Disparity Map Estimation using SAD and SSD algorithm

Mandar Kharde – 3574960 – st179460@stud.uni-stuttgart.de

Gopal Panigrahi – 3574876 – st179549@stud.uni-stuttgart.de

Rucha Golwalkar – 3575998 – st179996@stud.uni-stuttgart.de

1. Introduction

Disparity map calculation is an essential step in computer vision and stereo image processing. It refers to the calculation of the difference in position of corresponding pixels between two stereo images. The result of disparity map calculation is a 2D image, where each pixel value represents the disparity, or difference in position, between the corresponding pixels in the two stereo images. The main aim of disparity map calculation is to find the depth information of a scene captured by the stereo cameras.

There are two commonly used algorithms for disparity map calculation: Sum of Squared Differences (SSD) and Sum of Absolute Differences (SAD). SSD measures the difference between two image regions by calculating the sum of squared differences between corresponding pixels, whereas SAD measures the difference by calculating the sum of absolute differences between the same pixels. Both algorithms are used to find the best match between pixels in the two stereo images, and the resulting disparity map is then used to estimate the depth of the scene.

The choice of algorithm to use depends on the specific requirements of the application. SSD is computationally intensive and often slow, but it provides better accuracy for noisy images. On the other hand, SAD is faster and requires fewer computational resources, but it is sensitive to noise and may not provide accurate results in noisy images. In any case, both algorithms play a crucial role in stereo image processing and disparity map calculation.

The objective of this project is to perform disparity map estimation using an efficient algorithm on GPU as well as CPU. The tasks performed during the development of this project are as follows:

a. Implementation of Disparity map estimation on CPU using SSD algorithm.
b. Implementation of Disparity map estimation on CPU using SAD algorithm.
c. Implementation of Disparity map estimation on GPU using SSD algorithm.
d. Implementation of Disparity map estimation on GPU using SAD algorithm.
e. Comparison between the speed-up of GPU and CPU for both the algorithms.

2. Algorithm

We have used the SSD and SAD algorithm to perform the disparity map estimation on GPU. The same algorithms were used to calculate the disparity map on CPU.

a. Sum of Squared Differences (SSD) algorithm

The Sum of Squared Differences (SSD) algorithm is used for disparity map estimation in stereo vision. It compares the intensities of corresponding pixels between two images, and calculates the difference between them. The disparity map is created by finding the minimum SSD value between the left image and a search window in the right image for each pixel in the left image. The position of the minimum SSD value in the search window corresponds to the disparity value for that pixel. The algorithm estimates the disparity map by finding the closest match of pixels in the right image for each pixel in the left image, based on their intensity differences. The final disparity map is used to calculate the 3D position of each pixel in the scene, providing a depth map of the scene.

b. Sum of Absolute Differences (SAD)

The Sum of Absolute Differences (SAD) algorithm is a variation of the SSD algorithm for disparity map estimation in stereo vision. It calculates the difference between intensities of corresponding pixels in the left and right images, but instead of squaring the differences, it takes the absolute value. The disparity map is created by finding the minimum SAD value between the left image and a search window in the right image for each pixel in the left image. The position of the minimum SAD value in the search window corresponds to the disparity value for that pixel. The algorithm estimates the disparity map by finding the closest match of pixels in the right image for each pixel in the left image, based on their intensity differences. The final disparity map is used to calculate the 3D position of each pixel in the scene, providing a depth map of the scene.

3. Project contribution of team members

Implementation is available in the following GitHub repository:

https://github.com/gopalpani96/GPU_DisparityMap_Group1.git

Team member	Task
Mandar Kharde	• Literature review • Finalization of algorithm • GPU SAD implementation • Testing and Debugging
Gopal Panigrahi	• Literature review • GPU SSD implementation • Testing and Debugging • Analysis and review of results
Rucha Golwalkar	• Literature review • CPU SAD and SSD implementation • Testing and Debugging • Report writing

