Fast Census Transform-based Stereo Algorithm using SSE2

Young Ki Baik †, Jung Ho Jo, Kyoung Mu Lee ‡

† : Seoul National University, hyunxx00@snu.ac.kr
‡ : Seoul National University, kyoungmu@snu.ac.kr

Abstract In this paper, we propose a census transform-based fast stereo algorithm. Most conventional fast stereo algorithms are based on SAD (Sum of Absolute Difference) or NC (Normalized Correlation). However, SAD is often sensitive to noise or illumination changes. Image matching by census transform is known to be robust to radiometric distortion, since the differences in gain and bias between two images will not affect the ordering of pixels within a window. Thus, in this paper, we develop a fast and accurate census transform-based stereo algorithm using the moving window and parallel processing techniques. The resulting system computes 32 disparity of 320 by 240 images at 34 frames per second for the window size of 5x5.

1 Introduction

Stereo vision is the method to extract 3D information using 2D images from different view points. Stereo vision may can be used for other vision applications such as topographical survey, obstacle detection, object tracking, and face recognition, etc.

Recently, many stereo vision algorithms which have good performance in accuracy are proposed in the field of binocular system. However, in spite of rapid progress in hardware performance, implementation of a frame rate stereo vision algorithm with high accuracy is still difficult. Most of accurate algorithms usually have complex procedure and thus are computationally unattractive. Although accuracy is an important factor, in real situations, many stereo applications require real-time processing. For this reason, so far, many researchers have developed and proposed several fast stereo vision algorithms. One approach is to use a box filtering method. In this method, the performance is not affected by matching window size [1, 2]. This method uses simple correlation method by SAD (Sum of Absolute Difference) or ZNCC (Zero mean Normalized Cross Correlation), and reduces duplicated computation in the same disparity window. Sun [3] presented a pyramid method, which uses higher order disparity information to reduce the boundary of disparity range in computational stage. Also Mayer [4] proposed a fast cooperative stereo method that eliminates duplicated 3D window computation in each iteration step.

In practice, stereo images are usually acquired by different grabbers and the brightness is not consistent in each corresponding region. This problem raises many difficulties in some stereo algorithm which are assumed brightness consistency of two images. Census transform method uses relative intensity of input images which performs robust under different absolute intensities of input images and noises. Some researchers showed that census transform outperformed other LOG filter-based fast approaches [6, 7, 8].

In this paper, for the development of a fast and robust stereo system for real applications, we propose the fast census transform-based stereo algorithm using the moving window and parallel processing techniques based on Intel Pentium SSE2 architecture [10, 11].

2 Census Transform-based Stereo

Census transform is a non-parametric local transform method [5]. Let $C(P)$ denotes census transform of one point $\boldsymbol{P}$. $C(P)$ maps the local neighborhood of pixel ***P*** to a bit string representing the thresholded values of them by the intensity of a given pixel. $C(P)$ is defined as

$$C(P) = \bigotimes_{[i,j]\in D} \xi(P, P+[i,j])$$

,where $\otimes$ denotes concatenation, D denotes nonparametric window around ***P***, and ξ denotes transform defined by

$$\xi(P, P+[i,j]) = \begin{cases} 1, \textit{where } P > P+[i,j] \\ 0, \textit{otherwise} \end{cases}$$

Figure 1 shows an example of the census transform of a window image w.r.t. the center pixel. Census transform converts relative intensity difference to 0 or 1 in 1 dimensional vector form.

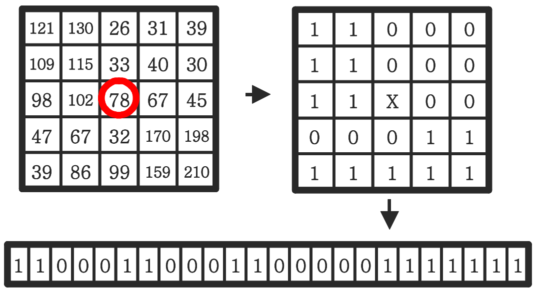


Fig. 1. Census transform example

Census transform is therefore invariant under changes in gain and bias. In this manner, vector is allocated to a pixel ***P*** and an image is transformed 3 dimensional data as in figure 2.

