

# Multipass Stereo Matching Algorithm Using High-Curvature Points on Image Profiles

Yuan-Chih Peng and Sheng-Jyh Wang

Department of Electronics Engineering, Institute of Electronics,  
National Chiao Tung University, Hsin-Chu, Taiwan, R.O.C.

## ABSTRACT

In this paper, we propose a new algorithm to do correspondence for stereo images. This algorithm applies two passes of feature-based matching to establish a coarse disparity map first. Then, by carefully matching the intensity information, a dense disparity map is generated. In this algorithm, instead of the commonly used "edge" points, the high-curvature points of image profiles are chosen as the feature points to be matched. These high-curvature points can be easily extracted from the images by checking the 2<sup>nd</sup> derivatives of the intensity profiles. These high-curvature features can faithfully catch the major characteristics of the profile shape and can thus avoid some ambiguities in feature matching. A dissimilarity measure, which is closely related to the profile shape, is thus defined using these feature points. To reduce the ambiguity in local matching, the dynamic programming technique is used to achieve a global optimal correspondence. After the feature matchings, an intensity-based approach is used to establish a dense disparity map. Both the sum-of-squared-difference method (SSD) and the dynamic programming method are used. By carefully checking the consistence between intensity continuity and disparity continuity, a fairly accurate disparity map can be efficiently generated even if the images are short of texture.

Keywords: Stereo, Correspondence, Dynamic Programming, High-Curvature Points

## 1. INTRODUCTION

Geometric Stereo is a way to estimate the 3-D depth information of a scene by using two or more images shot at difference positions. This kind of techniques has found plenty of applications in machine vision, automatic navigation, and virtual reality. In these geometric stereo techniques, one of the major difficulties is the correspondence problem -- that is, how to efficiently and accurately pair up points in the images such that both points in a pair are projected from the same point in space. Up to now, these correspondence algorithms are either intensity-based or feature-based, or a hybrid of both [2,3,4,7,11,12,14,15,16,18,21]. Intensity-based algorithms do correspondence by checking the intensity information directly and can generate a dense disparity map. However, the computation load is usually heavy. On the other hand, feature-based algorithms usually have advantages in computation and accuracy, but only a sparse disparity map can be built. In this paper, we present a new approach, which does feature matching in the first stage and does intensity matching in the second stage. Instead of using the commonly used "edges" for matching, we propose the usage of high-curvature points as the features of the images. Based on these high-curvature points, we can easily define a cost function, which is related to the shape of the intensity profiles. In Section 2, we will present the architecture of our system. The detail about feature-based matching and intensity-based matching will be mentioned in Section 3 and Section 4, respectively. Then, some simulation results will be shown in Section 5 and we conclude the paper in Section 6.

## 2. SYSTEM ARCHITECTURE

Figure 1 shows the basic architecture of our system. In this system, we first extract the feature points from both the left image and the right image. The features we extract are the high-curvature points on the intensity profile of the epipolar lines. Based on these high-curvature points, we can easily define a cost function, which is related to the shape of the intensity profiles. Then, the dynamic programming method is used to match features. This feature-based approach requires lower computation complexity and can achieve a reliable matching. After feature matching, a sparse disparity map is generated. To ensure the correctness of the matching, these matched feature pairs are double-checked and the feature disparity map is modified, using the intensity information between features. Then, the 2<sup>nd</sup> pass of feature extraction, matching, and verification is applied to detect finer features.

To generate a dense disparity map, most feature-based algorithms do linear interpolation using the disparity information of these matched feature points. However, applying linear interpolation directly may lead to an inaccurate result. One reason is due to the existence of occlusion regions. Another reason is due to the imperfection of the feature extraction algorithm, which may generate fake features or cause the missing of some important features. To estimate depth accurately, occlusion region should be identified correctly. On the other hand, using features only cannot detect the occlusion reasons correctly. In our system, we apply an intensity-based matching after the feature-based matching to establish the dense disparity map. Both the sum-of-squared-difference (SSD) method and an intensity-based dynamic programming algorithm are used. The SSD method has lower computation complexity but cannot deal with the problem of perspective distortion. On the contrary, the dynamic programming method allows more flexible situations, but is more computationally expensive. In our strategy, we apply the SSD method first. If the estimate turns out not correct, the dynamic programming method is applied then.

