World Academy of Science, Engineering and Technology
International Journal of Mechanical and Mechatronics Engineering
Vol:2, No:5, 2008

Fast Depth Estimation with Filters

Yiming Nie, Tao Wu, Xiangjing An, and Hangen He

***Abstract*—**Fast depth estimation from binocular vision is often desired for autonomous vehicles, but, most algorithms could not easily be put into practice because of the much time cost. We present an image-processing technique that can fast estimate depth image from binocular vision images. By finding out the lines which present the best matched area in the disparity space image, the depth can be estimated. When detecting these lines, an edge-emphasizing filter is used. The final depth estimation will be presented after the smooth filter. Our method is a compromise between local methods and global optimization.

***Keywords*—**Depth estimation, image filters, stereo match.

I. Introduction

FAST depth estimation is preferred in many vision applications of robots. To get the depth estimation the binocular vision method is especially popular. But most binocular vision algorithms cost too much time, so we focus on an image-processing technique that can fast estimate depth image from binocular vision images.

Given only two calibrated binocular vision images, depth estimation is a tough work.[1] In this work finding the matched area is the kernel issue, and it should consider the global correspondence and the local constraint. There are many global and local methods[2, 3] which can compromise the time cost and result accuracy. Such as dynamic programming[4], global optimization, graph cut and so on.[5]

Our technique takes the disparity space image as the major consideration. In the disparity space image there is the depth information. Different techniques go different ways. Our technique detects the significance short lines which are present the matched area. Each line means a matched area.

Note that in contrast to more difficult theories we do not require complex calculations, just using filters which are easily implement in special purpose hardware, like digital signal processors (DSP) or field programmable gate arrays (FPGA)DSP or image process hardware which will fast gives the result in very short time.[6]

Yiming Nie is a Doctoral candidate from Automation Institute, College of Mechatronical Engineering and Automation, National University of Defense Technology, Changsha, Hunan, P.R. China, 410073 (e-mail: e_ming_e_ming@hotmail.com).

Tao Wu and Xiangjing An are associate professors from Automation Institute, College of Mechatronical Engineering and Automation, National University of Defense Technology, Changsha, Hunan, P.R. China, 410073.

Hangen He is professor from Automation Institute, College of Mechatronical Engineering and Automation, National University of Defense Technology, Changsha, Hunan, P.R. China, 410073.

II. Overview of our Approach

For each line of binocular images, we generate disparity space image, and use Prewitt filter to get the short line from the disparity space image. Use the deep curve to get the rough result. Finally after applying middle value filter, the result is get. The following is the sequence of our method.

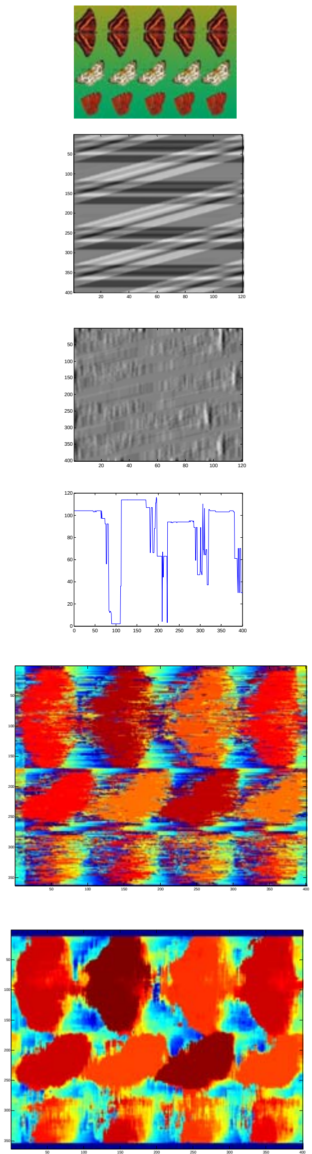

Fig. 1 Overview of our approach

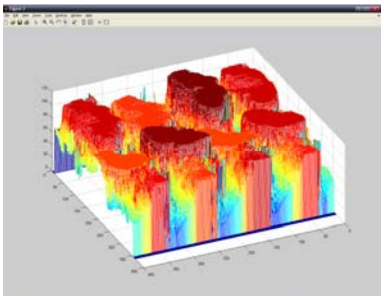

Fig. 2 Three dimension view of result

III. Get the Deep Curve

First we consider a pair of calibrated images. Every line in the left image has a corresponding line in the right image. Two corresponding lines have the same line number. Each pair of corresponding lines could generate a disparity space image by calculating the following formula.

$$I_{\mathrm{DSI}}(n,u,v) = I_l(n,v) - I_r(n,v+u) \tag{1}$$

Where n is the line number, u is the disparity, v is the pixel along the original image line.

