ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645
Indian Journal of Science and Technology, *Vol 8(S8), 397-404, April 2015* DOI: 10.17485/ijst/2015/v8iS8/70529

Comparison of Hardware Design Complexity to Extract Depth Map from Stereo Images

Ji-Yong, Lee[1], Young-Hyung, Kim[2] and Yong-Hwan, Lee[3*]

[1]Department of Electronics Engineering, Kumoh National Institute of Technology, Gumi, 730-701, South Korea
[2]Department of IT Convergence, Kumoh National Institute of Technology, Gumi, 730-701, South Korea
[3]School of Electronics Engineering, Kumoh National Institute of Technology, Gumi, 730-701, South Korea; yhlee@kumoh.ac.kr

Abstract

Computer vision technology has been researched in the field of mobile robots, intelligent vehicle navigation aids, obstacle avoidance and many others. We implement fast and accurate census transform algorithm in software and hardware for computer vision to extract depth map from different two images. We have designed in C language and Verilog hardware description language, verified by Modelsim, and implemented on FPGA for real-time processing capability. We have converted the census transform algorithm to hardware structure to compute depth map for two images in parallel processing. The resultant system extracts the depth information within search-range of 30 from 240 x 320 image using the window size of 3by3, 5by5 and 7by7, respectively.

Keywords: Depth Map, Image Matching, Stereo Vision

1. Introduction

Stereo vision automates distance extraction of a human visual system in computer vision area. As in Figure 1, the views of thing in left and right eyes are different. The principle of this algorithm is similar to the binocular disparity.

Figure 1. Principle of binocular disparity.

Stereo matching is one of the most active research areas in computer vision because of its increasingly widespread applications that require three-Dimensional (3D) depth information[1]. The basic steps for depth information of the two images are the feature extraction, stereo matching, and distance computation from parallax. The most important factor is matching element. Calculation of depth information by the stereo vision is processed in two steps. The first step is to determine whether each of the selected points watch to the reference point. The second step is to obtain the depth information for all the points of the image by using an interpolation and describe a three-dimensional space[2]. Figure 2 shows a flow of stereo matching.

2. Stereo Matching

Stereo matching is a method of extracting the corresponding points by finding the time difference information from

*Author for correspondence

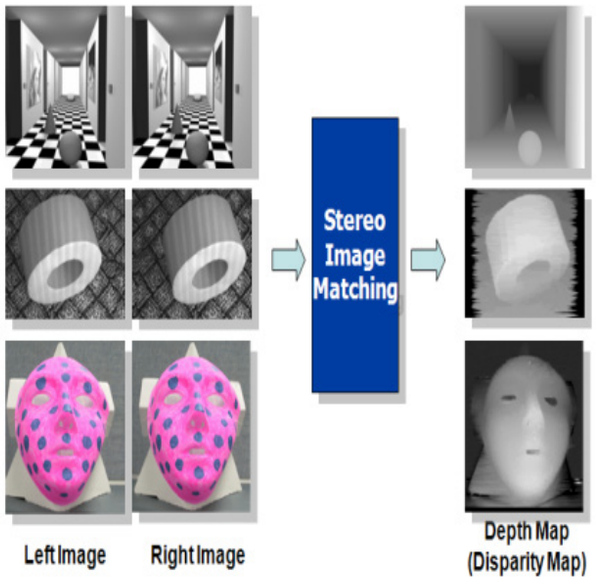

Figure 2. Stereo image sets with ground truth.

the left and right images to obtain three-dimensional image information using two cameras. Either left image or right image is used as a reference. If points from one image located in the same line of another image, distance information can be obtained by computing the difference of coordination in horizontal direction between a reference point and the corresponding point[3,4]. Fig 3 illustrates process of finding matching points of horizontal direction along epipolar-line in left and right image after rectification process.

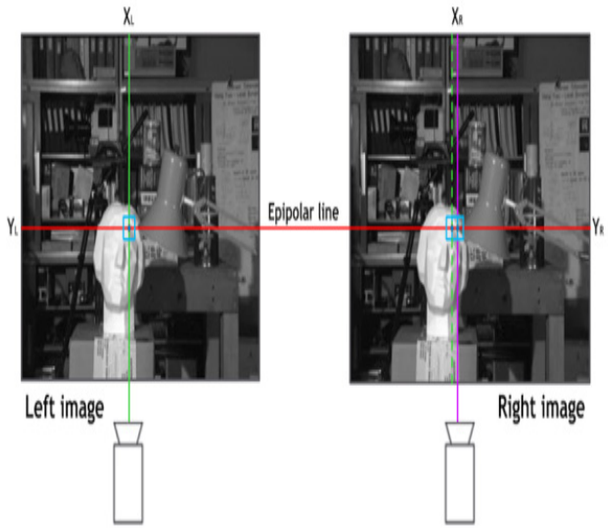

Figure 3. The process of finding a match point in epipolar line.

