AN EXPERIMENTAL COMPARISON OF STEREO VISION ALGORITHMS USING MIDDLEBURY DATASET

György Tamás[1], Bíró Károly-Ágoston[2], Szabó Tamás
Phd. Student, Profesor, Msc. Student
Technical University of Cluj-Napoca

Abstract: Stereo vision, the knowledge of deep information in a scene, has great importance in the field of machine vision, robotics and image analysis. In this article an analysis of existing stereo vision matching methods is presented. The presented algorithms are discussed in terms of speed, accuracy, disparity range and time consumption. Implementations of stereo-matching algorithms in hardware are also discussed in details.

INTRODUCTION

Stereo vision can produce a dense disparity map. The resultant disparity map should be smooth and detailed; continuous and even surfaces should produce a region of smooth disparity values with their boundary precisely delineated. While small surface elements should be detected as separately distinguishable regions. Though obviously desirable, it is not easy for a stereo algorithm to satisfy these requirements at the same time. Algorithms that can produce a smooth disparity map tend to miss details and those that can produce a detailed map tend to be noisy.

Detecting conjugate pairs in stereo images is a challenging research problem known as the correspondence problem to find for each point the left image the corresponding point in the right one **[1]**. To determine these two points from a conjugate pair, it is necessary to measure the similarity of the points. The point to be matched should be distinctly different from his surroundings. Several algorithms have been proposed in order to solve this problem; however every algorithm makes use of a matching cost function so as to establish correspondence between two pixels **[2]**.

Correlation based matching typically produces dense depth maps by calculating the disparity at each pixel within a neighborhood. This is achieved by taking a square window of a certain size around the pixel of interest in the reference image and finding the homologous pixel within the window in the target image, while moving along the corresponding scan line. The goal is to find the corresponding pixel within a certain disparity range d. **[3]**.

In brief, the matching process involves computation of the similarity measure for each disparity value, followed by an aggregation and optimization step. Since these steps consume a lot of processing power, there are significant speed-performance advantages to be had in optimizing the matching algorithm.**[4]**

EVALUATION METHODOLOGIES

The data set that we used is a multi-image stereo set from the University of Tsukuba, where every pixel in the central *reference* image has been labeled by hand with its correct disparity. The images we used for stereo matching and ground truth depth map are shown in figure 1.

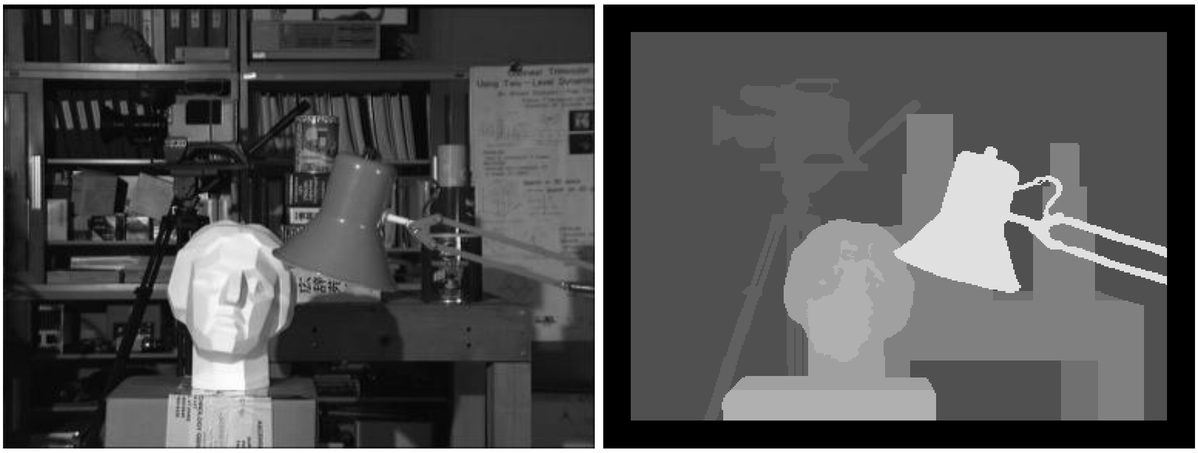

Image Ground truth

Fig. 1
Imagery from the University of Tsukuba

The ground truth images are smaller than the input images; we handle this by ignoring the borders.

The interesting regions in the Tsukuba imagery include **[5]**:

- Specular surfaces, which causes difficulty in computing depth, due to reflected motion of the light surface.

- Textureless regions, which are locally ambiguous, which is a challenge for stereo algorithms.

- Depth discontinuities, at the borders of all objects. It is difficult to compute depth at discontinuities, for a variety of reasons. It is especially difficult for thin objects.

- Occluded pixels, near some of the object borders.

