International Journal of
Electronic Commerce Studies
Vol.3, No.1 , pp.21-34, 2012

A STEREO MATCHING ALGORITHM BASED ON ADAPTIVE WINDOWS

Chuen-Horng Lin
National Taichung University of Science and Technology
No. 129, Sec. 3, Sanmin Rd., Central Dist., Taichung City 404, Taiwan
linch@nutc.edu.tw

Cheng-Hsin Kuo
National Taichung University of Science and Technology
No. 129, Sec. 3, Sanmin Rd., Central Dist., Taichung City 404, Taiwan
s18983117@nutc.edu.tw

Li-Jung Fu
National Taichung University of Science and Technology
No. 129, Sec. 3, Sanmin Rd., Central Dist., Taichung City 404, Taiwan
s1899B107@nutc.edu.tw

ABSTRACT

The aim of this paper is to develop a stereo matching algorithm based on adaptive windows for a stereo vision domain. This method retains the advantageous image processing speed of traditional methods, and proposes a means of decreasing the error rate, making it the best choice for application in real time systems. Depending on the characteristics of different regions, the proposed method provides a suitable window for stereo vision matching. The processing method is differentiated into disparity consistency, the disparity for a smooth region, the vote disparity between the 8-neighbors and the uniqueness of disparity. A different processing method is used in the lab with the sum of absolute difference (SAD), and the result is compared with a fixed window method; the result proves that this method improves on the SAD method, and yields more accurate depth information.

Keywords: Stereo Matching, Stereo Vision, Disparity, Adaptive Window, Sum of Absolute Difference

1. INTRODUCTION

Currently, the field of robotics is receiving a great deal of attention; in this field machine vision is one of the most important sources of detection. If machine vision possesses high image identification ability, then robots can detect more diversified environmental changes and react suitably to those environmental changes. Machine vision has been developing for almost 40 years; many related studies and topics are being explored, with most of them roughly sorted and rearranged[1, 2].

Stereo vision has become a very important topic of study, with special attention being paid to its application in machine vision. It is also a very important field in computer vision. There are several methods of conducting this type of research ranging for computer vision, using sensors like infrared rays, lasers, sonar, etc, to detect surrounding objects. Many of these sensors are able to acquire accurate three-dimensional coordinate information, but are very expensive. At present, camera and DV costs are decreasing, while resolution is continuously improving, and with the maturity of digital image processing and computer vision theory, stereo vision systems are being widely used for the measurement of distance in certain environments. A means of developing a relationship between three-dimensional coordinate systems and two-dimensional image coordinate systems for the measurement of three-dimensional stereo vision is an important study topic.

The purpose of camera calibration theory is to maintain horizontal direction consistency of overlapped images taken by several cameras. Stereo vision needs to be accurately calibrated before it can acquire more accurate three-dimensional information[3]; this process is a necessary pre-requisite to proficient stereo vision technique.

Stereo vision uses geometric relationships to determine the depth of a scene, on the basis of corresponding left and right images. Since the left and right images could possibly be affected by lighting to produce repetitive texture and occlusion[2], if suitable and correct correspondence is not found, then errors will occur in the calculation of the scene depth. The correspondence of stereo images is therefore considered to be the most important problem in stereo vision.

Applying the matching relationship between left and right images can determine the reciprocal relationship between depth and disparity, and we can therefore acquire depth if we can determine the disparity value of each pixel. Disparity is mainly used to calculate the degree of difference of corresponding pixels in the left and right images; thereby, the relationship can be derived easily, with effective performance in real time systems. The

means of determining correspondence can be differentiated into local [4]and global[5]; the local method is widely applied to real time applications because it is less complicated than the global method. However, the local method cannot acquire high accuracy about occlusion, uniform texture and ambiguity portions. Even though these problems can be solved using the global method, considerable time is expended in doing so.

D. Scharstein used the correspondence of a pair of left and right images for the matching method[6]. This process of calculation can be divided into 4 major steps: 1. Matching cost computation, 2. Cost aggregation, 3. Optimization and 4. Disparity refinement. Matching cost computation determines the color resemblance between corresponding parts of the two images; cost aggregation is widely used for the variation portion; optimization can also be divided into the local and global methods; disparity refinement aims at achieving improvements on one single characteristic.

Widely used matching cost computation methods include normalized cross-correlation (NCC), sum of squared differences (SSD), sum of absolute differences (SAD), and Rank and Census[2].