In contrast to the global approaches, the local approaches find corresponding points between a pair of stereo image on the epipolar-line by computing correlations of windows within a fixed disparity search range. The local algorithms have less computational complexity than the global algorithms. For these reasons, the local approach is widely used in real-time embedded applications[5,6].

2.1 Census Transform Algorithm

The census transform is a non-parametric local transform that characterizes a window as a sequence of bit streams[7–9].

2.1.1 Hamming Weight

Hamming-weight compares the left image with right image by the unit of 5by5 window. If a window is set to 5by5 as in Figure 4, (2, 2) of the pixel value is compared from (0, 0) to (4, 4) respectively to produce Hamming-weight. The resultant Hamming-weight of each pixel has a value of 1 or 0.

Figure 4. Hamming weight.

2.1.2 Hamming Bits

Hamming bits are calculated from left and right Hamming-weight. Each pixel in same position in left and right windows is compared. If two pixels are equal, hamming bit is set to 1 otherwise to 0. Figure 5 illustrates a Hamming bit obtained from the left and the right Hamming-weight by calculating each pixel. The number of 1's in Hamming bits means that two images match.

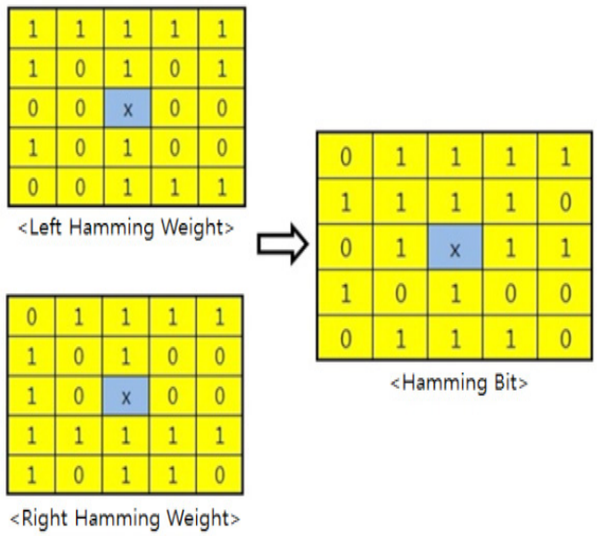

Figure 5. Calculation of Hamming Bit.

2.1.3 *Hamming Distance*

Hamming-distance is obtained in the left and right Hamming bits. In Figure 6, Hamming-distance in search range can be measured by comparing a left Hamming bit with the right Hamming bits. The number of 1's in a Hamming-distance means that the depth map set.

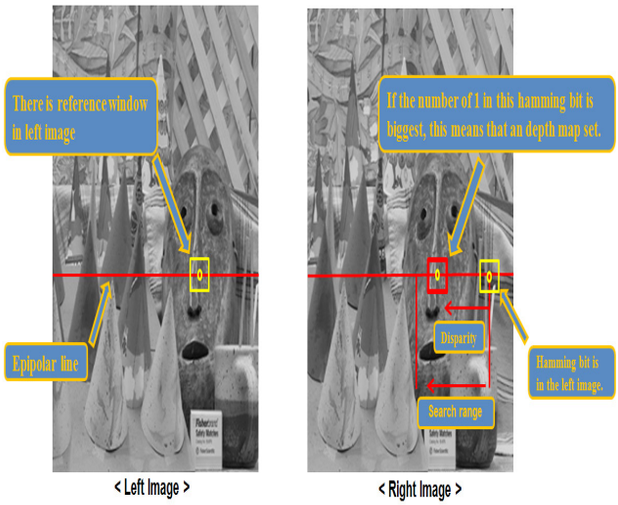

Figure 6. Hamming distance.

3. Hardware Structure

Flow chart of census transform algorithm is shown in Figure 7. The reference pixel in left image for obtaining the depth information of a point with respect to the search range is fixed and candidate pixel in right image is repeated in search range.