From feature-based matching to intensity-based matching, only the intra-scanline relationship has been considered. In most images, however, neighboring scanlines usually have high correlation. Hence, at the last step, we use the inter-scanline information to reduce the ambiguity or matching error in the generated dense disparity map. Here, we check the continuity of intensity and the continuity of disparity to further improve the quality of the disparity map.

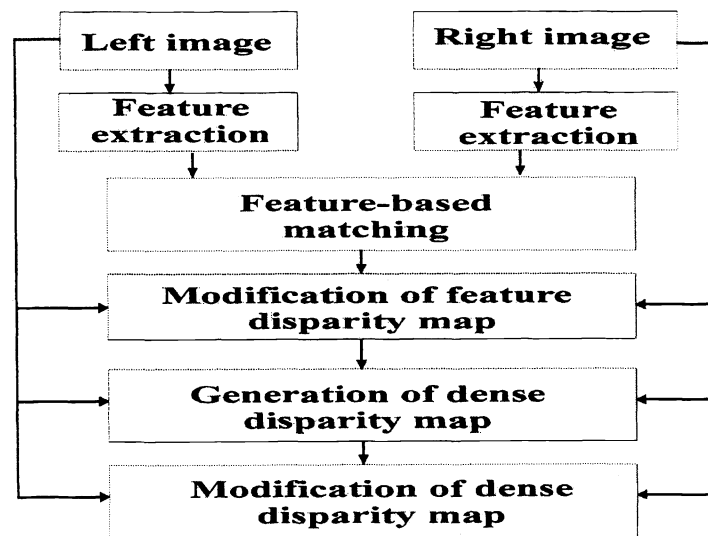


Figure 1 System block diagram

### 3. FEATURE-BASED MATCHING

#### 3.1 FEATURE EXTRACTION

The feature selection problem is highly related to the image representation problem. An appropriate image representation can facilitate the analysis of image. In image processing, “edge” and “region” are two commonly used features. A boundary-based segmentation tries to locate the places where the intensity value of the image has an abrupt change. These parts correspond to the edge of an object (see Figure 2(b)). On the other hand, a region-based segmentation merges parts of similar intensity value into regions (see Figure 2(c)). However, no matter which type of feature is used, it is very difficult to reconstruct the original profile back based on these extracted features. In this paper, we use a different type of feature: the corners of an image profile (see Figure 2(d)). These corners correspond to the high-curvature points of a profile. By using the locations and intensity values of these high-curvature points, we may simply use linear interpolation to reconstruct the original profile back. In other words, these high-curvature points contain the significant information of an image profile. This type of information becomes very useful when we want to define the shape of the profile or to verify the correctness of the disparity map. In our algorithm, we apply the one-dimensional version of Jong and Wang’s operator to extract the high-curvature points [9]. This operator convolves the image profile with the 2<sup>nd</sup> derivative of a Gaussian function. The result is thresholded then and the local maxima are marked as the feature points.

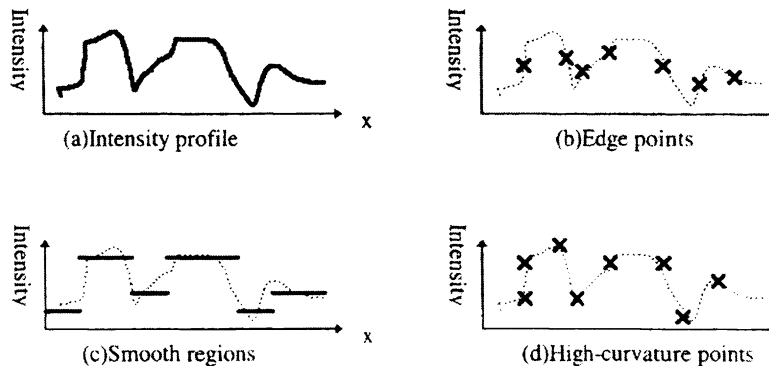


Figure 2 Image representations

#### 3.2 COST FUNCTION

In our opinion, the shape of the image profile should play an important role in doing correspondence. Since we choose the high-curvature points as the feature points, we want to find a measurement that can reflect the similarity of shape around the corners of the intensity profiles. Instead of considering a cost function based on a complex probability model, we try to use a more intuitive way to mimic the way human beings match two similar profiles. Here, we define the dissimilarity function between two feature points as