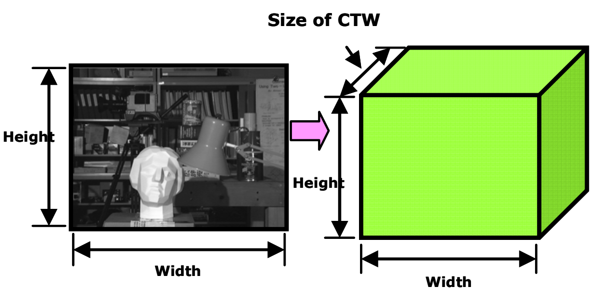


Fig. 2. Census transform of an image

With the transformed data, we match correspondences to extract depth map. When we match the corresponding point, we use the Hamming distance between two transformed vectors. The Hamming distance is the number of differences between two vectors as shown in Figure 3.

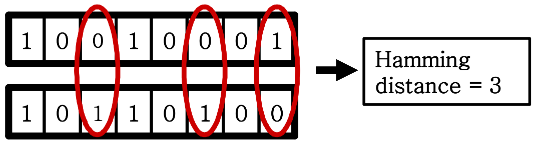


Fig. 3. Computation of the Hamming distance

For more robust matching, window-based correlation is usually employed, which requires the sum of Hamming distances. Figure 4 shows the window-based matching method of census transformed data. From the result, we can obtain disparity values.

3 Fast Algorithms

For real time process, we developed fast approaches for the calculation of Hamming distance and correlation between transformed vectors. The sum of the Hamming distance is used to compute the correlation between two census transformed vectors. The summing operation many redundancies, since each individual pixel has to be calculated in every step. We can overcome this problem by using moving window technique, which eliminates duplicated computations very efficiently [2].

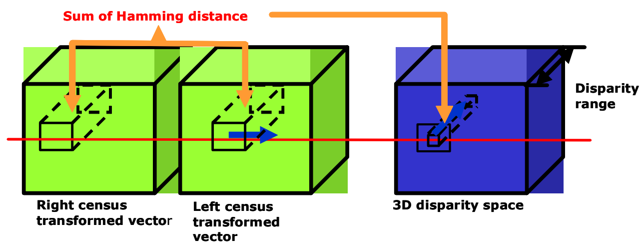


Fig. 4. Sum of Hamming distance

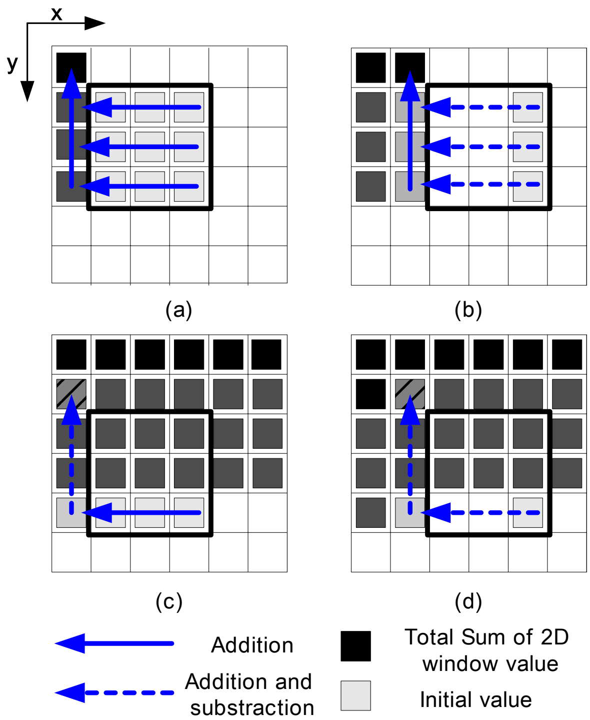
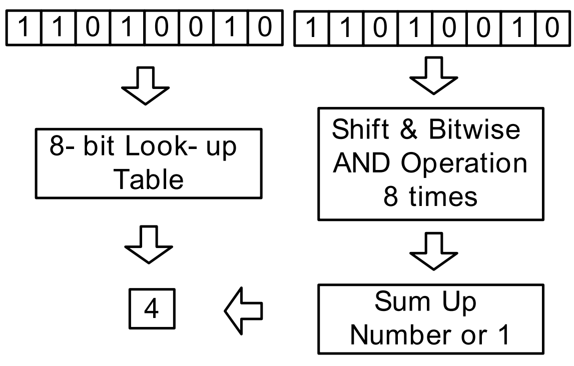