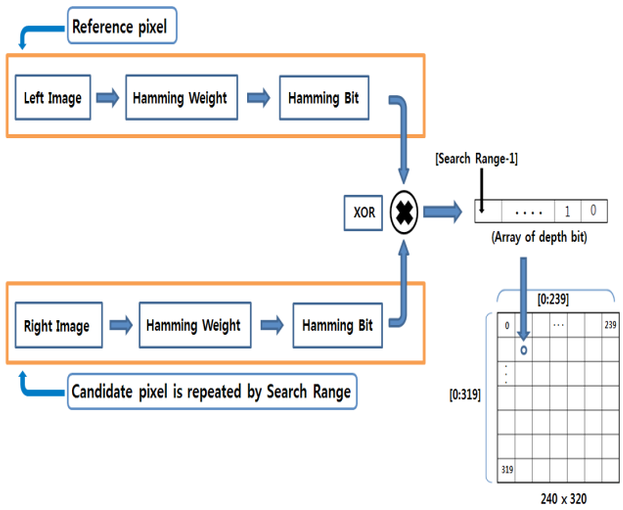

Figure 7. Flow chart of census transform algorithm.

Figure 8 is a combination circuit structure for parallel processing. When making 3by3 Hamming bits in 1 cycle, (1, 1) center pixel is saved in a register for a 3by3 window and (0, 1) and (2, 1) pixel is compared with center pixel for a depth information. (0, 0), (1, 0) and (2, 0) in 2 cycle and (0, 2), (1, 2), (2, 2) in 3 cycle is compared with center pixel.

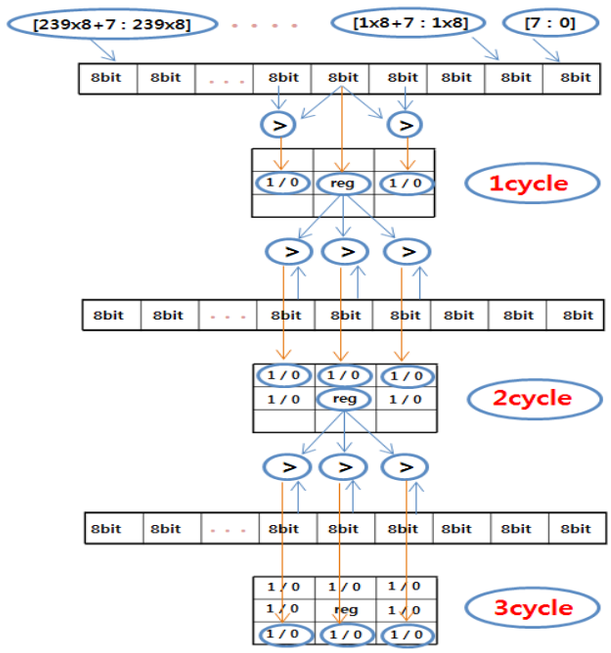

Figure 8. Parallel processing for combination circuit.

We designed a hardware structure to the Verilog hardware description language by using census transform algorithm. The advantage of the Verilog hardware description language is to enable the parallel processing. We compare the results in C language and Verilog to calculate the performance improvement. When calculating a 3by3 window Hamming-bit in C language, 8 pixels except for a center pixel are required to 8 cycles. The combination circuits for parallel processing in hardware structure can process multiple pixels in single clock. The proposed architecture is designed as shown in the Figure 9. Three lines of vertical axis in the block RAM are read to horizontal axis with 240x8 data lines for 3by3 window. First, N+1 line is read for center pixels of 3by3 window. Then, N and N+2 lines are read for comparison to process 3by3 windows simultaneously.

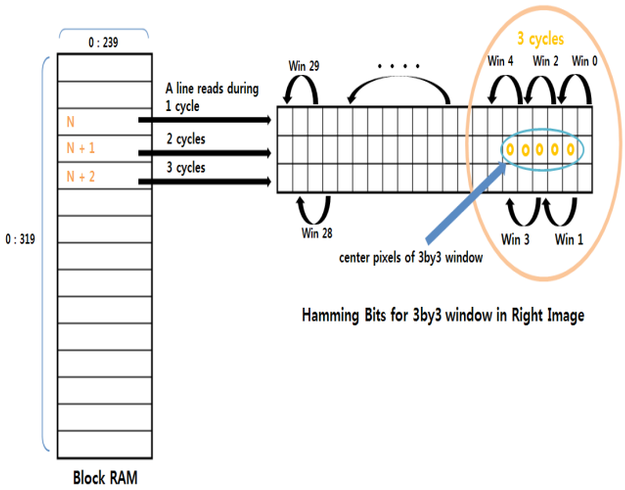

Figure 9. Parallel processing in Hamming Bits.