$$D = \left| \text{LeftContrast}_{left} - \text{LeftContrast}_{right} \right| + \left| \text{RightContrast}_{left} - \text{RightContrast}_{right} \right|$$

Figure 3 shows the definition of “contrast” near a high-curvature point. The contrast, defined as the intensity difference between two high-curvature points, contains the trend between the points. It can even represent the semi-global shape of an image profile.

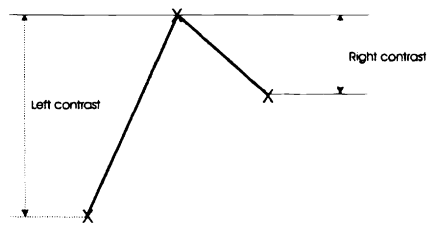


Figure 3 The definition of “contrast” near a high-curvature point.

Figure 4 shows a simulation result of the dissimilarity measurement. As shown in Figure 4(c), when the shape around the feature points is more similar to the shape of the matched feature, the value of  $D$  becomes smaller. This measurement can easily distinguish convex and concave shapes around a feature point. Furthermore, this measurement matches a high-contrast feature point to another high-contrast feature point.

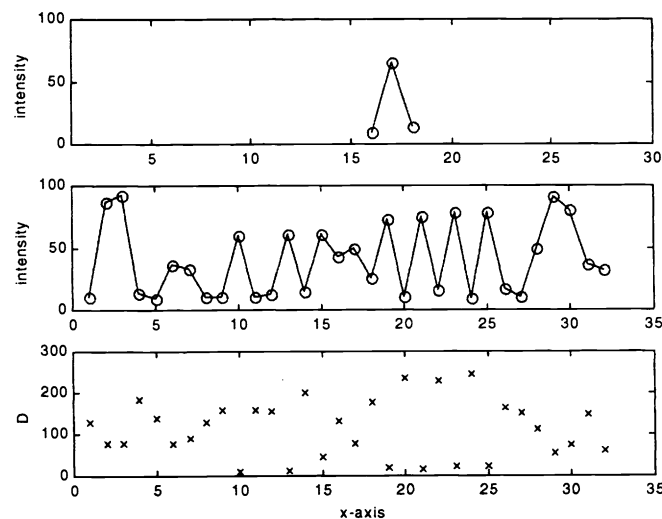


Figure 4 Dissimilarity measurement

- (a) The shape around a feature point in the left image profile.
- (b) The image profile of the right image with the feature points marked ‘o’.
- (c) The dissimilarity  $D$  of the features in the right image with respect to the left feature point.

The cost function is then defined as

$$Cost(path) = N \cdot C_{occ} + \sum D(x, y),$$

where  $N$  is the number of occluded feature points,  $C_{occ}$  is the occlusion cost, and  $D$  is the local dissimilarity measurement. The best matching is a matching sequence with the minimal matching cost. This cost accumulates the local dissimilarities to achieve a global cost to reflect the dissimilarity of a

matching sequence. The occlusion cost is the threshold to declare an occlusion. Each time the matching sequence skips a feature point, an occlusion cost is added.

### 3.3 SEARCHING FOR OPTIMAL MATCHING

Based on the cost function, the dynamic programming technique can be applied to find the optimal match sequence by searching over all the match sequences under the monotonic ordering assumption and the constraint on the minimum distance between cameras and objects. For a line with high-curvature points ( $n=60$ , typically), the rough upper bound of the computation complexity of exhaustive search is about  $n!$ . On the other hand, the computation complexity of dynamic programming is only about  $n^3$  without any constraints and simplification.

