correspond to un-occluded image pixels and their disparity error is greater than one integer disparity level. Compared to similar results

published in [24][25][26], our method outperforms [24] and [25] and is inferior to [26] which, however, assumes constant disparity for the dense features it detects. Also, Szeliski and Scharstein [15] report an error rate for the reliable matches for the Tsukuba dataset of 2.1% for 45% density which rises to 4% for 73% density.

Method	Tsukuba		Sawtooth		Venus		Map	
	error	density	error	density	error	density	error	density
Our results	1.18%	74.5%	0.27%	78.4%	0.20%	74.1%	0.08%	94.2%
Sara [24]	1.4%	45%	1.6%	52%	0.8%	40%	0.3%	74%
Veksler [25]	0.38%	66%	1.62%	76%	1.83%	68%	0.22%	87%
Veksler [26]	0.36%	75%	0.54%	87%	0.16%	73%	0.01%	87%

Table 1. Quantitative evaluation of density and error rate for the Middlebury stereo evaluation datasets

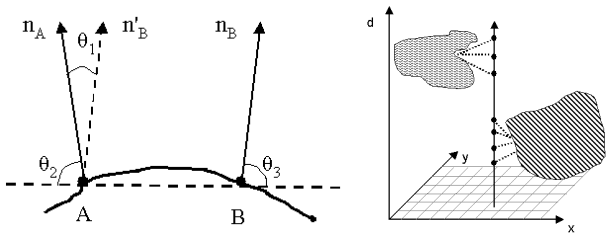


Fig. 3. (a) Surface compatibility test for surface segmentation. (b) Candidate generation for unmatched pixels based on segmented layers. Note that only matches from the appropriate layer vote at each candidate

7 Surface Growth

The goal of this module is to generate candidate matches for the unmatched pixels. Given the already estimated disparities and labels for a large set of the pixels, there is more information available now that can enhance our ability to estimate the missing disparities. Color similarity ratios are computed for each unlabeled pixel (x, y) as in Eq. 4, for all layers within the neighborhood. All ratios are normalized by their sum and layers with high normalized ratios are considered as possible surfaces for the pixel under consideration. For each candidate layer a range of potential disparities is estimated from pixels of the layer neighboring (x, y). The range is extended according to the disparity gradient limit constraint, which holds perfectly in the case of rectified parallel stereo pairs. These disparity hypotheses are verified on the target image by repeating

the same process, unless they are occluded, in which case we allow occluding surfaces to grow underneath the occluding ones. Votes are collected at valid potential matches in disparity space, as before, with the only difference being that only matches from the appropriate layer cast votes (see Fig. 3(b)). The most salient among the potential matches is selected and added to the layer, since it is the one that ensures the smoothest surface continuation.

Finally, there are a few pixels that cannot be resolved because they exhibit low similarity to all layers, or because they are specular or in shadows. Candidates for these pixels are generated based on the disparities of all neighboring pixels and votes are collected at the candidate locations in disparity space. Again, the most salient ones are selected. We opted to use surface smoothness at this stage instead of image correlation, or other image based criteria, since we are dealing with pixels where the initial matching and color consistency failed to produce a consistent match.

8 Experimental Results

This section contains results on the color versions of the four datasets of [1] and the two proposed in [22]. The initial matching in all cases was done using the three matching techniques presented in Section 4. The scale of the voting field was $\sigma^2 = 100$ (except for Tsukuba, where it was 50) which corresponds to a voting radius of 20, or a neighborhood of $41 \times 41 \times 41$. Layer segmentation was done using the thresholds of Section 6 and the color distance threshold c_{thr} was set to 10. The error metric used is the one proposed in [1], where matches are considered erroneous if they correspond to un-occluded image pixels and their disparity error is greater than one integer disparity level. Table 2 contains the error rates we achieved, as well as the rank our algorithm would achieve among the 27 algorithms in the evaluation. Due to lack of space we refer readers to the Middlebury College evaluation webpage (http://www.middlebury.edu/stereo) for results obtained by other methods. Based on the overall results for unoccluded pixels, our algorithm would rank first in the evaluation at the time of submission.