Fig. 5. Moving window technique

Figure 5 shows how moving window technique can be used to eliminate redundant operations. The row sum is calculated and stored in the near memory location. The following sum is calculated from the previous sum by adding next pixel and subtracting previous one. The column sum can be calculated in the same manner. Thus, once initial row sums and column sums are calculated, we can obtain the sum of following pixels in fixed computational cost (two additions, two subtractions). Thus, the computational complexity is reduced from N (size of window) to constant (fixed number of addition and subtraction). The main advantage of this technique is that the total computational complexity is independent of the correlation window size.

Figure 6 shows how look-up table is used in calculating the Hamming distance. Look-up table is frequently used for mapping data when computational complexity is relatively high. We used 8-bit look-up table to compute the Hamming distance when SSE2 instructions are not available. It showed better performance than the traditional bit-shift based algorithm.



(a) Look up table (b) Bit-shift

Fig. 6. Look up table vs. traditional bit-shift method

4 Parallel Processing

Intel Pentium IV processor provides SIMD (Single Instruction Multiple Data) instructions which can be effectively used for stereo matching. Particularly, MMX (Multi Media Extension) and SSE2 (Streaming SIMD Extensions 2) instruction sets can be effectively used because most of computing processes are near-memory calculations. MMX structure consists of eight 128-bit general purpose registers (XMM registers) and an ALU that supports arithmetic and logical operations for packed data. Fig 7 shows various data types that can be used in XMM registers.

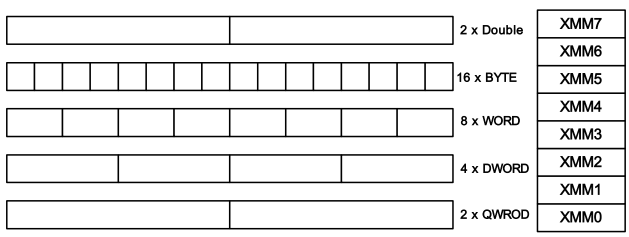


Fig. 7. Data Type and XMM Registers

The data in the memory should be continuous and 16-byte aligned to maximize the benefit of SSE2 instructions. However, the census transform stage has poor near-memory access condition in the n-direction, which denotes the census transformed vector. To solve this problem, we simultaneously computed the census transform of the same n index from each 16 continuous pixels in x-direction.