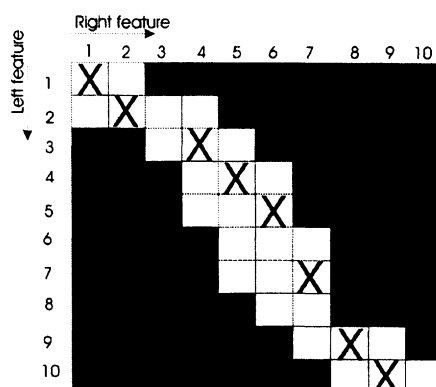


Figure 5. Cost map

The dynamic programming technique will generate a cost map which records the smallest cost accumulated from the initial matching point to the current matching point (see Figure 5). Let the y-axis represent the number of features in the left image and the x-axis represent the number of features in the right image. Each point  $(x,y)$  in the cost map indicates the accumulated cost from the initial matching point to the current matching point. Each path in the map represents a match sequence. Because we have assumed all the objects should be away from the cameras with a minimum distance, the maximum allowable disparity is defined. The impossible matching pairs, which are blackened in the cost map, are the matching pairs whose disparity is either larger than the maximum allowable disparity or smaller than zero. After the cost map is filled, the path with the minimal cost is then backtracked.

### 3.4 MODIFICATION OF FEATURE DISPARITY MAP

In the modification of the feature disparity map, we set two constraints and develop an error detection algorithm to correct the errors in the feature disparity map:

1. Depth is piecewise continuous. this constraint will remove the feature points with an isolated disparity. The isolated disparity usually relates to a feature matching error.
2. The connection of feature points in one image should not be broken after the matching.

To detect the matching errors, a simple and straightforward way is to check the intensity information between features. In our approach, one intensity profile is warped to match the other intensity profile by using the disparity information at the matched feature points. Since the feature points locate at the corners of the profiles, it is fairly easy to register the intensity profiles between the corresponding feature points. If the corresponding feature points are projected from the same point in the 3-D scene, the warped image should be very similar to the other image. Hence, we may use two error measures to identify the possibly mismatched feature points. These two measures we use are the maximal error and the bias of the mean intensity between the registered intensity profiles:

$$MaxErr(x) = \max_{x \in A} \{I(x) - R(x)\},$$

and

$$Bias(x) = \left( \sum_{i \in A} I(i) - \sum_{j \in A} R(j) \right) / L,$$

where  $I(x)$  is the original image profile,  $R(x)$  is the warped image profile,  $A$  is the interval where the point 'x' lies, and  $L$  is the length of the interval. Figure 6 demonstrates an example. Figure 6(a) shows the intensity profiles before registration and Figure 6(b) shows the profiles after registration. Figure 6(c) shows the  $MaxErr(x)$  and Figure 6(d) shows the  $Bias(x)$ .

