Real Time Rectification Using Differentially Encoded Lookup Table

Deuk Hyun Park
Department of Electrical and computer engineering
Sungkyunkwan University
Suwon, Korea
+82-31-290-7200
dhpark@vada.skku.ac.kr

Hyoung Seok Ko
Department of Mobile Systems Engineering
Sungkyunkwan University
Suwon, Korea
+82-31-290-7200
skyodin@skku.edu

Jae Gon Kim
Department of Mobile Systems Engineering
Sungkyunkwan University
Suwon, Korea
+82-31-290-7200
gon9031@skku.edu

Jun Dong Cho
Department of Electrical and computer engineering
Sungkyunkwan University
Suwon, Korea
+82-31-290-7127
jdcho@skku.edu

ABSTRACT

In this paper, we propose a new real time rectification technique based on the compressed lookup table. To compress the lookup table we adopt a differential encoding. As a result, we successfully constructed the rectification with obtaining the compression ratio of 73% so as to fulfill real-time requirement (i.e., 40 fps at 74.25Mhz). Furthermore, our result on performance is comparable to the result of [17] that obtains 85fps at 90MHz for 640x512 images.

Categories and Subject Descriptors

I.4.1[**Image Processing and Computer Vision**]: Digitization and Image Capture – *Camera calibration, 3D rectification, Real-time and embedded systems, optimization, simulation*

General Terms

Algorithms, Performance, Design, Experimentation, Human Factors.

Keywords

Real time, Rectification, Data compression, Differential encoding, Lookup table.

1. INTRODUCTION

The most critical part of 3D image processing is to obtain distance information of objects in image that can be obtained by computing disparity of corresponding pair of pixel of two images. Because searching the corresponding pair of pixel in two images takes much time, to reduce searching time, real time image rectification is an essential step in stereo vision. Given a pair of stereo images, rectification determines a transformation of each image plane such that pairs of conjugate epipolar lines become collinear and parallel to one of the image axis [1, 2, 3]. The rectified images can be obtained by rotating the original images [4, 5]. Rectification is essential for accurate disparity estimation prior to stereo matching.

The disparity of a pixel is the space between the pixel in the left image and its matching pixel in the right image. The process of ruling the matching pixel is referred to as stereo matching. The disparity of each pixel in paired images designates a disparity map [6]. Stereo matching algorithms to compute depth map can take advantage of the epipolar constraint and reduce the search space from two dimensions to one dimension [7,8].

There are two kinds of implementation method of rectification. One is rectification computing every pixel of two images. The other is using lookup table. The former needs smaller memory, but more time to compute than the latter. Thus the former is not suitable for real-time rectification**.** However, as the resolution of image increases, the real time rectification becomes a challenging task. The previous works on real time rectification use excessive memory with high resolution of image. Also, lookup table occupies very large memory space when an image is of high quality [9].

In "abstraction Lookup Table method" in [9], lookup table does not hold all information for every image pixel to reduce the size of lookup table. For 64x64 image, the lookup table size is 20Kbyte. To decide the missing pixels of rectified image, interpolation is performed. However the interpolation causes distortion in rectified image and delay for computing interpolation. Thus, the method is not suitable for real time rectification. Therefore, in this paper, we propose a real time rectification technique using a "differentially coded lookup table" to reduce the size of lookup table.

This paper is organized as follows. Section 2 will present an overall flow of our rectification strategy. Section 3 will discuss on the hardware structure to realize the rectification in real time. Section 4 will present the experimental results and finally, Section 5 will draw a conclusion.

ICUIMC '11, February 21-23, 2011, Seoul, Korea

2. OUR RECTIFICATION STRATEGY

Rectification has been studied broadly in stereo matching systems. Simple radial and tangential distortion models modify the misalignment of lenses with optical axes for distortion correction [17]. Most previous works on rectification have focused on regulating the parameters of the distortion model. However, Gribbon [11] proposed a calibration method based on a lookup table with a built-in memory that can be executed in an FPGA. This idea has been used to correct the single camera misalignment.

In this paper, we use a calibration map for instantaneous rectification for stereoscopic camera. Figure 1 shows the overall flow of our proposed rectification method. The calibration procedure determines displacements of pixels in the distorted image from their ideal position. The rectification procedure displaces them from the distorted image to the correct position.