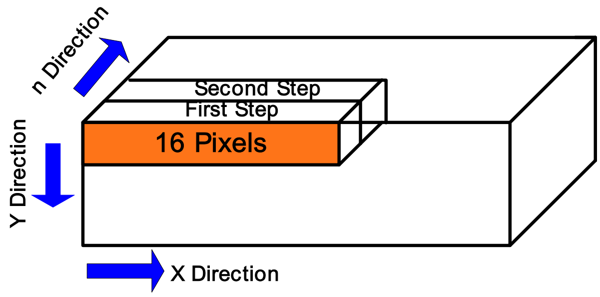


Fig. 8. Memory allocation for SSE2 in census transform

Figure 8 shows the memory allocation in the census transform stage. This approach constructs census transform vector that is continuous along *x*-direction instead of *n*-direction, which has some advantages in the next process. The right edge components of this memory cannot be calculated in the same manner because they cross over the border of the image, so we added additional routine for this region. Hamming distance can be calculated directly in disparity coordinate because census transform vector is arranged in *x*-direction which is the same direction with disparity range. However, this feature limits disparity range as a multiple of 16 for the use of SSE2 instructions. Though we can adjust algorithm to fit any size of disparity range, it will cost duplicated computation and high penalty. To build disparity map, we combined moving window technique which was mentioned above and SSE2 instructions as shown in Figure 9. Since the Hamming distances were stored in continuous manner, it is easy to calculate disparity map.

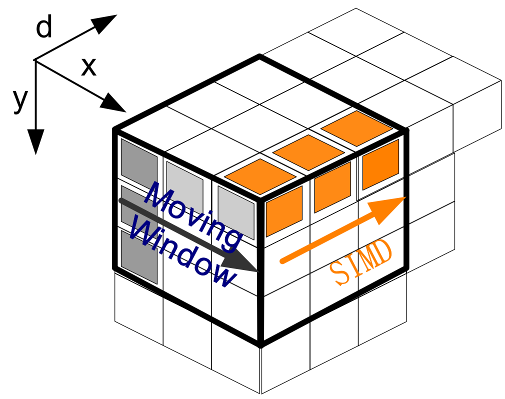


Fig. 9. Combination of moving window and SSE2

5 Experimental Results

We have performed an experiment with the proposed algorithm under the conditions denoted in Table 1. The measured processing time is an average of 100 trials. The performance of the proposed census transform-based stereo algorithm is summarized in Table 2. The census transforming (CT) part is only affected by the size of census transform window (CTW), whereas calculating the Hamming distance (CalHD) is affected by both CTW size and disparity range.

Table 1. System specifications

CAMERA : stereo MEGA-D (Videre design)
CPU : Intel Pentium-IV 2.4GHz
Data : 320 x 240 Gray stereo images
Census transform window : 5x5, 7x7, 9x9
Disparity searching range : 16, 32
Correlation window : 11

Therefore calculating Hamming distance spends most of time in the algorithm. Processing 320x240 8-bit grey images with 32 disparity search range and 5x5 CTW size takes 29.2ms (34.2 frame rate), which is fast enough for real-time applications. Figure 10 shows a resulting disparity map by the proposed fast census transform-based stereo vision system.

Table 2. Performance evaluation

Disparity Range = 32

Method	CT Window	Time (msec)				Frame Rate
		CT	Cal HD	Corr	Sum	
Census with SSE2	5	5.9	17.5	5.8	29.2	34.2
	7	11.6	29.6	5.6	46.8	21.4
	9	35.4	52.6	4.3	92.3	10.8
Census without SSE2	5	34.3	24.1	14.2	72.5	13.8
	7	66.2	35.1	14.1	115.4	8.7
	9	208.3	66.5	14.2	289.0	3.5

Disparity Range = 16

Method	CT Window	Time (msec)				Frame Rate
		CT	Cal HD	Corr	Sum	
Census with SSE2	5	5.9	9.6	2.3	17.8	56.1
	7	11.6	16.1	2.3	29.9	33.3
	9	35.4	29.3	2.2	66.9	14.9
Census without SSE2	5	34.3	14.2	6.7	55.2	18.1
	7	66.2	20.4	6.9	93.5	10.7
	9	208.3	39.3	6.6	254.2	3.9

CT : Census Transforming Stage / Corr : Correlation Matching Stage / HD : Calculating Hamming Distance Stage

6 Conclusion and Future work

In this paper, we proposed a new fast census transform-based stereo algorithm using moving window technique and parallel processing by SSE2 instructions. We showed that real-time census transform can be implemented on PC-based system with variable window sizes and disparity ranges.

We are going to apply the fast census transform-based matching technique to other computer vision problems.