From numerous related study methods, of all the correspondence matching computation methods cost aggregation is the easiest to change, by altering steps like mask size, shape, weight, etc. In recent years, many scholars have changed the window matching of the aggregation window to an adaptive support window and adaptive support weight. In the adaptive support window method[7], it is first necessary to consider whether the adjacent pixel value is from a similar object; then the similarity between pixels can be measured. This problem needs to be solved for the support window. The adaptive support-weight approach by Yoon et al.[4, 8, 9, 10], also utilizes characteristics between pixels to determine suitable weight, and is usually applied by means of an adaptive support window.

Scharstein and Szeliski divided the methods of disparity into two categories[6] : the local and the global methods. In the local method[2, 6], mask size and shape should be determined first. The Best Match Point is determined based on the sum of the square difference value and absolute difference value of the windows calculation between the two pictures. In the global method, commonly used methods are: dynamic programming (DP), scanline optimization (SO), graph cut (GC) and belief propagation (BP).

Disparity optimization aims to solve the occlusion problem; it will identify incorrect correspondence and then use other methods to refill the disparity value, like bilinear interpolation or the adjacent disparity voting methods. The occlusion problem arises from the different placements of

cameras that cause a single image to appear in one area of the scene without any correspondence in the other image. With the incorrect correspondence detection method, the main concept of left-right checking (LRC)[11] is that the disparity value should be consistent for correspondence obtained using the left image to find the right image, and vice versa.

Disparity study mainly targets image processing speed and the disparity error rate. This study therefore retains the advantage of the image processing speed of traditional methods, and decreases the problematically high error rate to offer the best choice for real time systems. A stereo matching algorithm based on an adaptive window is proposed in order to decrease the error rate of traditional computation methods. The four methods used in this study are: disparity consistency, the disparity for a smooth region, the vote disparity between the 8-neighbors and the uniqueness of disparity.

2. REVIEW OF DISPARITY

In recent years, many studies of pixel-corresponding-point searching have been proposed[4, 5], each with its own unique characteristics; however, the correctness of the depth of pixels measured will be affected by the accuracy of the pixel-corresponding-point.

2.1 Comparison of SSD and SAD

SAD selects a fixed-size square window of a certain pixel's neighboring pixel and compares it with the fixed sized square window of another image to calculate the aggregation of the absolute differences of the corresponding positions. The result will be used as the WTA (winner-takes-all) of similar windows[6].

In recent years, numerous researchers have offered many different algorithms in related studies. The comparisons of the disparity and ground truth found in different algorithms are shown in Table 1[12]. In Table 1, Non-Occ represents the non-occlusion region, Disc represents depth discontinuity regions and Time represents the computing time of the computer. The results show that the computing time for SAD is very fast. Despite SAD's disparity error being slightly higher, it is not so different from other techniques found in other algorithms.

The theorems of SAD and SSD are simple and easily coded. The goal of speedy instant image processing, even with a higher disparity error rate, cannot be reached by other algorithms. For this reason, this article, based on

the premise that computation speed remains unaffected, aims to improve the SAD and SSD algorithms, and to decrease the disparity error rate.

Table 1. Accuracy and running time of the performance evaluations

Algorithm	Tsukuba			Venus			Teddy			Cones			Time
	Non-Occ	Disc	All	Non-Occ	Disc	All	Non-Occ	Disc	All	Non-Occ	Disc	All	(mm:ss)
Segment support [9]	**1.25**	**6.68**	**1.62**	**0.25**	**2.59**	**0.64**	8.43	**18.2**	14.2	**3.77**	9.77	9.87	**9:34**
Adaptive weight [10]	1.38	6.90	1.85	0.71	6.13	1.19	**7.88**	18.6	**13.3**	3.97	**8.26**	**9.79**	18:14
Segment. Based [13]	8.18	18.77	2.27	8.06	20.85	1.22	15.78	29.66	19.4	13.22	24.55	17.4	0:02
SSD (25W)[14]	13.20	26.35	9.94	22.39	11.48	14.9	33.93	47.25	30.95	47.78	45.82	52.41	0:20
SSD (9W) [14]	9.7	21.22	14.92	13.72	16.36	11.69	29.33	31.49	23.49	46.59	35.15	43.88	0:08
SSD (5W) [14]	12.75	27.49	25.31	17.43	26.57	14.46	34.98	21.98	26.81	34.82	29.49	38.43	0:05
SAD (25W)[14]	10.58	23.32	8.93	17.68	10.47	12.14	31.20	40.88	26.69	46.73	39.53	48.24	0:20
SAD (9W) [14]	8.63	20.54	14.26	13.80	15.69	10.56	28.7	25.83	23.49	43.24	30.86	39.74	0:07
SAD (5W) [14]	12.36	26.89	24.88	19.52	26.14	14.22	34.44	19.34	28.62	32.79	29.41	36.54	0:05
SAD (3W) [12]	19.59	38.23	36.73	31.31	37.77	21.27	44.56	20.13	39.04	34.82	35.97	42.28	**<1s**
Best/Ratio of SAD(9W)	6.90	3.07	8.80	55.2	6.07	16.5	3.64	1.41	1.76	11.4	3.73	4.05	82