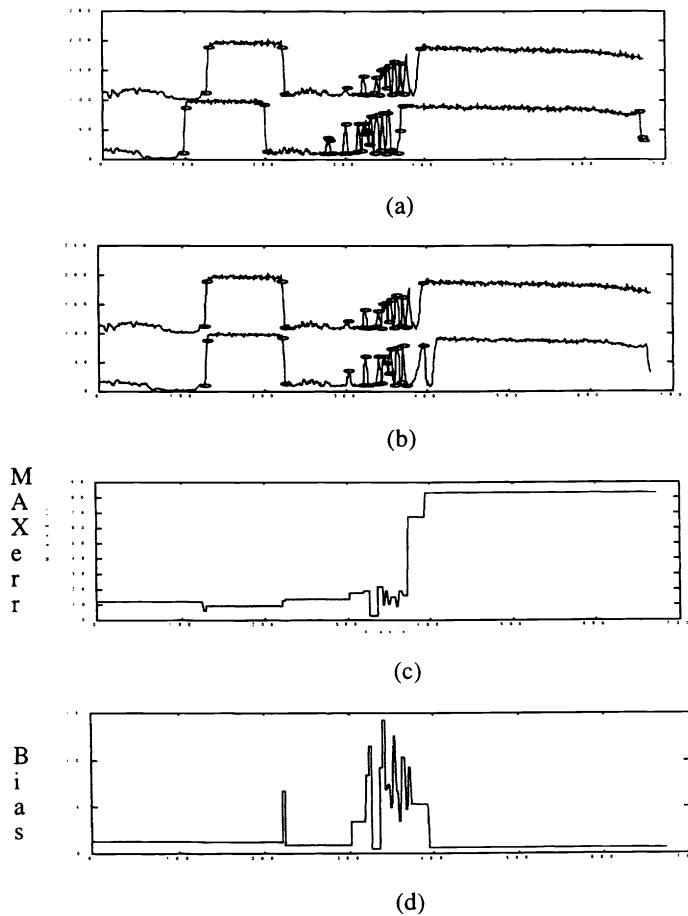


Figure 6 Error Measurement

- (a) Intensity profiles before registration
- (b) Intensity profiles after registration
- (c)  $MaxErr(x)$
- (d)  $Bias(x)$

## 4. INTENSITY-BASED MATCHING

### 4.1 INTENSITY-MATCHING

To generate a dense disparity map after feature-based matching, a conventional method is to do linear interpolation between features or to apply the dynamic programming technique again to the regions between every two matched features points. However, the former method is usually not accurate enough and the latter method needs a heavy computation. Since the image has already been segmented and the significantly varying parts of the profiles have been allocated by the feature points, we may just use a simple matching index to measure the similarity between two segmented regions. Figure 7 shows the process of the intensity-based matching. The striped blocks represent the matched feature points. Each solid line indicates a matched feature pair. The matching algorithm shifts one intensity segment with respect to the other segment and uses the sum-of-squared-difference (SSD) method to determine how these two segments can be best matched to each other. The optimal value of the shift is assigned as the disparity of this segment pair and the unmatched pixels are marked as occluded. This SSD block matching algorithm is fast but cannot deal with projective distortion. Hence, if no appropriate match can be achieved, the dynamic programming method is applied to match the intensity values of these two segments. This dynamic programming method needs higher computation complexity but can allow more complicated situations.

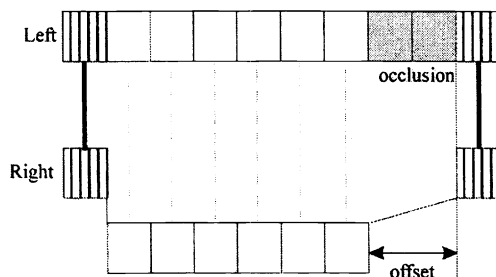


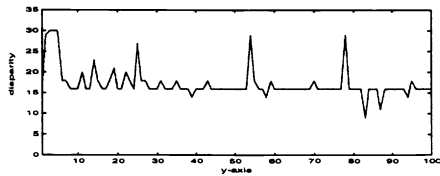
Figure 7 An illustration of the intensity-based matching.

### 4.2 MODIFICATION OF DENSE DISPARITY MAP

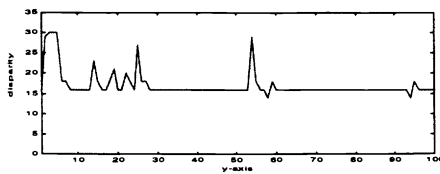
Based on the same constraints adopted in the modification of feature disparity map, we first remove isolated disparity points. We check the map along both the x-direction and the y-direction. If there is a narrow impulse in the disparity map, the disparity of that point will be replaced by the disparity of neighboring points. Figure 8(a) illustrates an image profile plotted along the y-direction. Figure 8(b) shows the profile modified by this operation. We also extend the modification method to correct the window-shape disparity along the y-direction.

On the other hand, if there is no significant intensity variation, there should be no depth discontinuity. We first check the points at which there is a discontinuity in disparity. If the length between two disparity discontinuity points is under a threshold, the region will be marked. Then, we check both the intensity variation within the marked regions and the intensity variation at the boundaries. If there is no intensity variation both within the region and at the boundaries, the disparity of the marked region will be interpolated from the adjacent unmarked regions. If there is an intensity

variation within the region or at the boundaries, the disparity of the region will be estimated from the neighboring regions (see Figure 9).