The following image is the disparity image, and to be simpler we take $I_l = I_r$, and 2<u<40:

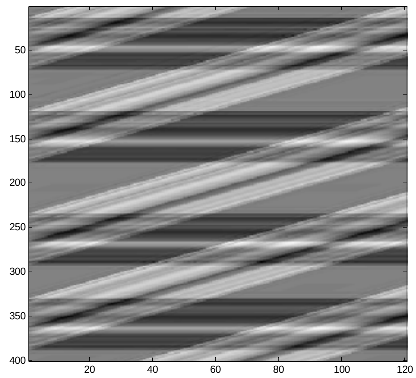

Fig. 3 Disparity space image

Other algorithms don't use the DSI image for comparing directly, because it is the difference of direct pixels, cannot preferably reveal the relativity of corresponding pixels. The other reason is that when extracting depth information from DSI image global optimization should be implemented to find a curve expressing the depth. Results by using local methods would not be as well as global optimization.

In our application, the target objects are planar and perpendicular to the seeing line. So the objects in the disparity space image are some short lines and the line values are about zero.

In the following image the areas in the ellipses are the short lines mentioned above. Those are the matched areas.

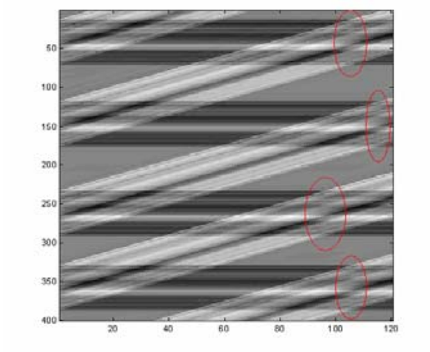

Fig. 4 Appropriate matched areas

In the above image, those marked with ellipses are appropriate matched areas, abscissas of which represent depth.

To simplify, the global optimization curve can be approximately considered as lines jointed together. Thus we can use the local information to find the longest line of each part, which is the eclipse area, and then joint these lines into depth curve.

To identify those lines we use the edge detection method, which also called edge template matching. This is a method of edge detection in image processing which calculates the maximum response of a set of convolution kernels to find the local edge orientation for each pixel.

There are many edge templates to detect the edge, such as Sobel edge detection operator, Roberts operator, Canny operator and Prewitt operator. In our method we take Prewitt operator as the edge detector[7]. Because the edges or the lines we wanted to identify is vertical lines in the image. Other orientation lines are not necessary to identify. So the Prewitt operator is good enough and simply to implementation in practice.

$$H = \begin{matrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{matrix} \tag{2}$$

In practice the result of Prewitt operator convoluting the disparity space image directly is not good. Because there are many outliers with big absolute values affecting the convolution result. It is necessary to cross out the outliers.

And in our experiment setting a proper threshold does a good job to exclude the outliers. This threshold will protect the regions of interest. And the short lines will be identifying easily.

The following image is the result of convolution:

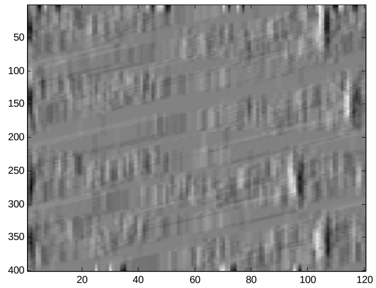

Fig. 5 Result of convolution

From the image the Prewitt operator not only identify the edges but also automatically joint the very short lines to longer lines. This achieves the global optimization.

In fact the above steps equals the Sum of Absolute Differences method, which is fast implement in practice.[]

$$\sum_{u,v} | I_1(u,v) - I_2(u+d,v) | \tag{3}$$

Because the Prewitt operator sums the disparity space image in this way:

$$\sum_{v-1,v,v+1} I_{DSI}(n,u-1,v) - \sum_{v-1,v,v+1} I_{DSI}(n,u+1,v) \tag{4}$$

And after that, we find the maximize value in each line of the result to form the deep curve for the chosen line.

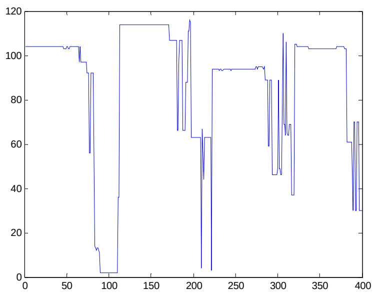

Fig. 6 Deep curve

IV. Refine the Result

Each line of the original image after the process above will get deep information. And all the lines will combine as the deep image as follows:

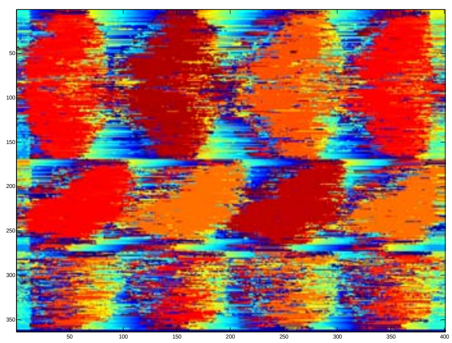

Fig. 7 Rough result

This image is not good. There are lots of noisy dots. Because the match area is get from the Sum of Absolute Differences of the intensity of the image. There will be noisy dots.