Therefore, a Hamming-bit is calculated only in 3 cycles. We have made five Hamming bits during 3 cycles. If five Hamming bits in search range of 30 during 3 cycles are made simultaneously, a depth map can be made during 18 cycles.

4. Comparison Results

In order to evaluate the performance of the census transform algorithm, we have designed the C language and Verilog hardware description language, verified by Modelsim, and implemented for real-time processing capability on FPGA and Hardware implementation and verification of census transform algorithm was carried out by the process of Figure 10.

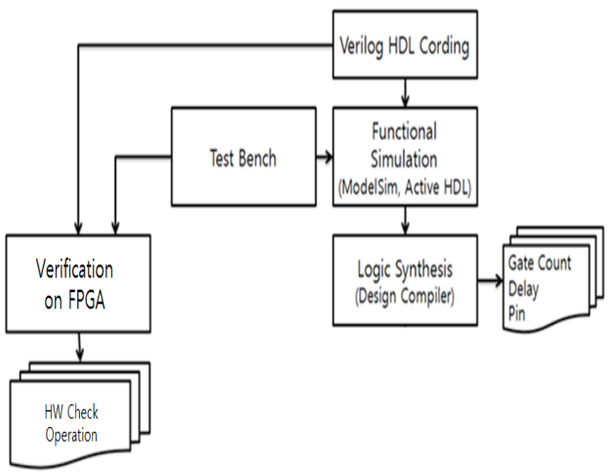

Figure 10. Parallel processing in Hamming Bits.

First, the census transform for matching accuracy in search range 30 was measured varying the window size. Figure 11 shows the matching accuracy of the algorithm. As the window size increase, the matching accuracy is improved but the operating speed is decreased. Therefore, window size is the most dominant factor that effects on the accuracy and the operating speed.

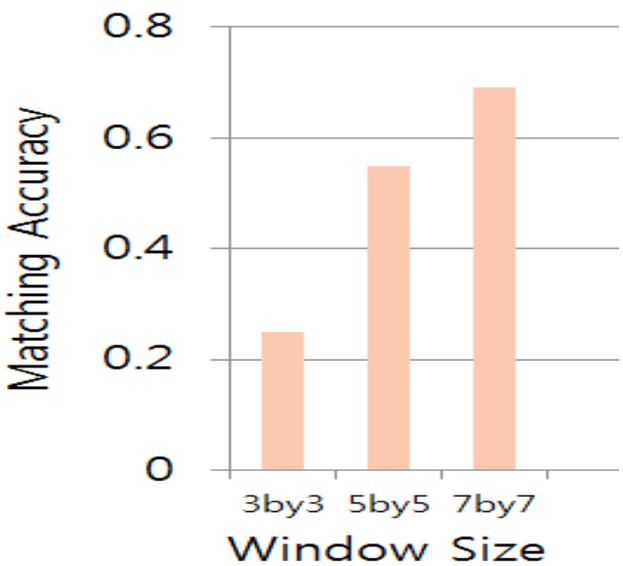

Figure 11. Matching accuracy of window size for census transform.