Dataset	Unoccluded		Untextured		Discontinuities	
	error	rank	error	rank	error	rank
Tsukuba	2.19%	10	0.92%	5	11.93%	11
Sawtooth	0.53%	4	0%	1	4.91%	6
Venus	0.36%	1	0.16%	2	5.00%	4
Map	0.33%	9	-	-	4.69%	10

Table 2. Quantitative evaluation for the original Middlebury stereo datasets

Table 3 reports results for the two datasets of [22] and results of three stereo algorithms, sum of squared differences (SSD), dynamic programming (DP) and graph cuts (GC) implemented by the authors of [22]. To our knowledge, our results are the best for these datasets.

Dataset	Our result	SSD	DP	GC
Cones	5.57%	17.8%	17.1%	12.6%
Teddy	9.10%	26.5%	30.1%	29.3%

Table 3. Quantitative evaluation for the new Middlebury stereo datasets

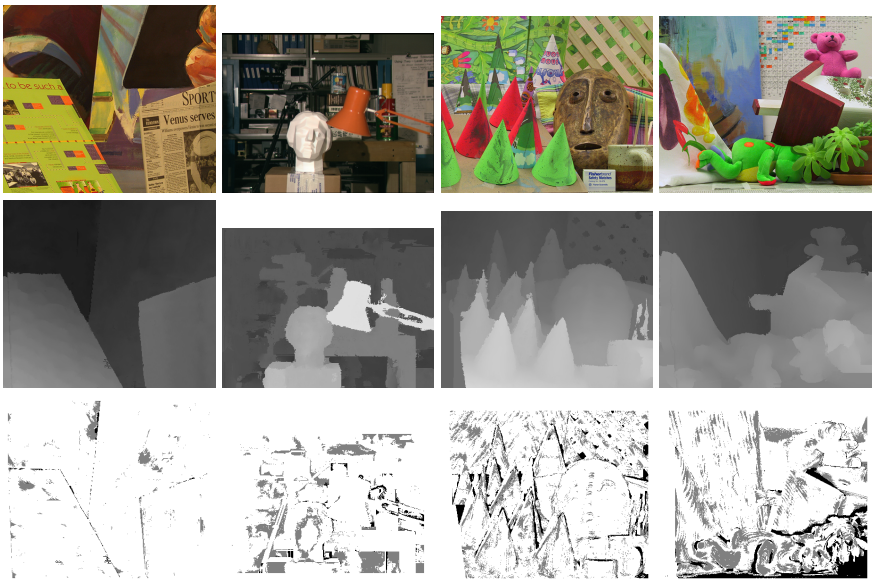


Fig. 4. Left images, final disparity maps and error maps for the "Venus", "Tsukuba", "Cones" and "Teddy" datasets from the Middlebury Stereo evaluation

9 Discussion

We have presented a novel stereo algorithm that addresses the limitations of binocular matching by incorporating monocular information. We use tensor voting to infer surface saliency and use it as a criterion for deciding on the correctness of matches as in [20] and [21]. However, the quality of the experimental results depends heavily on the inputs to the voting process, that are generated by the new initial matching stage, and the notion of geometric and photometric consistency we have introduced for the layers. Careful initial matching and the use of smoothness with respect to both surface orientation and color complement each other to derive more information from the stereo pair. Textured pixels are typically resolved by binocular matching, while untextured ones by the smooth extension of neighboring surfaces guided by color similarity. Arguably the most significant contribution is the segmentation into layers based on geometric properties and not appearance. We claim that this is advantageous over other methods that use color-based segmentation, since it utilizes the already

computed disparities which are powerful cues that provide very reliable initial estimates for the color distribution of layers.

Other contributions include the initial matching stage that allows the integration of any matching technique without any modification to subsequent modules. Information propagation in 3-D via tensor voting eliminates interference between adjacent pixels from different world surfaces. The proposed color similarity model works very well, despite its simplicity, because, locally, similar colors tend to belong to the same layer. The choice of a local non-parametric color representation allows us to handle surfaces with heterogeneous and varying color distributions, such as the ones in the Venus dataset, on which image segmentation may be hard. An important contribution of this scheme is the elimination of over-extending occluding surfaces. Finally, the implicit assumption that scene surfaces are frontoparallel is only made in the initial matching stage, when all pixels in a small window are assumed to have the same disparity. After this point, the surfaces are never assumed to be anything other than continuous.