The goal of this paper is to analyze the effectiveness of different methods in these different regions. Using ground truth we determine the depth discontinuities and occluded pixels. Our statistics count the number of pixels whose disparitt differs from the ground truth by more than ±1. This makes sense because the true disparities are usually fractional.

ALGORITHMS

For stereo vision algorithms methods computing dense stereo depth are divided in two classes. The first class called *Local Methods Based on Correlation* allows every pixel to independently select its disparity by analyzing the intensities in fixed rectangular window. The second class of methods relies on global methods, and typically finds the depth map that minimizes some function, called the energy or the objective function. These methods generally use an iterative optimization technique, such simulated annealing.

For this article we implemented a number of standard correlation-based methods that use fixed size square windows. We define the radius of a square whose side length is $2r+1$ to be r. The methods we chose were:

- Sum of absolute Differences (SAD)
- Sum of Squared Differences (SSD)
- Locally scaled Sum of Squared Differences (LSSD)
- Normalized Cross Correlation (NCC)
- Sum of Hamming Distances (SHD)

Sum of Absolute Differences (SAD) is one of the simplest of the similarity measures which is calculated by subtracting pixels within a square neighborhood of the pixel between the reference image $I1$ and the target image $I2$ followed by the aggregation of absolute differences within the square window and optimization using *winner-take-all* strategy (WTA). **[3]**

In Sum of Squared Differences, the differences are squared and aggregated within the square window and later optimized using WTA strategy. This measure has higher computational complexity compared to SAD algorithm as it involves numerous multiplication operations, which leads to longer running times.

Normalized Cross Correlation is well developed and often used now for the feature point matching in most of the image stitching algorithm. NCC matching is characteristic with easy operation and good anti-noise ability. **[6].**

Sum of Hamming Distances is normally employed for matching census-transformed images by computing bitwise-XOR of the values in left and right images, within a square window. This step is usually followed by a bit-counting operation which results in the final Hamming distance score.

IMPLEMENTATION

We have run the mentioned algorithms on the Tsukuba imagery, and used ground truth to analyze the results. We created a stereo camera system to check results on real-life data sets. We were interested to determine an average running time for the algorithms, the error in all regions, errors in images in regions near discontinuities and errors in disparity of non-occluded regions.

For the Tsukuba imagery the calculations were done in Matlab using an Intel i7 @ 3.40 GHz with 12 GB of memory. As an output we got the running time and the disparity map given by every algorithm. For the data sets from our camera system the calculus were made on an Nvidia Tegra K600.

RESULTS ON THE TSUKUBA IMAGERY

Figure 2 shows the depth maps computed by five different local algorithms presented in chapter 3. SAD is the simplest algorithm and can be easily embedded into FPGA and SOC, it is an ideal choice for real time systems but it is more prone to intensity variation among the two images.

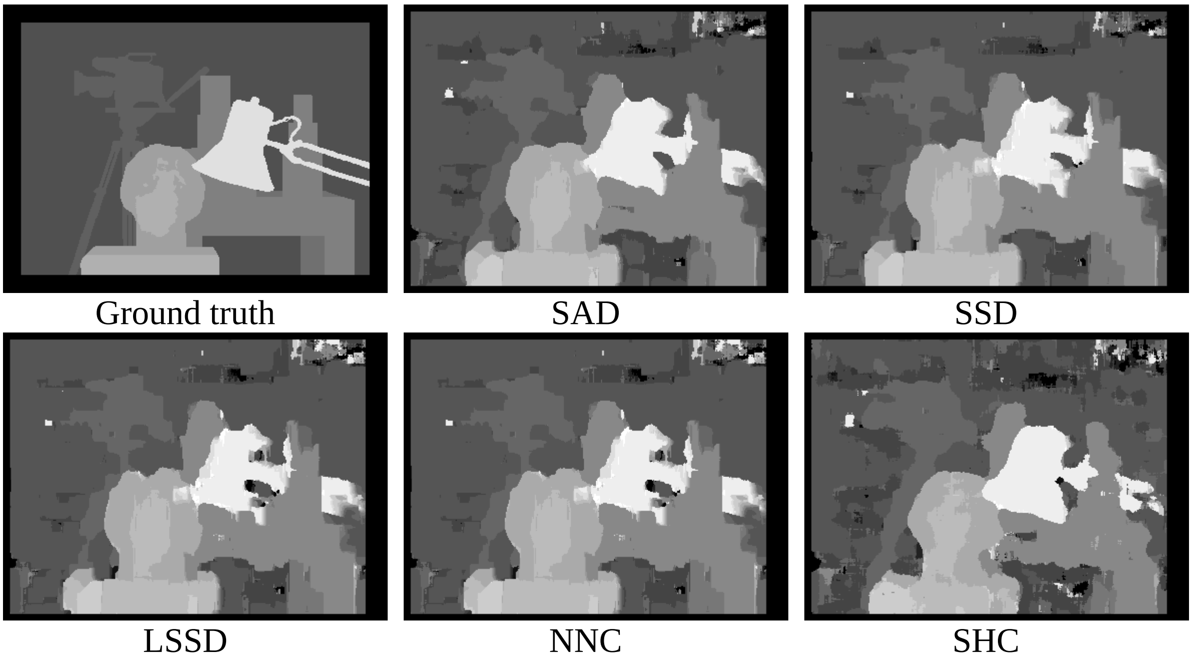

Fig. 2
Depth map calculated using different correlation-based methods.[7]