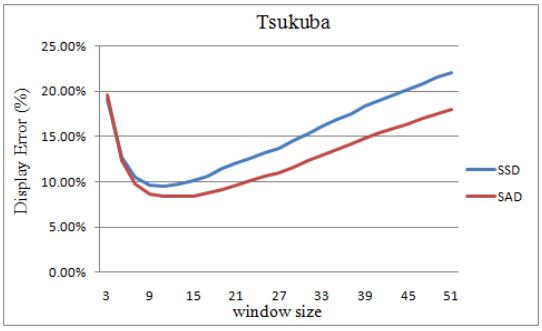
where (n W) is the window size of n × n pixels.

The best ratio of SAD (9W), as shown in the table, has a lower error rate value than the average error rate of 10.21 times more than SAD, but the execution time is 82 times faster.

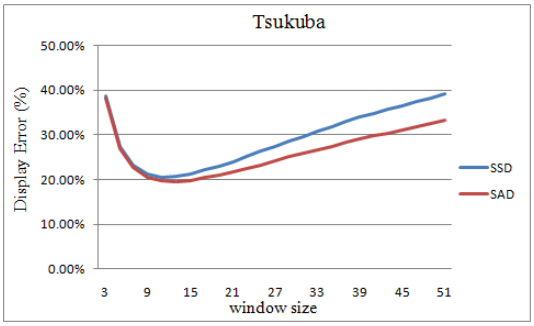
2.2 Weaknesses of Disparity for Fixed Window Size

The SAD and SSD algorithms are designed to select fixed-size square windows in order to compute the similarity between them. The different sizes of the windows they select will affect the correctness of the disparity. This paper, based on SAD and SSD's Non-Occ, Disc and All regions, conducts an experiment in obtaining different disparities from different square windows. The result shown in Figure 1(a) is the original image of Tsukuba, and Figure 1(b) shows the result of the Non-Occ region. We find that when the window increases from 3 × 3 to 9 × 9, the disparity error lowers considerably and rapidly; and when the window increases from 9 × 9 to 48 × 48, the disparity slowly increases. The result of the Disc region is shown in Figure 1(c), the result of all regions is shown in Figure 1(d), and the findings are similar to the Non-Occ region.

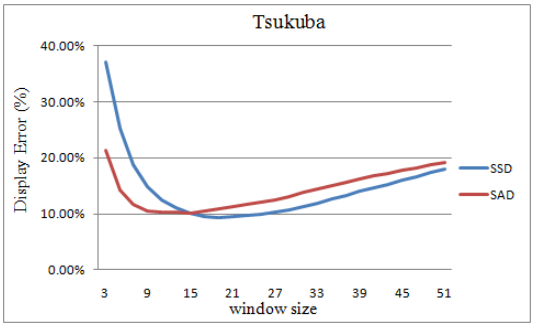
(a) Original image of Tsukuba



(b) Non-Occ area



(c) Disc area



(d) All area

Figure 1. Disparity errors of different windows

From the results of the experiment, it can be seen that a small window is more suitable to be selected for the region with a big difference of depth; for the contrary, the bigger window should be selected.

3. PROPOSED METHOD

From the discussion above, SAD has a considerable advantage in terms of processing speed; it is the optimal choice for an instant system, even though it has a higher error rate than other algorithms do. In order to lower the error rate of the SAD algorithm, the following methods are explored in this paper: disparity consistency, the disparity for the smooth region, the vote disparity between the 8-neighbors and the uniqueness of disparity.