To solve this problem we also take filters. The middle value filter will solve this problem. From the signal and system theory, the noisy is the high frequency energy. What we need to do is to minimize the high frequency energy and leave the middle and low frequency energy. The middle value filter is low pass filter and the different form other low pass filter is it can protect the contrast information.

We get the deep line information form the horizontal lines. If we use horizontal information to filter the noise is not a good idea. But use vertical information is a good one. The vertical information is not used and in this place will take all the information from the original image. This will get a good result.

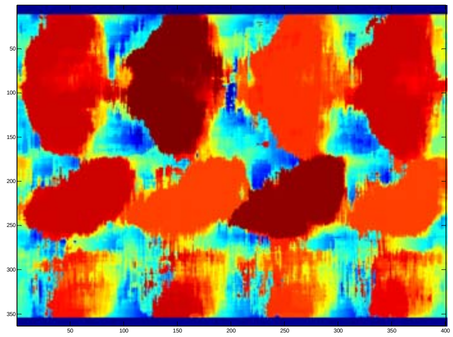

Fig. 8 Final result

V. Experimental Results and Conclusion

Form our method some experiment result is as following:

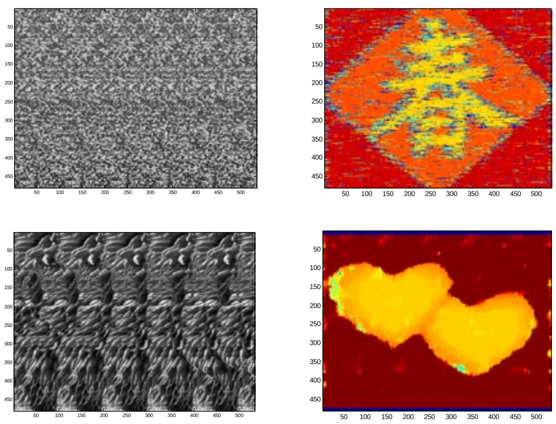

Fig. 9 Final results for random stereo image

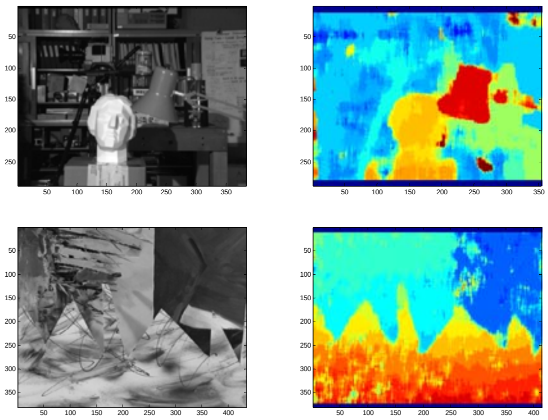

Fig. 10 Final results using the Middlebury datasets

The method we proposed is the same as some other methods in functional. But in the calculation structure this method is based on filters which are easily implement in special hardware to fast give result.

REFERENCES

[1] D. S. a. R. S. a. R. Zabih, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," Daniel Scharstein, Richard Szeliski, and Ramin Zabih. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. In Proceedings of the IEEE Workshop on Stereo and Multi-Baseline Vision, Kauai, HI, Dec. 2001., 2001.

[2] P. R. a. E. H. a. J. Biemond, "Synthesis of multi viewpoint images at non-intermediate positions," P.A. Redert, E.A. Hendriks, and J. Biemond, Synthesis of multi viewpoint images at non- intermediate positions, Proceedings of ICASSP Munchen, Germany, pp. 2749-2752, 1997., 1997.

[3] A. D. a. P. T. a. S. R. a. R. Cipolla, "Combining Single View Recognition and Multiple View Stereo for Architectural Scenes," pp. 268--274.

[4] G. V. M. a. M. V. a. M. P. a. L. V. Gool, "A Hierarchical Symmetric Stereo Algorithm Using Dynamic Programming," G. Van Meerbergen, M. Vergauwen, M. Pollefeys, and L. Van Gool. A Hierarchical Symmetric Stereo Algorithm Using Dynamic Programming, International Journal of Computer Vision, 47, 275-285, 2002., 2002.

[5] M. B. a. D. B. a. G. Hager, "Advances in computational stereo," BROWN M. Z., BURSCHKA D., HAGER G. D.: Advances in computational stereo. PAMI 25, 8 (Aug. 2003), 993--1008. 1, 2003.

[6] J. a. K. R. Woetzel, "Real-time multi-stereo depth estimation on GPU with approximative discontinuity handling."

[7] http://en.wikipedia.org/wiki/Prewitt.