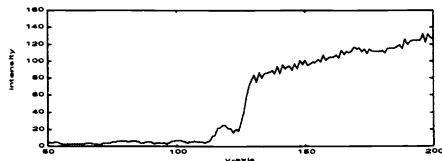
(a)



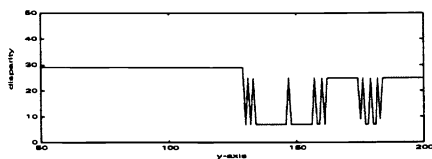
(b)

Figure 8 (a) Disparity along the y-axis.

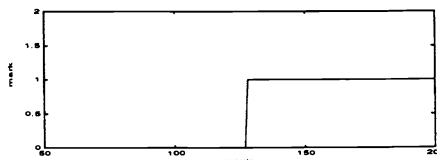
(b) Modified disparity after removing impulse-like parts.



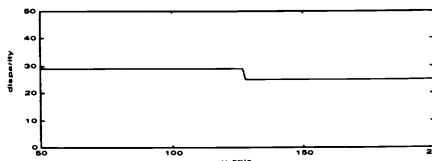
(a)



(b)



(c)



(d)

Figure 9 (a) An intensity profile along the y-direction.

(b) The disparity profile along the y-direction..

(c) Marked regions.

(d) The disparity profile after modification.



## 5. SIMULATION

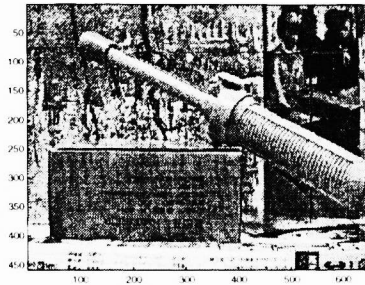
In this section, we do simulation on two image pairs: the boxes-and-pipe pair and the Mocca pair. To show the simulation results, we display the disparity map using different gray levels. The larger the disparity is, the brighter the gray level is. If the disparity map is a feature disparity map, it only shows the disparity value at the feature points. For each image pair, we show the original images, the disparity map after feature matching, the modified feature disparity map, the dense disparity map after intensity matching, and the modified dense disparity map.

Figure 10 shows the simulation result of the “boxes-and-pipe” pair. In Figure 10 (a) (b), there are three objects in the foreground: a box in the middle, a pipe laying on the box, and a box in the right. The background is full of texture. Figure 10(c) and (d) show the feature disparity map and the modified feature disparity map, respectively. Figure 10(e) illustrates the dense disparity map estimated by the intensity-based algorithm presented in Section 4. Figure 10(f) shows the modified dense disparity map. Figure 11 shows the simulation result of the “Mocca” pair. In Figure 11(a) (b), there are three objects in the foreground: a box in the middle, a tea bag in the left, and a cylinder in the right. The background is filled with almost constant intensity. Only a little amount of shading appears in the background. The noise level of the images is large and there is a fixed pattern in each image. Figure 11(c) and (d) show the feature disparity map and the modified feature disparity map, respectively. Figure 11(e) illustrates the dense disparity map. Figure 11(f) shows the modified dense disparity map.

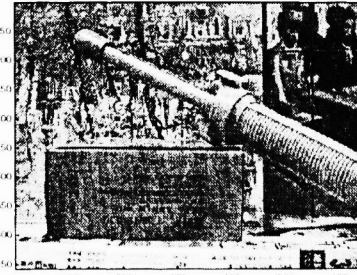
## 6. CONCLUSIONS

In the paper, a stereo correspondence algorithm, combining both feature-based matching and intensity-based matching, is presented. By using the high-curvature points on the intensity profiles as the features, a dissimilarity measurement can be easily defined to compare the shape of the corresponding intensity profiles. The dynamic programming method is then used to do feature matching. With this choice of high-curvature points, the generated sparse feature disparity map can be easily verified by checking the intensity information in between. Then, both SSD method and the dynamic programming method are used alternatively to match the rest regions and generate a dense disparity map. At the last stage, some constraints based on the continuity of intensity and the continuity of disparity are used to further modify the dense disparity map.