3.1 Consistent Disparity

An adaptive window is proposed by this article. The adaptive windows proposed by this paper are square windows with different sizes, which are not the same as other adaptive windows in which irregular sizes or shapes are adopted. Firstly, five square windows of 3 × 3, 7 × 7, 11 × 11, 15 × 15

and 19 × 19 are adopted, and five disparity errors of the five windows in each pixel are computed using the SAD algorithm. Take the example of the left image: its coordinate (x, y) represents $p_L(x, y)$; its disparity is $d(p_L(x, y))$. The computation is as follows:

$$d(p_L(x,y)) = \min\left\{ \sum_{c\in\{R,G,B\}} SAD_c\left(p_L(x,y) - p_R(x,y-i)\right)\right\}, \quad i = 0,1,..., y\,, \qquad (1)$$

where $p_R(x, y)$ is the coordinate of the right image. SAD_c represents the SAD values of the three color components R, G and B. To avoid the noise affecting SAD values, a limit of the maximum differential threshold value T (truncation)[6] is added for each disparity computed. The disparity of the two points $\left(p_L(x, y) - p_R(x, y-i)\right)$, which is bigger than T , will be replaced by T . The value of T is set at 60 in this paper, and equation (1) is rewritten as follows:

$$d(p_L(x,y)) = \min\left\{ \sum_{c\in\{R,G,B\}} SAD_c\left(p_L(x,y) - p_R(x,y-i), T\right)\right\}, \quad i = 0,1,..., y \qquad (2)$$

If the five received disparities are the same after calculation, it is called consistent disparity (CD), and the region composed of all CD is called the certain disparity region (CDR); on the other hand, the region composed of uncertain points is called the uncertain disparity region (UCDR). The result is shown in Figure 2, in which CDR occupies 65.52% and is expressed by white, while UCDR occupies 34.48% and is expressed by black. In the region of Non-Occ, CD occupies the larger area, and is also the smoother region in the image.

Figure 2. Certain disparity region of Tsukuba image

3.2 The Disparity for the Smooth Region

After CD treatment, the majority of the remaining UCDR belongs to deep discontinuity or coarse color; therefore, the disparity of the UCDR region is figured out in accordance first with the image index of coarse degree. Lin and Syu[15] took the coarse degree as the image feature, which was separated into five levels, k=1~5, which showed the coarse degree for each pixel; the smaller the coarse degree, the coarser the pixel. After treatment, the image of coarse degree is as shown in Figure 3.

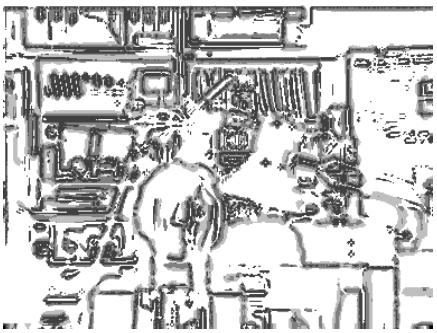


Figure 3. Tsukuba image of coarse degree

According to the image of coarse degree in Figure 3, the occupied percentage of the two regions of Non-Occ and Disc is as shown in Table 2, in which the occupied percentage is much more when k=5. In addition, the occupied percentage of coarse degree in the UCDR region is shown in Table 3, in which the occupied percentage is also much more when k=5. Therefore, we calculate the disparity in the region of k=5 of the UCDR. As it belongs to a smooth region when k=5, we choose two disparity values of 15 × 15 and 19 × 19 square windows for disparity amendment.

Table 2. The occupied percentage of coarse degree in whole image

k	**Non-Occ**	**Disc**
k=1	18.38%	18.70%
k=**2**	15.78%	18.70%
k=**3**	11.90%	17.61%
k=**4**	11.49%	17.37%
k=**5**	**42.46%**	**27.63%**

Table 3. The occupied percentage of coarse degree in the UCDR region

k	**Non-Occ**	**Disc**
k =1	14.67%	15.42%
k =**2**	14.24%	15.09%
k =**3**	11.30%	14.47%
k =**4**	12.57%	19.52%
k =**5**	**47.21%**	**35.50%**

First it must be determined if the difference of the two disparity values of 15 × 15 and 19 × 19 windows is within ± 1 of the UCDR region, and k =5. If yes, it is the required disparity value of this pixel, and is further concluded to CDR.