However it is interesting that the different correlation-based methods are very similar in terms of their performance, in addition the variation of error is surprisingly little after the windows were sufficiently large (figure 3).

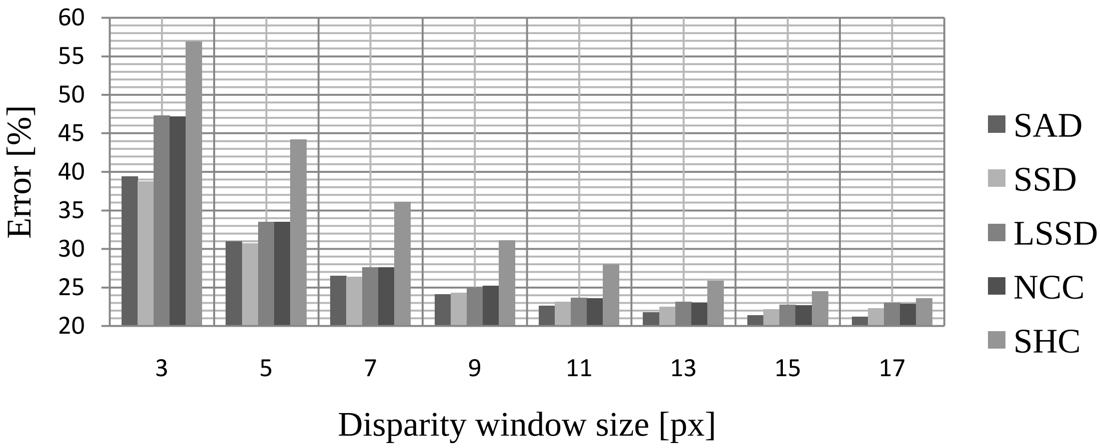

Fig. 3
Performance of standard correlation-based methods
as a function of window radius.

The only exception is the SHC algorithm which on low window sizes has the worst results, however on parallel computing devices like FPGA or GPU it has the best speed performance due the nature of the algorithm which can be calculated during one clock cycle.

One of the biggest problems of the correlation-based methods is presented in figure 3 and it is related to matching near discontinuity regions.

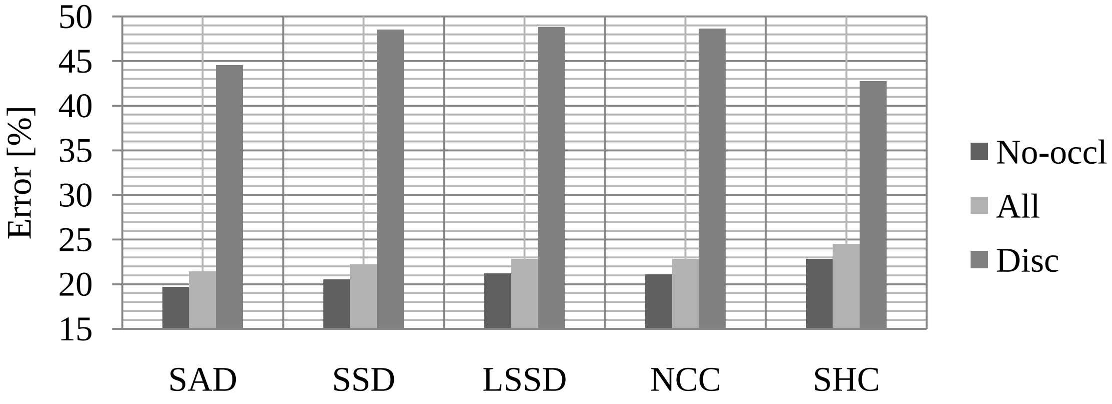

Fig.3
Accuracy of the method in a). non-occluded areas b). total image c). discontinuity areas

Errors introduced by these methods especially near discontinuities are enormous, approximately 45% in every situation, which is a poor result compared to the global matching methods.

RESULTS USING OWN STEREO SYSTEM

Our camera system consists of two webcams aligned on a 3D printed board and an algorithm to take pictures simultaneously. The test bench was built to test the capabilities of stereo matching algorithms in real-time conditions.