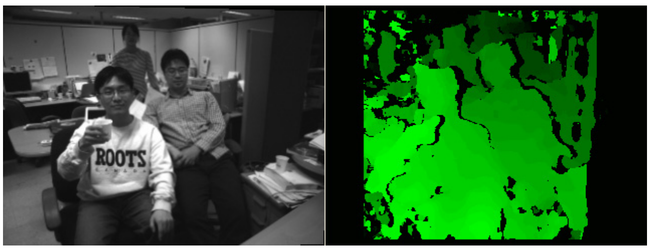


Fig. 10. Disparity map obtained by our method

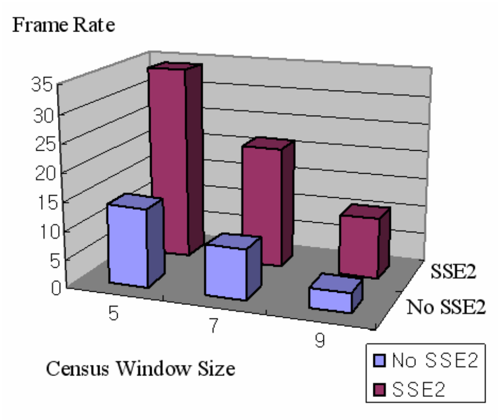


Fig. 11. Disparity map obtained by our method

Acknowledgment This work is supported by the Ministry of Information & Communication, Korea, under the information Technology Research Center (ITRC) Support Program.

References

[1] O. Faugeras, et al. "Real time correlation-based stereo : algorithm, implementations and applications", INRIA Technical Report 2013, 1993.

[2] K. Mühlmann. et al. "Calculating Dense Disparity Maps from Color Stereo Images, an Efficient Implementation," IEEE Workshop on Stereo and Multi-Baseline Vision. In Conjunction with IEEE CVPR-2001, pp. 30-36, Kauai, Hawaii, December 9-10. 2001.

[3] C. Sun, "Rectangular Subregioning and 3-D Maximum-Surface Techniques for Fast Stereo Matching," IEEE Workshop on Stereo and Multi-Baseline Vision. In Conjunction with IEEE CVPR-2001, pp. 44-53, Kauai, Hawaii, December 9-10. 2001.

[4] H. Mayer, "Analysis of means to improve cooperative disparity estimation", ISPRS Conference on Photogrammetric Image Analysis, Stockholm, Sweden, May 2003.

[5] R. Zabih and J.Woodfill, "Non-parametric Local Transforms for Computing Visual Correspondence", Third European Conference on Computer Vision, Stockholm, Sweden, May 1994.

[6] J. Banks and P. Corke, "Quantitative evaluation of matching methods and validity measures for stereo vision," Int. J. Robotics Research, vol. 20, pp. 512-532, July 2001.

[7] Heiko Hirschmller, "Improvements in Real-Time Correlation Based Stereo Vision", Proceedings IEEE Workshop on Stereo and Multi-Baseline Vision, pp. 141-148, Kauai, Hawaii, December 2001.

[8] S. Gautama, S. Lacroix, and M. Devy, "Evaluation of Stereo Mathing Algorithms for Occupant Detection", Proceedings of the International Workshop on Recognition, Analysis, and Tracking of Facs and Gestures in Real-Time Systems, pages 177-184, Cofus, Greece, Sept 1999.

[9] J. Woodfill, G. Gordon, R. Buck, "Tyzx DeepSea High Speed Stereo Vision System", Proceedings IEEE Computer Society Workshop on Real Time 3-D Sensors and Their Use, Conference on Computer Vision and Pattern Recognition, Washington, D.C., June 2004.

[10] Intel Corporation, IA-32 Intel Architecture Software Developer's Manual, 2001, 245470, 245471, 245472.

[11] Intel Corporation, Intel Pentium4 and Intel Xeon Process Optimization Reference Manual, 2001, 248966.