3.3 The vote disparity between the 8-neighbors

This section will determine the disparity of the UCDR by voting on the adjacent disparities. This means that, for the pixels on the CDR, we applied voting on the disparities of the eight neighboring pixels, and took them as the basis of the decision.

The method of voting disparity mainly takes pixels of the UCDR as targets to process. First, we calculated a total of five different disparities for window sizes 3 × 3, 7 × 7, 11 × 11, 15 × 15 and 19 × 19, respectively, using the target pixel and its eight adjacent pixels, we got 45 disparities. After voting on these different disparities, if the disparity with the highest number of votes has a disparity higher than the threshold τ, we defined this disparity value as the one for the target pixel, and moved this target pixel to the CDR group. The remaining target pixels which did not meet the threshold requirements would still be in the UCDR group. This study assumes that the threshold τ should be bigger than the half value, and is set as 20.

3.4 The Uniqueness of Disparity

For the UCDR, this study proposes that the disparities of the remaining region be decided by the uniqueness of disparity. The uniqueness of disparity requires considering different window sizes when calculating the disparity of each target window; we divided them into large and small windows to get the differences between them. The one with a smaller difference is defined as the uniqueness of that window.

First, each pixel in the UCDR area is called a target pixel; for this target, small windows of 3 × 3 and 7 × 7, as well as large windows of 15 × 15 and 19 × 19 were selected. We then calculated the disparities of each window of both the small and large window groups, as well as the differences between disparities of each window group. Finally, disparity values of large/small window groups were compared, and the disparity of the group with smaller difference was taken as a candidate for the disparity of that target. In addition, we averaged the disparities in that window group, and took the result as the disparity of the target pixel; this target pixel was then put into the CDR. Therefore, after this processing step, the disparities of all pixels in the entire image could be derived.

4. EXPERIMENTS

To verify the accuracy of the algorithm in this study, we used standard images with three-dimensional vision from the Middlebury stereo image database[16]. The experiments in this study used the MatLab7.8.0 (R2009a) platform. Scharstein and Szeliski[4] , from 2001 to 2006, shot four three-dimensional visual images for users to perform experiments on. They used structured light technology in electrical engineering to obtain the ground truth disparity maps of images with 3D vision. This article used their ground truth of images for the comparison of experimental data.

The disparity for adaptive window size results are shown in Table 4. Among them, the comparison region is the disparity-region divided by the consistency disparity, while the values in the table are their error rates. In three of the regions, we can clearly see that the error rates of disparities obtained by CD, which is proposed by this article, are lower than those obtained by SAD. Moreover, the complete region occupies up to approximately 50 - 65% of the entire image.

Table 4. Performance comparison in UD region for consistency disparity

Method / Image	**SAD (window size = 5 ×5)**			**SAD (window size = 9 × 9)**			**Consistency disparity**			
	Non-Occ	**Disc**	**All**	**Non-Occ**	**Disc**	**All**	**Non-Occ**	**Disc**	**All**	**Area**
Tsukuba	2.85%	9.82%	4.01%	1.84%	9.60%	3.06%	**1.73%**	**9.51%**	**2.95%**	65.52%
Venus	4.60%	15.36%	5.41%	1.87%	15.46%	2.70%	**1.47%**	**15.36%**	**2.31%**	59.28%
Teddy	7.12%	16.47%	10.83%	5.22%	15.70%	8.99%	**4.88%**	**15.38%**	**8.67%**	53.01%
Cones	5.07%	12.00%	10.20%	3.57%	10.54%	8.76%	**3.33%**	**10.25%**	**8.54%**	59.21%

This paper proposes the disparity for the smooth region, and its experimental results are shown in Table 5. Among them, three images in the Disc region had higher error rates, while the rest were lower. Images in the

All area also had lower error rates; the completion area achieved about 14% to 25% of the entire image.