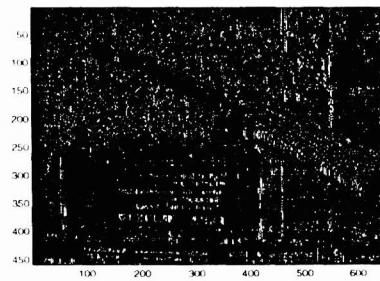
This algorithm provides a simple and fast matching based on the high-curvature points. The simulation demonstrates the capability of this algorithm to handle both scenes with complicated contents and scenes lack of texture.



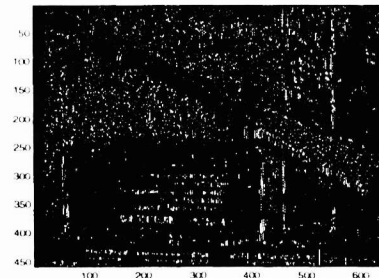
(a)



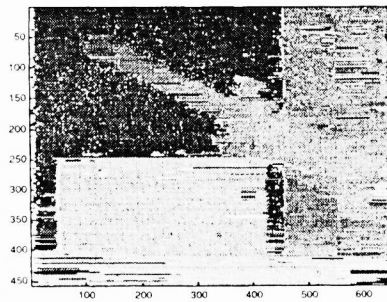
(b)



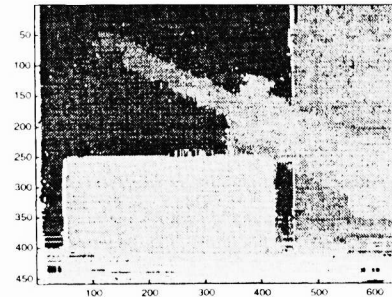
(c)



(d)



(e)

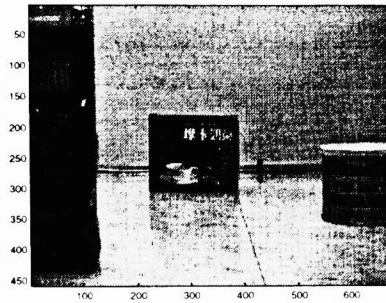


(f)

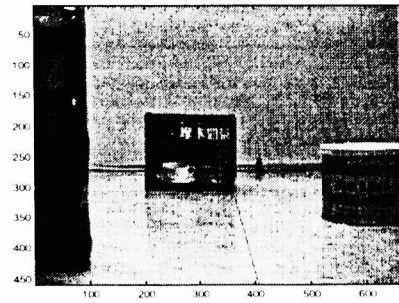
Figure 10 "boxes-and-pipe" pair

(a) left image (b) right image (c) feature disparity map

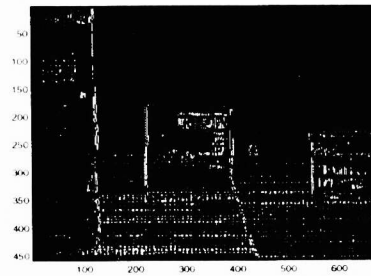
(d) modified feature disparity map (e) dense disparity map (f) modified dense disparity map.



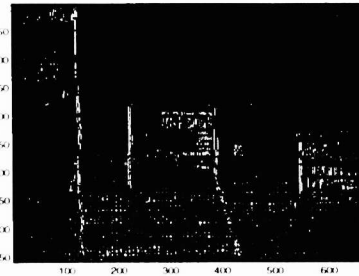
(a)



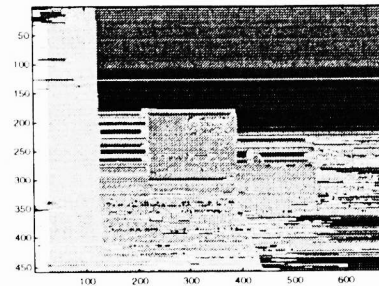
(b)