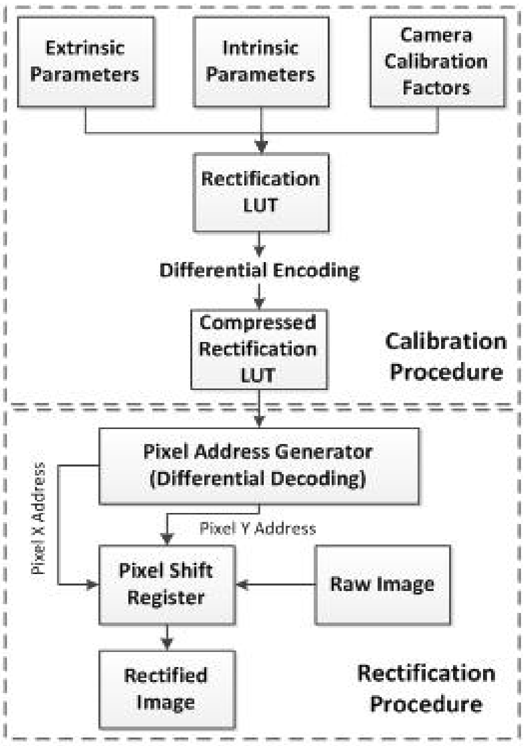


Figure 1. Overall flow of our rectification strategy

The calibration map is a kind of lookup table in the memory to correct images in real-time. Our calibration map method shows accurate disparity estimation from the rectified images directly.

Lookup tables are results of computing camera calibration with intrinsic and extrinsic parameters [9,12]. The real-time rectification method based on the above-mentioned lookup table can be implemented with compact hardware. The initial rectification lookup table can be too large to fit into memory on FPGAs.

We noticed that the differentially calibrated position values of neighbouring pixels are one or two. Therefore, using differential encoding, the size of lookup table can be reduced

3. OUR RECTIFICATION STRUCTURE

Structure of our rectification module is shown in Figure 2. The module consists of lookup table address counter, compressed lookup table, and pixel shifter. Input images consist of right and left images. Lookup tables can be constructed by using software as a pre-processing. Because the two stereoscopic cameras are fixed, the parameters in the lookup table are unchanged [13]. Therefore, in rectification process it is not necessary to calculate for every pixel after lookup table is constructed.

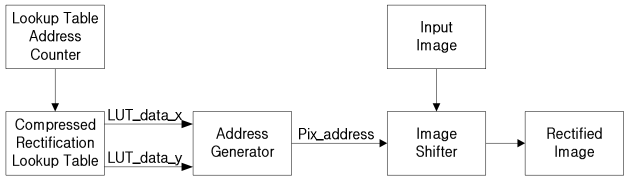


Figure 2. Structure of rectification module

Lookup table has the same number of memory address as many as rectified image memory. The memory address of rectified image corresponds to one of lookup table. Figure 3 shows the overall flow of rectification data transfer process.

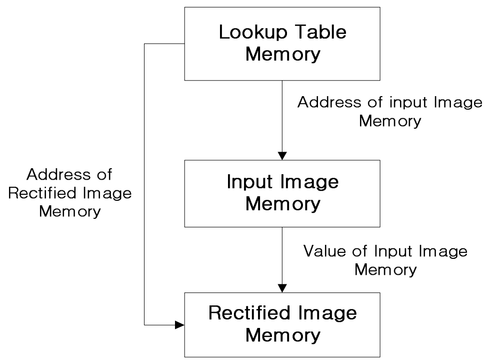
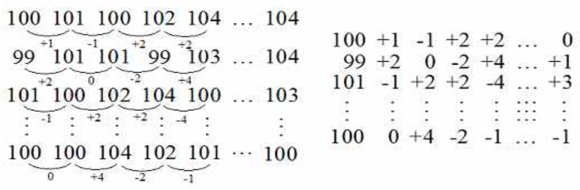


Figure 3. Flow of data transfer process

As shown in Figure 3, lookup table contains address of input image memory. In input image memory, the value located at address in lookup table is transferred to rectified image memory. If we rectify the high resolution images, because the lookup table contains value on every pixel, the size of lookup table becomes excessive. In this case, external memory might be needed to store this large lookup table. To load the large lookup table to the rectification module, it also results in delay increase. Therefore, lookup table should occupy small memory so as to be loaded in rectification module in real time. In this paper, we compress the lookup table by adopting differential encoding. An example is shown in Figure 4.



(a) (b)

Figure 4. Differential encoding: (a) before differential encoding, (b) after differential encoding

Differentially encoded value is the difference between previous and present value. In the differential lookup table, the first column values are original values, and other values are differences between previous and present value [14]. This encoding algorithm is not complex in both encoding and decoding, and thus implementation overhead does not influence the real-time processing.