Table 5. Performance comparison in region of UD on disparity for smooth region

Method / Image	SAD (window size = 5 ×5)			SAD (window size = 9 × 9)			The disparity for smooth region			
	Non-Occ	**Disc**	**All**	**Non-Occ**	**Disc**	**All**	**Non-Occ**	**Disc**	**All**	**Area**
Tsukuba	31.96%	**29.17%**	32.38%	18.71%	40.59%	19.18%	**13.77%**	68.05%	**14.27%**	14.36%
Venus	58.07%	**57.50%**	58.48%	29.75%	73.82%	30.39%	**8.02%**	83.38%	**8.82%**	25.05%
Teddy	51.18%	49.10%	55.30%	27.96%	**46.91%**	33.98%	**16.92%**	53.30%	**23.20%**	21.58%
Cones	49.24%	53.18%	54.59%	24.00%	39.61%	31.77%	**9.60%**	**30.13%**	**17.86%**	17.06%

The results of the vote disparity between the 8-Neighbors are shown in Table 6. Similarly, in the All area, the results also show low error rates; the completion area achieved is about 11% to 19% of the entire image.

Table 6. Performance comparison in region of UD on the vote disparity between the 8-neighbors

Method / Image	SAD (window size = 5 ×5)			SAD (window size = 9 × 9)			The vote disparity between the 8-neighbors			
	Non-Occ	**Disc**	**All**	**Non-Occ**	**Disc**	**All**	**Non-Occ**	**Disc**	**All**	**Area**
Tsukuba	28.91%	**29.67%**	32.60%	18.00%	36.98%	21.96%	**14.35%**	40.58%	**17.85%**	17.35%
Venus	40.08%	**46.12%**	42.94%	21.66%	58.68%	25.18%	**16.04%**	63.09%	**19.52%**	11.88%
Teddy	43.66%	47.68%	53.98%	35.03%	**45.70%**	46.23%	**33.27%**	46.38%	**44.25%**	16.44%
Cones	35.36%	39.62%	46.52%	19.76%	30.05%	32.32%	**16.77%**	**28.83%**	**29.14%**	18.41%

The experimental results processed by the uniqueness of disparity are shown in Table 7. With the exception of the Teddy image in the All area, which had a higher error rate than SAD with window size = 5 × 5, all of the images had lower error rates than the SAD; the completion area achieved is about 2.7% to 8.9% of the entire image.

Table 7. Performance comparison in region of UD on the uniqueness of disparity

Method / Image	SAD (window size = 5 ×5)			SAD (window size = 9 × 9)			The uniqueness of disparity			
	Non-Occ	**Disc**	**All**	**Non-Occ**	**Disc**	**All**	**Non-Occ**	**Disc**	**All**	**Area**
Tsukuba	53.56%	**40.31%**	57.74%	42.09%	45.21%	47.24%	**35.47%**	51.00%	**40.35%**	2.77%
Venus	84.52%	**68.47%**	86.09%	79.74%	85.20%	81.59%	**67.87%**	80.05%	**70.19%**	3.79%
Teddy	**75.86%**	**70.43%**	**84.60%**	76.50%	75.56%	84.93%	77.92%	76.17%	85.10%	8.97%
Cones	67.45%	62.65%	85.24%	65.29%	66.93%	83.36%	**58.76%**	**61.23%**	**77.63%**	5.32%

Finally, we combined the four methods, of disparity consistency, the disparity for smooth region, the vote disparity between the 8-neighbors, and the uniqueness of disparity, in Table 8, where it can be clearly seen that the proposed method and the received result are both better than the traditional method.

Table 8. Performance comparison of the disparity

Method / Image	SAD (window size = 5 ×5)			SAD (window size = 9 × 9)			The proposed method			
	Non-Occ	**Disc**	**All**	**Non-Occ**	**Disc**	**All**	**Non-Occ**	**Disc**	**All**	**Area**
Tsukuba	12.10%	**17.26%**	13.92%	7.61%	20.15%	9.47%	**6.12%**	24.00%	**7.88%**	100%
Venus	25.21%	**30.27%**	26.45%	13.98%	35.99%	15.36%	**7.01%**	37.35%	**8.44%**	100%
Teddy	26.55%	34.63%	34.05%	19.08%	**34.10%**	27.21%	**16.26%**	34.89%	**24.39%**	100%
Cones	19.24%	26.79%	28.20%	11.34%	22.10%	20.83%	**8.04%**	**20.18%**	**17.45%**	100%

In order to better show the measured result, the error rate of comparison with regard to the four images towards three different regions is expressed as an average error rate percentage. The four compared methods are: SAD mask size 5 × 5, mask size 9 × 9, the proposed method, and the optimized result, as shown in Figure 4. This further means that a lower error rate will be achieved if the proposed stereo matching algorithm on the four images, Tsukuba, Venus, Teddy and Cones is carried out.

