

image and the estimated corresponding mesh for the right image are shown in Fig. 3 and Fig. 5. The patch borders match with the edges where depth changes are expected. The total number of stripes and patches can be increased to get a denser mesh in order to improve the depth model approximation, but it also increases the computation time and reduces the robustness of the algorithm. Taking the original left images as reference images and using the estimated depth maps, synthetic right images are produced, which are also shown in Fig. 3 and Fig. 5.

To assess the performance of the algorithm, we used a squared frame difference error criterion between the original right and estimated right images. It is seen that error drops dramatically after the first resolution level and converges smoothly in the further levels.

5. CONCLUSION

In this contribution, a depth estimation algorithm using adaptive stripe based quadrilateral patch matching is introduced and discussed. The obtained results prove the method is promising. After a preprocessing stage, a mesh is generated from the edge maps and used to find patch correspondences between the left and right images. In each patch, the depth of the scene is assumed to be changing linearly. The equivalent of this assumption in the disparity domain is that the disparity function is affine. Following the matching stage, a depth map is produced and postprocessed.

In the stripe matching part, we have used a depth continuity assumption. This assumption constrains the depth values of the corresponding nodes in the adjoint patches to be equal. In regions experiencing occlusion this constraint is not valid. In order to handle occlusion regions, we should allow the depth values of corresponding nodes to be estimated independently. This remains a subject of our future studies.

Thus far, we have considered only the case where the surface of each patch is modeled as a plane. One future work is to extend the present algorithm to more complicated surface models. Another challenging problem is to solve the matching problem in special regions such as occluded areas [6], regions including periodic patterns or flat patterns.

6. REFERENCES

[1] U.R. Dhond and J.K. Aggarwal, "Structure from stereo - A review", *IEEE Trans. Systems, Man, and Cybernetics*, vol. 19, no. 6, pp. 1489-1510, November 1989.



Figure 4: *Sergio* pair, original left and right images



Figure 5: Generated mesh, estimated mesh and estimated right image

- [2] S. Barnard and M. Fisher, "Computational Stereo", *ACM Comput. Surveys*, vol. 14, no. 4, pp. 553-572, December 1982.
- [3] T. Kanade and M. Okutomi, "A stereo matching algorithm with an adaptive window", *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 16, no. 9, pp. 920-931, September 1994.
- [4] S. Panis, M. Ziegler and J.P. Cosmas, "System approach to disparity estimation", *Electronic Letters*, vol. 31, no. 11, pp. 871-873, May 1995.
- [5] Y. Wang and Q. Lee, "Active mesh-a future seeking and tracking image sequence representation scheme", *IEEE Trans. Image Process.*, vol. 3, no. 5, pp. 910-924, September 1994.
- [6] S. Cochran and G. Medoni, "3-D surface description from binocular stereo", *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 14, no. 10, pp. 981-994, October 1992.
- [7] J.M. Prager, "Extracting and labeling boundary segments in natural scenes." *IEEE Trans. PAMI* 2, pp. 16-27, January 1980.
- [8] A. Tamtaoui and C. Labit, "Constrained disparity and motion estimators for 3DTV image sequence coding", *Signal Process. Image Comm.*, vol. 4, no. 1, pp. 45-54, November 1991.

which means disparity at this point is a function of the coordinates and depths of corresponding patch corners. The function g depends on the depth model and should have a degree of freedom up to four.

Let $\Psi_{s,r}(X, Y)$ be the depth model function that models the depth changes in the world coordinates corresponding to the r^{th} patch of the s^{th} stripe. If the *world* patch is assumed to have a constant depth value, i.e. $\Psi_{s,r}(X, Y) = c$, where c is a constant nonnegative number, then g is also a constant independent of the position $\mathbf{p}$. If the depth of the world patch is modeled as a plane $\Psi_{s,r}(X, Y) = aX + bY + c$, then the function g is an *affine* function of $\mathbf{p}$ and its parameters are the functions of $M(s, r_i)$, $Z(s, r_i)$ (which in turn depends on $d(s, r_i)$), focal length and baseline distance. These parameters can be derived by a least square fitting. If a nonlinearly changing depth surface is assumed then g can be approximated by a rational function up to a certain order.