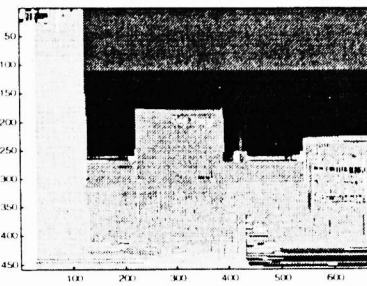
(c)



(d)



(e)



(f)

Figure 11 "Mocca" pair

(a) left image (b) right image (c) feature disparity map

(d) modified feature disparity map (e) dense disparity map (f) modified dense disparity map.

## REFERENCE

1. T. Aach and A. Kaup, "Disparity-Based Segmentation of Stereoscopic Foreground/Background Image Sequences", *IEEE Transactions on Communications*, vol. 42, no. 2/3/4, February/March/April, 1994.
2. H. H. Baker and T. O. Binford, "Depth from edge and intensity based stereo," in *Proc. 7<sup>th</sup> Int. Joint Conf. Artif. Intell.*, Aug. 1981, pp. 631-636.
3. S. Birchfield and C. Tomasi, "Depth Discontinuities by Pixel-to-Pixel Stereo".
4. I. J. Cox, S. L. Hingorani, and S. B. Rao, "A maximum Likelihood Stereo Algorithm", *Computer vision and image understanding*, vol. 63, no. 3, May 1996, pp.542-567.
5. U. R. Dhond and J. K. Aggarwal, "Stereo Matching in the presence of Narrow Occluding Objects Using Dynamic Disparity Search", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 7, July 1995, pp719-724.
6. G. D. Forney, JR., "The Viterbi Algorithm", *Proc. IEEE*, vol. 61, no. 3, March 1973.
7. W. E. L. Grimson, "Computational Experiments with a Feature Based Stereo Algorithm", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-7, no. 1, January 1985.
8. W. Hoff and N. Ahuja, "Surfaces from Stereo: Integrating Feature Matching, Disparity Estimation, and Contour Detection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 2, February 1989.
9. H. H. Jong, "The extraction of High-Curvature Feature Points in Image Surface", Master Thesis, Department of EE, NCTU, Taiwan, June 1998.
10. Julesz, *Foundations of Cyclopean Perception*, The University of Chicago Press.
11. T. Kanade, M. Okutomi, "A Stereo Matching Algorithm with an Adaptive Windows: Theory and Experiment", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 9, September 1994, pp920-932.
12. M. S. Lew, T. S. Huang, and K. Wong, "Learning and Feature Selection in Stereo Matching", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 9, September 1994, pp869-881.
13. H. H. Liu and S. J. Wang, *The representation of Image by Using Curvature Information of Image Surfaces*, Master Thesis, Department of EE, NCTU, Taiwan, June 1997.
14. S. B. Marapane and M. M. Trivedi, "Region-Based Stereo Analysis for Robotic Applications", *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 19, no. 6, November/December 1989.
15. D. Marr and T. Poggio, "Cooperative Computation of Stereo Disparity", *Science*, vol. 194, pp. 283-287, October 1976.
16. Y. Ohta, and T. Kanade, "Stereo by Intra- and Inter-Sacnline Search Using Dynamic Programming", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-7, no.2, March 1985, pp139-154.
17. M. Okutomi and T. Kanade, "A Multiple-Baseline Stereo", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 4, April 1993.
18. L. Robert and O. D. Faugeras, "Curve-based Stereo: Figural Continuity and Curvature".
19. C. Tomasi and Roberto Manduchi, "Stereo Matching as a Nearest-Neighbor Problem", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 3, March
20. J. H. Twu and S. J. Wang, *The reconstruction of Image Surface by Using Curvature Information*, Master Thesis, Department of EE, NCTU, Taiwan, June 1997.
21. Y. Yang, A. Yuille, and J. Lu, "Local, Global, and Multilevel Stereo Matching", *Proceedings: 1993 IEEE Computer Society Conference on CVPR*, pp.274-279.