To decode the values of differential encoded lookup table, as shown in Figure 5, address generator is needed. It consists of two accumulators, one multiplier and one adder. As the accumulator pluses differences in differential encoded lookup table, the original value for lookup table is restored [15,16].

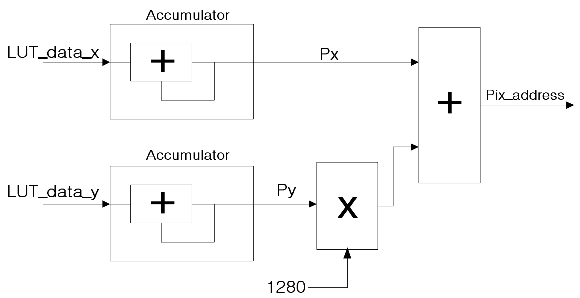


Figure 5. Address generator

4. EXPERIMENTAL RESULTS

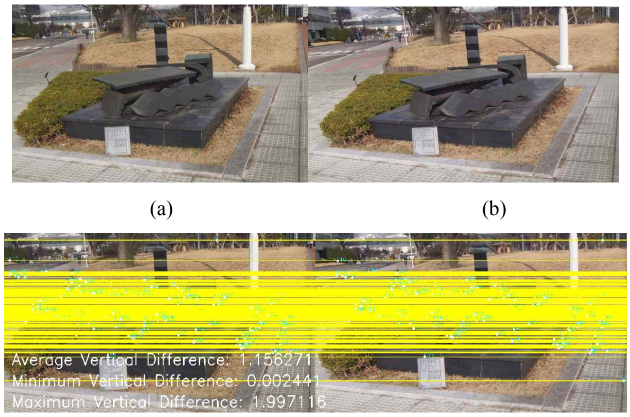
We implemented our rectification module in Xilinx FPGA Vertex5. Initially, compressed lookup table is stored in external flash ROM. Next, when power is turned on, lookup tables are loaded into FPGA. Resolution of input images we used is 1280 x 720. Images are acquired by digital cameras through HDMI. Before performing differential encoding, the size of lookup table is 4.8M bytes, and after performing differential encoding, it was 1.3 M bytes.

Table 1. Comparison of Lookup Tables in Size

Kind of Lookup Tables	Size	Compression ratio
Uncompressed	4.8 M bytes	73%
Compressed	1.3M bytes	

Rectified image pair is generated with 40 fps at 74.25MHz clock speed. Furthermore, our result on performance is comparable to the result of [17] that obtains 85fps at 90MHz for 640x512 images. Table 1 shows the compression ratio of lookup table.

Input image pair and rectified image pair are illustrated in Figures 6 and 7, respectively. They are input images and rectified images with corresponding (i.e., matching) points and epipolar lines. Corresponding points imply that they are same point in real view [18]. The yellow lines in image (c) of Figures 6 and 7 are epipolar lines. Epipolar lines are generated by connecting the corresponding points on input images [19].

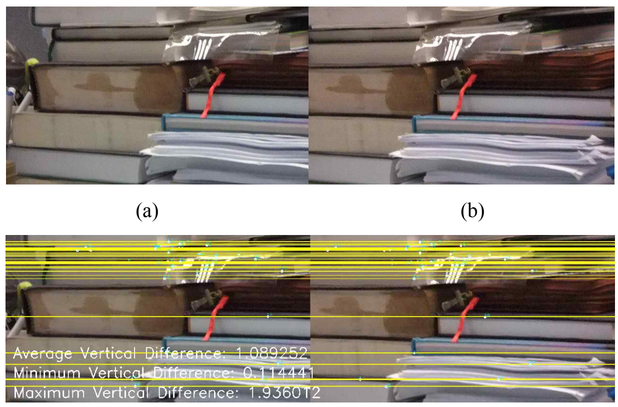


(a) (b)

(c)

Figure 6. Input image pair: (a) left, (b) right, (c) correspondence between rectified image pair

Because the two images of Figure 6(a) and (b) have vertical lines with different x coordinates mainly, more corresponding points must be found for rectification than the images that have horizontal lines mainly. Whereas, input images of Figure 7 have corresponding horizontal lines mainly, thus less corresponding points are founded than the input images of Figure 6. There are 185 corresponding points in image (e) of Figure 6 and 39 corresponding points in image (e) of Figure 7.