3.2. Multi-Resolution Matching Algorithm

To determine the depth values $Z(s, r_i)$, we first determine $d(s, r_i)$, which essentially requires matching each node $M(s, r_i)$ in the left image with a corresponding node $\widehat{M}(s, r_i)$ on the right image. After generating the mesh, preprocessing the input stereo pair to compensate possible luminance differences [4], and cropping out the sides (left side of the left and right side of the right image for zero camera system convergence angle) which are not visible in both of the images, matching is done by minimizing a matching error between corresponding trapezoid elements in the stereo pair for each stripe.

At the beginning, there is no initial depth map for the first stripe to start the iterative algorithm. The depth values of the first stripe at this level is initialized with an average depth. The first order gradient descent method is employed to minimize the matching error. The matching error function for the s^{th} stripe is defined as

$$E_s = \frac{1}{2} \sum_{j=1}^{P_s} \sum_{\mathbf{x} \in R^j} [f_L(\mathbf{x}) - f_R(\mathbf{x} - g(\mathbf{x}, M(s, r_i), Z(s, r_i)))]^2$$

where P_s is the number of patches in the stripe and R^j is the set which includes all pixels in the j^{th} patch. The updated positions of two nodes along each border line are evaluated from the matching error over the *two* adjoint patches. In these two patches, disparity values of the image patch points are derived from $Z(s, r_i)$. The depth values for the world patch are obtained from

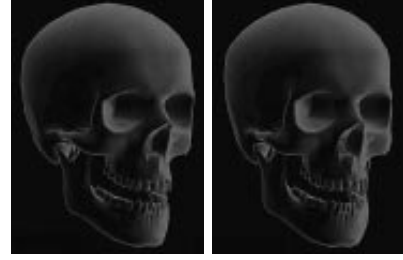


Figure 2: *Skull* pair, original left and right images

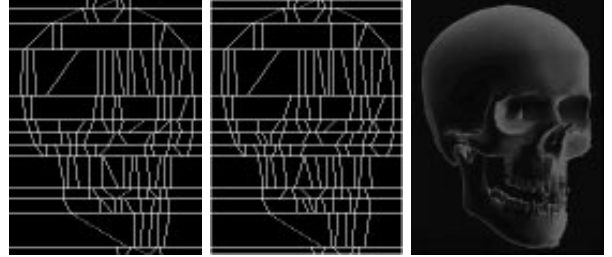


Figure 3: Generated mesh, estimated mesh and estimated right image

Ψ and transferred into image plane disparity domain by the function g .

Once the depth values of one stripe are obtained, they are used to initialize the depth values in the next stripe by using a vertical continuity criterion. This criterion is acquired from the horizontal edge map and assumes the depths of two points on the same vertical line are similar if there is no horizontal edge between them. Successive stripes are processed in the same way.

To speed up the computation time as well as providing a global continuity constraint which is suitable to most of the cases, a multi-resolution approach is employed. The previously described algorithm is first applied to the coarsest resolution and the resultant depth map is then propagated to a finer resolution. Lastly, a postprocessing filter is applied in order to smooth the resultant depth map.

4. EXPERIMENTAL RESULTS

The proposed matching technique is evaluated using both synthetic and natural images. Fig. 2 shows Skull image pair of size 180×240 and Fig. 4 is Sergio image pair of size 128×128 with depth range $0.1m - 3m$. Since the depth model function Ψ is chosen as a planar function, the resultant disparity function g is a strictly affine function. The idea is to approximate depth values inside each patch sufficiently close to real values while keeping the disparity estimation function as simple as possible. The generated mesh for the left

adjoint *two* patches on the stripe. Besides, a matching error function calculated for a stripe element (i.e. a trapezoid) is faster in computation time than the one calculated for an arbitrary triangle or quadrangle. The left and right borders of patches is chosen such that the borders correspond to distinctive depth changes and the patch approximates a flat surface in the 3D scene. This provides disparity field segmentation which is necessary to avoid smoothing disparity field over object boundaries. The shape of the patches are determined depending on the local intensity changes due to depth variations in the scene. Thus, our method overcomes the deficiencies of the block-based matching methods which use constant sized and shaped matching windows.