The algorithm is able to smoothly extend partially visible surfaces to infer the disparities of occluded pixels, but fails when entire surfaces are only monocularly visible, or when occluded surfaces abruptly change orientation. It also fails when objects are entirely missed and are not included in the set of reliable matches. Over or under-segmentation is not catastrophic. For instance a segmentation of the Venus dataset into three instead of the correct four layers yields an error rate of 0.63%.

Acknowledgement

This research has been supported by the National Science Foundation grant IIS 03 29247.

References

1. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. IJCV **47** (2002) 7–42
2. Kanade, T., Okutomi, M.: A stereo matching algorithm with an adaptive window: Theory and experiment. PAMI **16** (1994) 920–932
3. Veksler, O.: Fast variable window for stereo correspondence using integral images. In: CVPR03. (2003) I: 556–561
4. Belhumeur, P., Mumford, D.: A bayesian treatment of the stereo correspondence problem using half-occluded regions. In: CVPR92. (1992) 506–512
5. Geiger, D., Ladendorf, B., Yuille, A.: Occlusions and binocular stereo. IJCV **14** (1995) 211–226
6. Birchfield, S., Tomasi, C.: Depth discontinuities by pixel-to-pixel stereo. In: ICCV98. (1998) 1073–1080
7. Bobick, A., Intille, S.: Large occlusion stereo. IJCV **33** (1999) 1–20
8. Ishikawa, H., Geiger, D.: Occlusions, discontinuities, and epipolar lines in stereo. In: ECCV98. (1998) I: 232–248

9. Kolmogorov, V., Zabih, R.: Computing visual correspondence with occlusions via graph cuts. In: ICCV01. (2001) II: 508–515
10. Luo, A., Burkhardt, H.: An intensity-based cooperative bidirectional stereo matching with simultaneous detection of discontinuities and occlusions. IJCV **15** (1995) 171–188
11. Zitnick, C., Kanade, T.: A cooperative algorithm for stereo matching and occlusion detection. PAMI **22** (2000) 675–684
12. Zhang, Y., Kambhamettu, C.: Stereo matching with segmentation-based cooperation. In: ECCV02. (2002) II: 556 ff.
13. Scharstein, D., Szeliski, R.: Stereo matching with nonlinear diffusion. IJCV **28** (1998) 155–174
14. Sun, J., Shum, H., Zheng, N.: Stereo matching using belief propagation. In: ECCV02. (2002) II: 510 ff.
15. Szeliski, R., Scharstein, D.: Symmetric sub-pixel stereo matching. In: ECCV02. (2002) II: 525–540
16. Zhang, Z., Shan, Y.: A progressive scheme for stereo matching. In: LNCS 2018, Springer Verlag (2001) 68–85
17. Birchfield, S., Tomasi, C.: Multiway cut for stereo and motion with slanted surfaces. In: ICCV99. (1999) 489–495
18. Tao, H., Sawhney, H., Kumar, R.: A global matching framework for stereo computation. In: ICCV01. (2001) I: 532–539
19. Lin, M., Tomasi, C.: Surfaces with occlusions from layered stereo. In: CVPR03. (2003) I: 710–717
20. Lee, M., Medioni, G., Mordohai, P.: Inference of segmented overlapping surfaces from binocular stereo. IEEE Transactions on Pattern Analysis and Machine Intelligence **24** (2002) 824–837
21. Mordohai, P., Medioni, G.: Perceptual grouping for multiple view stereo using tensor voting. In: ICPR02. (2002) III: 639–644
22. Scharstein, D., Szeliski, R.: High-accuracy stereo depth maps using structured light. In: CVPR03. (2003) I: 195–202
23. Medioni, G., Lee, M., Tang, C.: A Computational Framework for Segmentation and Grouping. Elsevier (2000)
24. Sara, R.: Finding the largest unambiguous component of stereo matching. In: ECCV02. (2002) III: 900–914
25. Veksler, O.: Dense features for semi-dense stereo correspondence. IJCV **47** (2002) 247–260
26. Veksler, O.: Extracting dense features for visual correspondence with graph cuts. In: CVPR03. (2003) I: 689–694