4. Results and Discussion

Figure 1. Left camera input image

Figure 2. Right camera input image

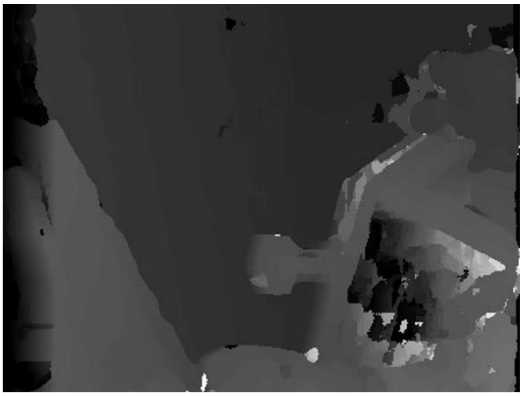

Fig 3. Disparity map using SAD on CPU

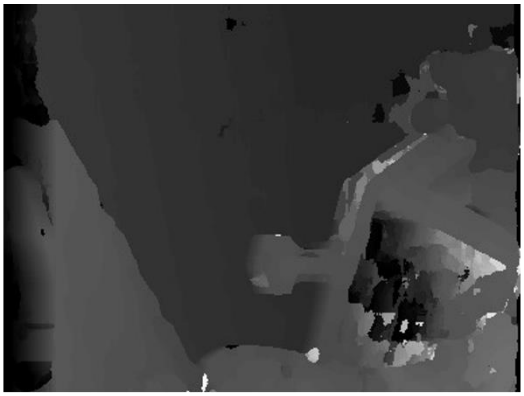

Fig 4. Disparity map using SAD on GPU

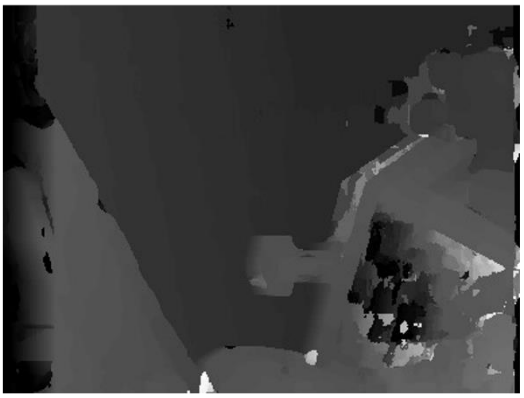

Fig 5. Disparity map using SSD on CPU

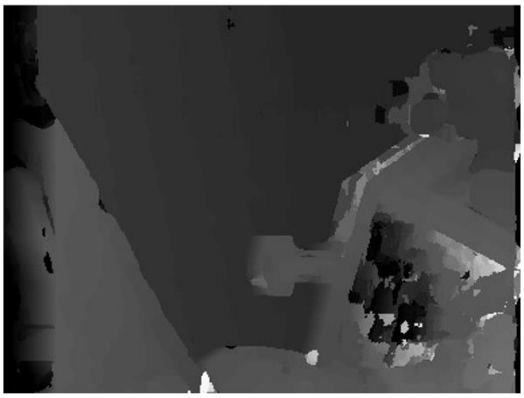

Fig 6. Disparity map using SSD on GPU

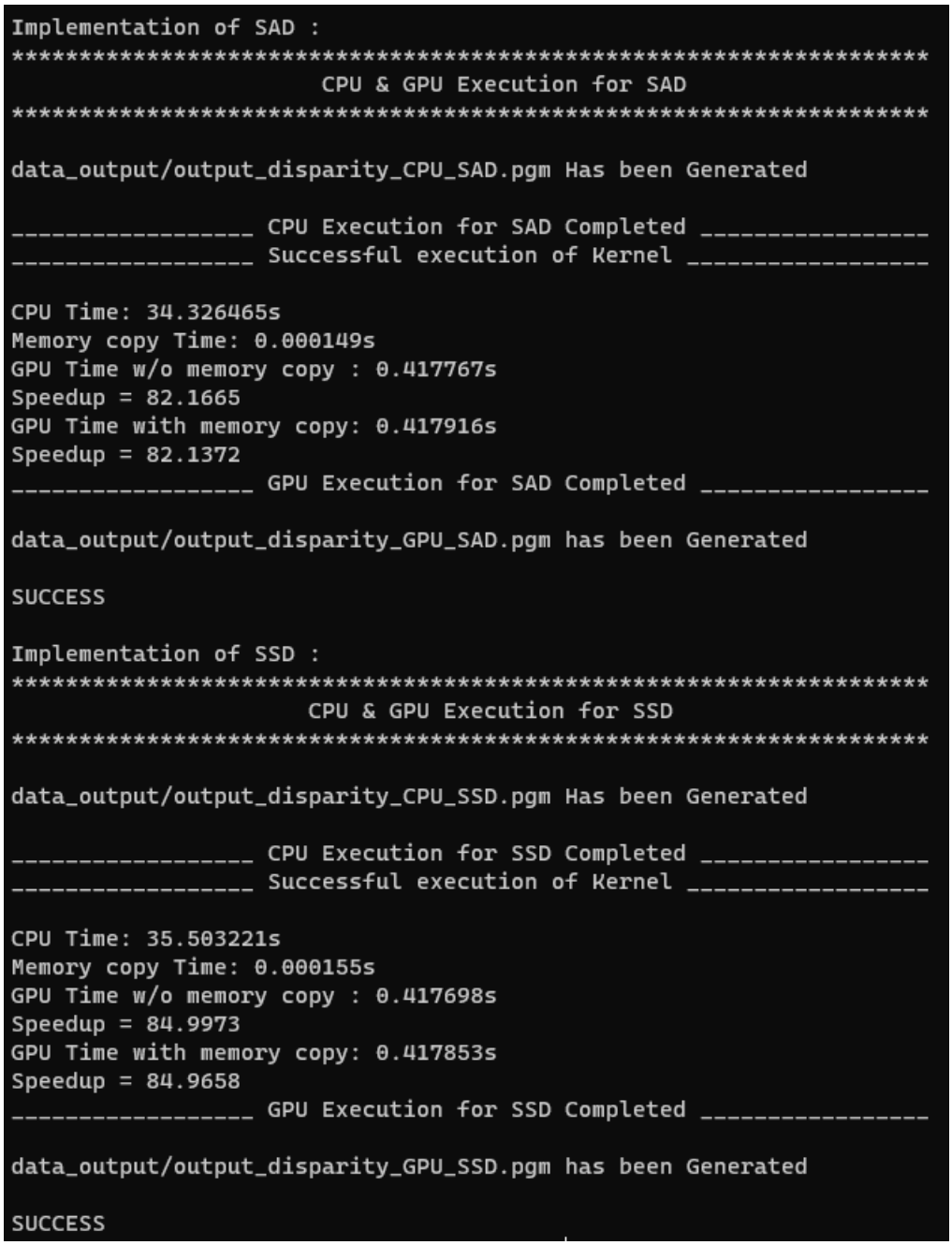

Fig 7. Console of Implementation of Disparity Map Estimation using SSD and SAD algorithm.

The results show that implementation of disparity map estimation on GPU is significantly faster as compared to that on CPU for SSD and SAD algorithm. The implementation of disparity map estimation using the SSD algorithm is faster on GPU than the SAD algorithm while the CPU executes disparity map estimation faster for the SAD algorithm than the SSD algorithm.

5. Conclusion

The performance of the GPU is significantly faster than that of the CPU. It takes lesser computation time to execute the same task as the CPU due to the capability of parallel processing.

6. References

[1] Hwang, Jae-Jeong & Wu, Hong Ren. (2011). Stereo Image Quality Assessment Using Visual Attention and Distortion Predictors. TIIS. 5. 1613-1631.

[2] Scarpino, Matthew. OpenCL in action: how to accelerate graphics and computations. Simon and Schuster, 2011.

[3] Gasim Mammodov, Exercise 3, Sobel filter in Opencl, High Performance Programming with Graphic Cards Lab Course

[4] Ambrosch K., Humenberger M., Kubinger W., Steininger A. Hardware Implement of an SAD Based Stereo Vision Algorithm. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; Minneapolis, MN, USA. 17–22 June 2007; pp. 1–6.

[5] Aboali, Maged & Abd Manap, Nurulfajar & Darsono, Abd & Yusof, Zulkalnain. (2017). Performance Analysis between Basic Block Matching and Dynamic Programming of Stereo Matching Algorithm. Journal of Telecommunication, Electronic and Computer Engineering. 9. 2289-8131.