In Section II, we develop a mesh generation algorithm. The depth modeling and estimation algorithm are described in Section III. Section IV provides experimental results.

2. ADAPTIVE MESH GENERATION

Let the baseline be parallel to the x -axis and the camera imaging planes be coplanar. Let the stereo intensity images after some preprocessing be $f_L(\mathbf{x})$ and $f_R(\mathbf{x})$ where $\mathbf{x}$ is a two dimensional position vector. Then f_L and f_R are related by

$$f_R(\mathbf{x}) = f_L(\mathbf{x} - \mathbf{d}(\mathbf{x})) + n(\mathbf{x})$$

where $\mathbf{d}(\mathbf{x})$ is the disparity vector function and $n(\mathbf{x})$ represents intensity change due to photometric effects as well as noise. In the present study, we ignore this term for simplicity. The horizontal edge $H(\mathbf{x})$, and the omnidirectional edge maps $E(\mathbf{x})$, which will be used for mesh generation, are obtained by applying the 3×3 *sobel* filter to the left image followed by a confining confidence algorithm [7].

The mesh $M(s, r_i)$, where s stands for stripe number, r stands for patch number, i is one of the four corners, is derived from $H(\mathbf{x})$ and $E(\mathbf{x})$. Firstly, an edge strength for each row is calculated by adding the horizontal edge magnitudes $H(\mathbf{x})$ along a band around each row. All rows are ordered with respect to their edge strengths and the row with maximum strength value is selected if there is no previously selected row close to it. The minimum inter-row distance constraint and an edge strength threshold are included to limit the number of stripes as well as to avoid overly fine divisions. After the stripes are obtained, the next step in the algorithm segments each of these stripes into quadrilateral regions. Based on the edge map, $E(\mathbf{x})$, the left and right borders of each quadrilateral are determined. This is accomplished by ordering all possible

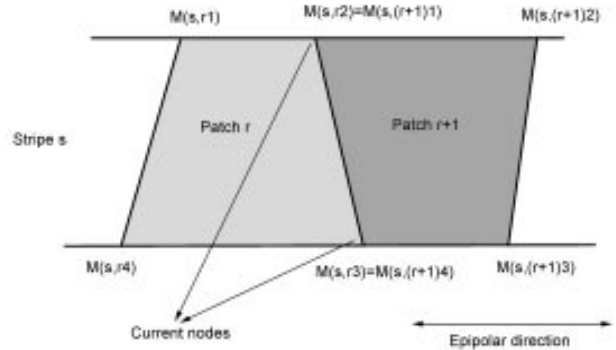


Figure 1: Two patches from the mesh

borders according to an edge score, then selecting the first N of them. The edge score for a borderline is defined as the summation of the edge magnitudes in pixels along the neighborhood of the borderline. To avoid two borders to become too close, the selection is done sequentially, so that the new border line selected is sufficiently far away from the previously chosen borders. These will yield the values of $M(s, r_i)$ for $i = 1 \dots 4$, the coordinates of r^{th} patch in s^{th} stripe.

3. DEPTH ESTIMATION

3.1. Depth and Disparity Modeling

Let $Z(s, r_i)$ denote the depths at the patch corners. There are two options for estimating these depth values. In the first option, the depth of imaged scene changes continuously inside each patch as well as across patch borders. With the second option, the depth is assumed to be continuous inside each patch but jumps are allowed between the neighboring patches. When the depth continuity constraint for mesh is employed, the value of $Z(s, r_i)$ is equal to $Z(s, (r+1)_j)$ for the pairs $(i, j) = (2, 1), (3, 4)$ as shown in Fig. 1.