(a) (b)

(c)

Figure 7. Input image pair: (a) left, (b) right, (c) correspondence between rectified image pair

Note that processing speed of computing the two image pairs is same because the resolution and lookup table being used are the same. The only difference is the number of corresponding points.

5. CONCLUSIONS

In this paper, we showed a new real time rectification using compressed lookup table. From the test result, our differential encoding was successfully applied to rectification to reduce the size of lookup table so as to be performed in real-time (i.e., 40fps at 74.25 MHz clock). Lookup table compression ratio we obtained is 73%. Furthermore, our result on performance is comparable to the previous result. Also, our lookup table is constructed by software as a pre-processing, and thus complicated calculation of processing in camera system can be avoided. The rectification step determines the next stage of image processing performance. Finally we notices that as the size of lookup table can be reduced, the remaining memory resources in FPGA can be used for other purpose to enhance performance.

6. ACKNOWLEDGEMENTS

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology(S-2010-1119-000).

7. REFERENCES

[1] Forsyth, D. and Ponce, J. 2003. *Computer vision: a modern approach*, Prentice Hall.

[2] Chengqing, T. and Xuejing, D. 2010. A Rectification Algorithm for Distorted Images from Cone Surface. *Wireless Communications Networking and Mobile Computing,* (Sep.2010), 1-4.

[3] Ayache, N. and Hansen, C. 1988. Rectification of images for binocular and trinocular streovision. *Proc of International Conference on Pattern Recognition*, vol.1. (Nov.1988), 11-16.

[4] Fusiello, A., Trucco, E. and Verri, A. 2000. A compact algorithm for rectification of stereo pairs. *Machine Vision and Applications,* vol.12. (Mar.2000), 16-22.

[5] Pollefeys, M., Koch, R. and Van Gool, L. 1999. A simple and efficient rectification method for general motion. *Proc IEEE Computer Vision*, vol.1. (Sep. 1999), 496-501.

[6] Brown, M. Z., Burschka, D. and Hager, G. D.2003. Advances in computational stereo. *IEEE Trans. Pattern Anal. Mach. Intel.*, vol. 25 (8). (Aug. 2003), 993–1008.

[7] Faugeras, O. 1993. *Three-Dimensional Computer Vision: A Geomatric Viewpoint*, MIT Press.

[8] Hartley, R. and Zisserman, A. 2003. *Multiple view geometry in computer vision*, Cambridge University Press.

[9] Kjaer-Nielsen, A., Jensen, L., Sorensen, A.S., Kruger, N. A Real-Time Embedded System for Stereo Vision Preprocessing Using an FPGA. 2008. *Reconfigurable Computing and FPGAs*. (Sep. 2008), 37-42.

[10] Zhang, Z. 2000. A flexible new technique for camera calibration. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 11. (Nov. 2000), 1330-1334.

[11] Gribbon, K. T., Johnson, C. T. and Bailey, C. T. 2003. A real-time FPGA implementation of a barrel distortion correction algorithm with bilinear interpolation. *In Proceedings of the Image and Vision Computing New Zealand*. (2003), 408-413.

[12] Fusiello, A., Trucco, E. and Verri, A. 2000. A compact algorithm for rectification of stereo pairs. *Machine Vision and Applications* vol.12. (Mar. 2000), 16-22.

[13] Lusk, J. A. and Nutter, B. 2009. Automated 3-D reconstruction of stereo fundus images via camera calibration and image rectification. *Computer-Based Medical Systems*. (Aug. 2009), 1-7.

[14] Saywood, K. 2000. *Introduction to Data Compression*, Morgan Kaufmann Publishers.

[15] Ciletti, M. D. 2002. *Advanced Digital Design with the VERILOG HDL*, Prentice Hall.

[16] Palintkar, S. 2003. Verilog *HDL: a guide to digital design and synthesis*, Prentice Hall.

[17] Vancea, C. and Nedavschi, S. 2007. LUT-based Image Rectification Module Implemented in FPGA. *Intelligent Computer Communication and Processing*. (Sep. 2007), 147-154.

[18] Kang, Y.S., Lee, C. and Ho, Y.S. 2008. An Efficient Rectification Algorithm for Multi-View Images in parallel Camera Array. *3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video*. (May. 2008), 61-64.

[19] Hartley, R. I. 1999. Theory and practice of projective rectification. *International Journal of Computer Vision*, vol.35, no.22. (Nov. 1999), 115-127.