To present the behavior of the algorithms on oversaturated and unfocused image pair we choose a picture from our dataset. The results presented in figure 4.

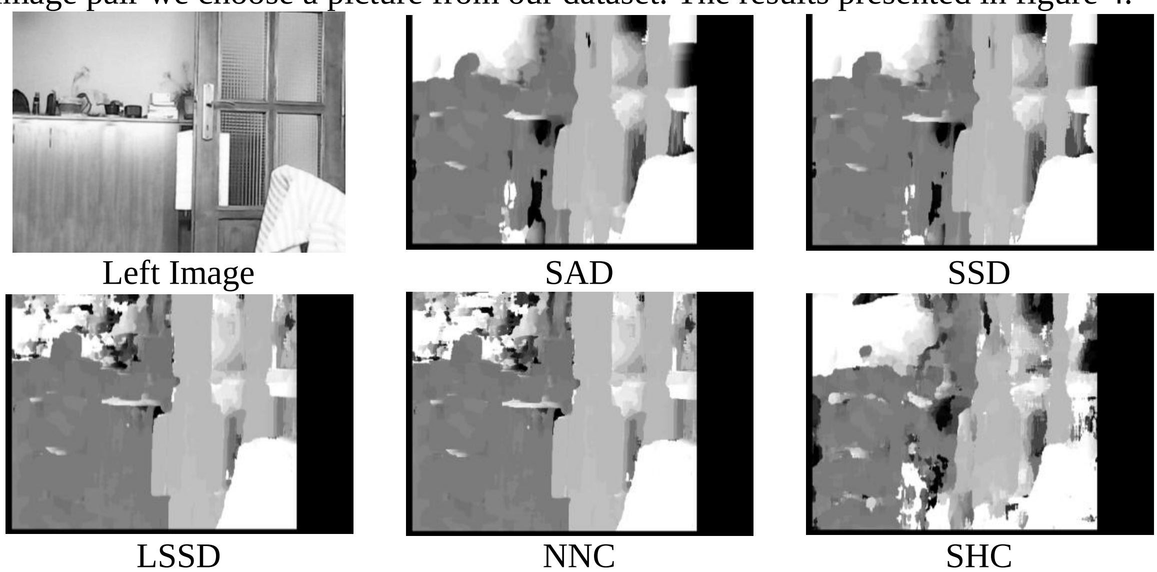

FIG 4
Depth map calculation using the methods presented above on own dataset.

It can be stated that SAD is one of the best algorithms in ideal illumination conditions; however in saturated image it loses precision. However NCC algorithm is more robust to the linear variation in the brightness due to different illumination conditions to the cameras; but the complexity of the algorithm leads to an increase in computational time compared to SAD or SSD.

CONCLUSION

In ideal illumination conditions all the algorithms are working similarly, increasing the search window size decrease the sharpness of the image but reduces the mismatches. In non-ideal illumination conditions the NCC outperforms all the other algorithms in precision but has the limitation of high demand on computation time. For real-time stereo vision the best correlation-based method is the SAD and SSD.

FUTURE SCOPE

This paper only deals with the methods of aggregation, used in most of the real-time stereo vision algorithms. But along within the presented methods the step of the cost computation and disparity selection are also important. Improving these steps make the stereo system more efficient. Our next step is to create an embedded stereo vision system on FPGA and to implement the presented algorithms on it in real-time applications like face detection or 3D mapping.

REFERENCES

1.S. T. BARNARD, W. B. THOMPSON: ***Disparity Analysis of Images***, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-2, no. 4, July 1980
2. N. LAZAROS, G. C. SIRAKOULIS, A. GASTERATOS:***Review of Stereo Vision Algorithms: From Software to Hardware***, International Journal of Optomechatronics, vol. 2:435-462,2008
3. T. KANADE, H. KANO, S. KIMURA: ***Development of a video-rate stereo machine***, Image UnderstandingWorkshop, Monterey,CA, 1994, p. 549–557.
4. D. SCHARSTEIN, R. SZELISKI, ***A taxonomy and evaluation of dense two-frame stereo correspondence algorithms***, International Journal of Computer Vision, vol. 47(1/2/3), pp. 7-42, Apr. 2002.
5. R. SZELINSKI, R. ZABIH: ***An Experimental COmparison of Stereo Algorithms***, International Workshop on Vision Algorithms, 1-19, Kerkyra, Grece, 1999
6. B. CYGANEC, J. P. SIEBERT: ***An Introduction to 3D Computer Vision Techniques and Algorithms***, John Wiley and sons, 2009
7. Middlebury standard dataset for evaluation, Tsukuba image. Available: http://vision.middlebury.edu

Acknowledgement: *This paper is supported by the Sectoral Operational Programme Human Resources Development POSDRU/159/1.5/S/137516 financed from the European Social Fund.*