If the corresponding mesh in the right image is $\widehat{M}(s, r_i)$ then we relate M and $\widehat{M}$ by

$$\widehat{M}(s, r_i) = M(s, r_i) + d(s, r_i)$$

Note that because epipoles are in the horizontal direction, the disparity in the vertical direction is zero which reduces the disparity vector to a real number. If the vector $\mathbf{p}$ is the coordinates of a point inside r^{th} patch of s^{th} stripe, the disparity of this point is found by

$$d(\mathbf{p}) = g(\mathbf{p}, M(s, r_i), Z(s, r_i), i = 1..4)$$

ADAPTIVE STRIPE BASED PATCH MATCHING FOR DEPTH ESTIMATION

Fatih Murat Porikli *Yao Wang*

Polytechnic University
Brooklyn, NY 11201, USA
fporikli@imagex1.poly.edu
yao@vision.poly.edu

Cassandra Swain

AT&T Research Lab
Holmdel, NJ 07733, USA
swain@research.att.com

ABSTRACT

In this contribution, a novel stereo matching technique for depth estimation in stereoscopic image pairs is presented. The input image pair is preprocessed in the intensity domain and edge maps together with an adaptive mesh in which individual elements approximate linearly modeled regions obtained. Then, an iterative stripe based, quadrilateral patch matching technique is employed to estimate the depth map from the image pair in a hierarchical manner. Finally, the resultant map is postprocessed to smooth the depth map at the patch borders. The quality of test results demonstrates the effectiveness of the technique.

1. INTRODUCTION

Locating the same point in multi-viewed images is one of the most difficult aspects of developing computational algorithms for stereopsis. With the correspondence of pair-wise related image points which represent a single point in the physical scene, the structure or depth can be recovered from a priori knowledge of the camera system geometry. The majority of the stereo matching approaches can be classified as the block-based and feature-based methods. In block-based methods, matching is done based on intensity pattern similarity of a block around the point. The aim is to obtain correspondences for every image point. On the other hand, feature-based methods assign disparity which is the lateral difference between the matched points, only to feature points such as corner points, edges or zero-crossings. However, most stereo algorithms tend to produce error due to noise, low feature content, depth discontinuities, occlusion, photometric differences and etc. It is well known that for certain types of matching primitives, finding correspondence is an ill-posed problem [1]. To regularize this problem, a smoothness constraint on the disparity field is generally incorporated, although smoothing over disparity boundaries is physically incorrect. Previous research on

block-based techniques [2],[3], showed that the matching window size must be large enough to include sufficient intensity variation for matching but small enough to avoid the effects of projective distortion. If the window is too small or doesn't cover enough intensity variation, the estimation is poor because of low intensity to noise ratio. Conversely, if the window is too large, the estimated depth may not represent correct matching because of over averaging of the spatial information.

In this paper, we propose a novel depth estimation technique. The principal constituents of our technique are: 1) generation of a mesh, 2) modeling depth functions for mesh components, and 3) extraction of disparity and depth by error minimization in a hierarchical manner. The novelty of the technique comes from introducing a special mesh structure into stereo vision. Rather than using an arbitrarily structured mesh, we use a stripe mesh in which an image is divided into horizontal stripes, and each stripe is further divided into quadrilaterals. This mesh belongs to the general category of the quadrilateral mesh [5], but each quadrilateral element is constrained to have parallel sides on the top and bottom, i.e, a trapezoid. See Fig. 1 for an example. The motivation of using stripes comes from the nature of the correspondence problem. For a non-vergent camera system geometry, disparity vectors lie parallel to the epipolar lines [8]. This epipolar constraint allows for a controlled correspondence search strategy. Thus, a stereo matching scheme which specializes in epipolar-directional searching is more effective in terms of speed and accuracy. The epipolar line constraint and the uniqueness constraint, which asserts a given point in the image may be assigned at most one disparity value, can be applied simultaneously to a mesh consists of stripes along the epipolar lines. By using a stripe mesh rather than an arbitrary mesh, we keep the set of all potential correspondences for a patch in a simple form, therefore processing on this set, namely on the stripe, becomes more efficient. Unlike an arbitrary mesh, reallocation of nodes only affects the