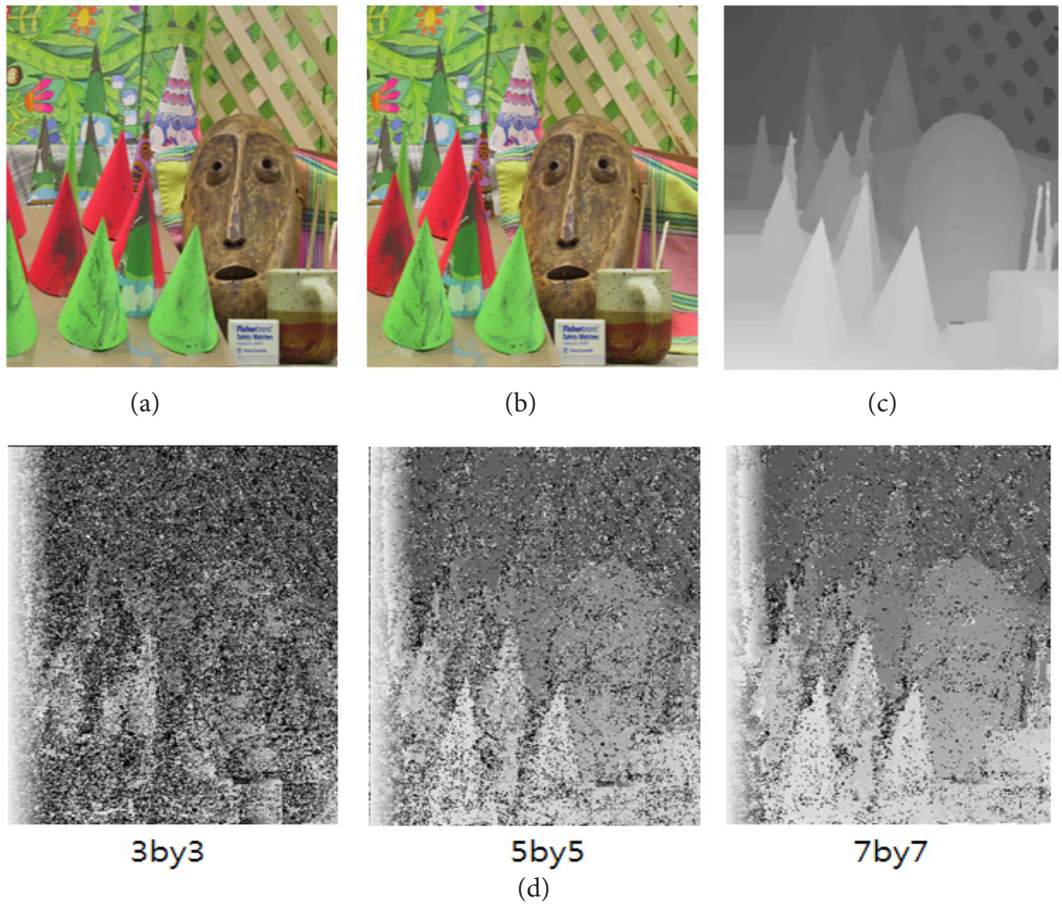

Figure 12. Cone Stereo Image from Middlebury Benchmark10: (a) Left Image; (b) Right Image; (c) Ground Truth Image; (d) Resultant Images for each Window Size.

Figure 12 shows resultant images about matching accuracy.

Second, we implemented 3 hardware structures each with window size of 3by3, 5by5 and 7by7 to compute performance. To process real-time for census transform from two images, we have applied the structure of Figure 8, 9. In search-range 30, parallel processing of the search-range is set to 3, 5 and 10. Table 1 and 2 shows performance comparisons about each parallel processing.

Table 1. C Language performance comparisons

C Language			
Window Size	**Maximum Speed**	**Maximum Frequency**	**Frames / Second**
3by3	0.813[s]	1.23[Hz]	1.29[fps]
5by5	1.33[s]	0.75[Hz]	0.76[fps]
7by7	1.96[s]	0.51[Hz]	0.51[fps]

Table 2. Verilog HDL performance comparisons in parallel processing of 3, 5 and 10 in search range of 30

Parallel Processing = 3			
Window Size	**Maximum Speed**	**Maximum Frequency**	**Frames / Second**
3by3	17.98[ns]	56.61[MHz]	23.92[fps]
5by5	22.06[ns]	45.33[MHz]	12.16[fps]
7by7	22.60[ns]	44.24[MHz]	8.703[fps]
Parallel Processing = 5			
Window Size	**Maximum Speed**	**Maximum Frequency**	**Frames / Second**
3by3	19.92[ns]	50.2[MHz]	37.88[ns]
5by5	22.86[ns]	43.74[MHz]	20.92[fps]
7by7	22.14[ns]	45.17[MHz]	15.95[ns]
Parallel Processing = 10			
Window Size	**Maximum Speed**	**Maximum Frequency**	**Frames / Second**
3by3	21.22[ns]	47.13[MHz]	50.76[fps]
5by5	23.08[ns]	43.33[MHz]	30.12[fps]
7by7	24.48[ns]	40.19[MHz]	20.88[fps]

Third, in order to verify the proposed hardware architecture, we implemented the census transform on FPGA. Figure 14 shows each synthesis results and Figure 15 is the resultant image in window size 7by7 on TFT-LCD.