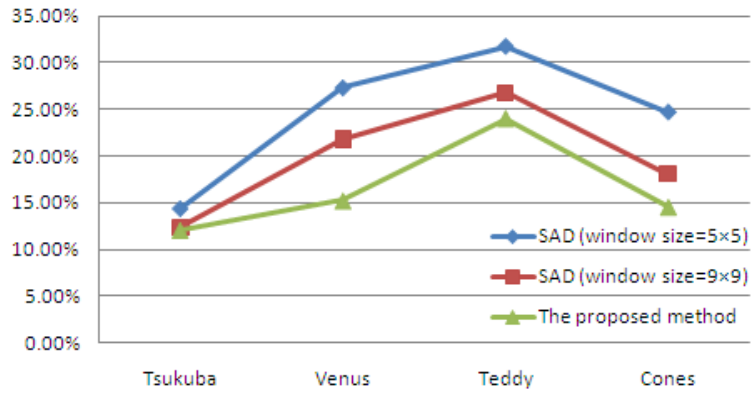


Figure 4. Stacked line chart of average error of stereo matching algorithm

5. CONCLUSION

A stereo matching algorithm based on adaptive windows was proposed in this paper. This research took Middlebury as a basis to carry out the

experiment of disparity value in order to compare the results obtained from different methods and SAD, based on a fixed window. The result proves that the proposed method effectively improves the SAD method; it not only retains the advantage of speed, but also reduces the SAD error rate and obtains more precise information in depth.

6. REFERENCES

[1] R.C. Gonzalez, and R.E. Woods, *Digital Image Processing*. 2 ed. United States: Prentice Hall, 2002.

[2] M.Z. Brown, D. Burschka, and G.D. Hager, Advances in computational stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(8), p993-1008, 2003.

[3] J.Y. Bouguet, *Camera Calibration Toolbox for Matlab*. Retrieved on March 10, 2011, from http://www.vision.caltech.edu/bouguetj/calib_doc/htmls/example5.html.

[4] K.J. Yoon, and I.S. Kweon, Locally adaptive support-weight approach for visual correspondence search. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2, p 924-931, 2005.

[5] V. Kolmogorov, and R. Zabih, Computing visual correspondence with occlusions using graph cuts. *Presented at International Conference on Computer Vision*, Vancouver, Canada, July 7-14, 2001.

[6] D. Scharstein, and R. Szeliski, A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47, p7-42, 2002.

[7] K. Wang, Adaptive stereo matching algorithm based on edge detection. *Presented at International Conference on Image Processing*, Singapore, October 24-27, 2004.

[8] Z. Zhai, Y. Lu, and Hong Zhao, Stereo matching with adaptive support-pixel set and adaptive support-weight. *Presented at International Conference on Computational Intelligence and Security*, Beijing, China, December 11-14, 2009.

[9] F. Tombari, S. Mattoccia, and L. Di Stefano, Segmentation-based adaptive support for accurate stereo correspondence. *Presented at Advances in Image and Video Technology, Second Pacific Rim Symposium*, Santiago, Chile, December 17-19, 2007.

[10] K.J. Yoon, and I.S. Kweon, Adaptive support-weight approach for correspondence search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4), p650-656, 2006.

[11] P. Fua, A parallel stereo algorithm that produces dense depth maps and preserves image features. *Machine Vision and Applications*, 6, p 35-49, 1993.

[12] F. Tombari, S. Mattoccia, L.D. Stefano, and E. Addimanda, Classification and evaluation of cost aggregation methods for stereo correspondence. *Presented at IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Anchorage, Alaska, USA, June 24-26, 2008.
[13] M. Gerrits, and P. Bekaert, Local stereo matching with segmentation-based outlier rejection. *Presented at Canadian Conference on Computer and Robot Vision*, Quebec City, Canada, June 7-9, 2006.
[14] H. Hirschmuller, P. Innocent, and J. Garibaldi, Real-time correlation-based stereo vision with reduced border errors. *International Journal of Computer Vision*, 47, p229-246, 2002.
[15] C.H. Lin, and Y.J. Syu, Fast segmentation of porcelain images based on texture features. *Journal of Visual Communication and Image Representation*, 21(7), p707-721, 2010.
[16] *Middlebury Stereo Datasets*, Retrieved on March 10, 2011, from http://vision.middlebury.edu/stereo/data/.