Table 3. Parallel processing 3 for window size 3by3 in search range of 30

Parallel Processing	**3**	**5**	**10**
Window Size	**3by3**	**3by3**	**3by3**
Slice resisters	299	307	342
Slice LUTs	41,863	64,037	120,205
Occupied slices	21,312	32,465	60,929
Logic	41,863	64,037	120,205
Block Ram	146	146	146

Table 4. Parallel processing 3 for window size 5by5 in search range of 30

Parallel Processing	3	5	10
Window Size	5by5	5by5	5by5
Slice resisters	327	335	375
Slice LUTs	100,575	159,538	307,137
Occupied slices	51,179	80,951	159,239
Logic	100,641	159,538	310,512
Block Ram	146	146	146

Table 5. Parallel processing 3 for window size 7by7 in search range of 30

Parallel Processing	3	5	10
Window Size	7by7	7by7	7by7
Slice resisters	363	382	410
Slice LUTs	193,934	306,925	586,671
Occupied slices	97,001	153,499	300,283
Logic	193,934	306,925	603,823
Block Ram	146	146	146

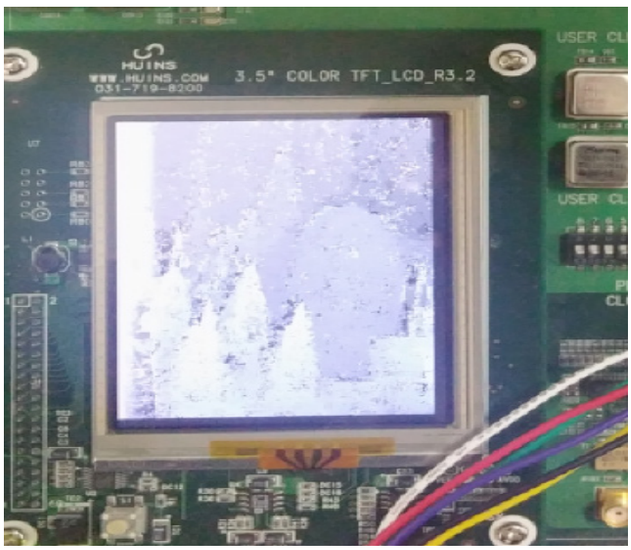

Figure 15. Resultant image implemented on FPGA.

5. Conclusion

In this paper, both C Language and Verilog hardware description language by using census transform algorithm for real-time stereo vision system have been implemented on C simulator and FPGA. C language is very slow and cannot be used for real-time environment. The hardware implementation is accurate and fast enough for real-time applications. We are going to study new census transform-based fast and accurate matching technique to obtain depth information for fast real-time.

6. Acknowledgment

This paper was supported by Research Fund, Kumoh National Institute of Technology.

7. References

1. Szeliski R. Computer vision: algorithms and applications. New York: Springer-Verlage; 2010.
2. Trucco E, Verri A. Introductory techniques for 3-D computer vision. Prentice Hall: 1998.
3. Kim JS. FPGA implementation of matching algorithm for real time stereo vision system.
4. Klette R, Schluns K, Koschan A. Computer vision: three-dimensional data from images. Springer; 1998. p. 129–226.
5. Ok SH. A hardware-friendly modified adaptive support-weight with a robust disparity search range estimation.
6. Zucheul L, Khoshabeh R, Juang J, Nguyen TQ. Local stereo matching using motion cue and modified census in video disparity estimation. 2012 Proceedings of the 20th European Singnal Processing Conference (EUSIPCO); 2012. p. 1114–8.
7. Zabih R, Woodfill J. Non-parametric local transforms for computing visual correspondence. Proceedings of the third European conference on Computer Vision (Vol. II); Stockholm, Sweden: 1994.
8. Zinner C, Humenverger M, Ambrosch K, Kubinger W. An optimized software-based implementation of a census-based stereo matching algorithm. Proceedings of the 4th International Symposium on Advances in Visual Computing; Las Vegas, NV: 2008.
9. Humenberger M, Zinner C, Weber M, Kubinger W, Vincze M. A fast stereo matching algorithm suitable for embedded real-time systems. Comput Vis Image Understand. 2010; 114:1180–202.
10. Scharstein D, Szeliski R. Hight-accuracy stereo depth maps using structured light. Proceedings, 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol.1; 2003. p. I-195–I-202.