CS-3D-Api | Semi-global matching
An alternative 3D reconstruction method

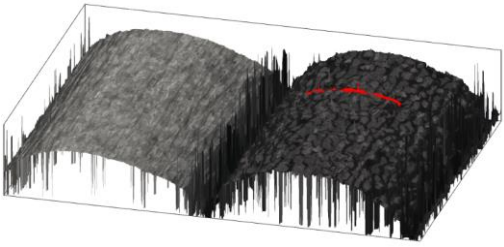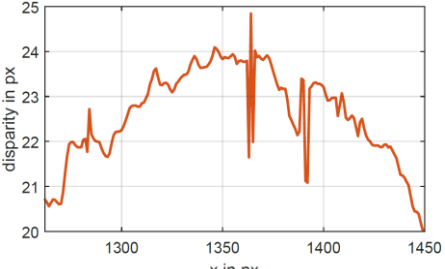*Technical note*
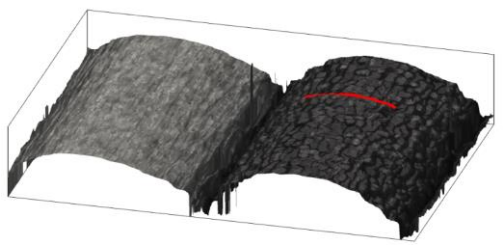
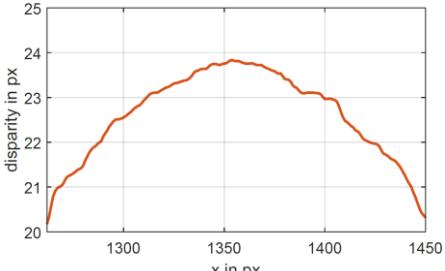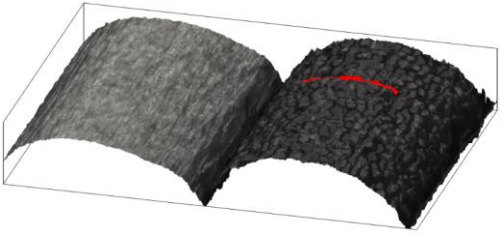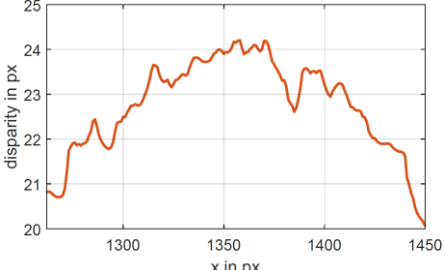Authors: Jia Eckhard, Robin Zirkler, Timo Eckhard

1. Introduction

CS-3D-Api now has a GPU implementation of an alternative 3D reconstruction method to the existing block-matching (BM) approach. The approach is based on the semi-global matching (SGM) algorithm[1], which is very accurate at object borders, fine structures and in weakly textured terrain. It combines concepts of global and local stereo methods for accurate, pixel-wise matching at low runtime.

The following examples illustrate some applications for which SGM outperforms BM in terms of reconstruction quality:

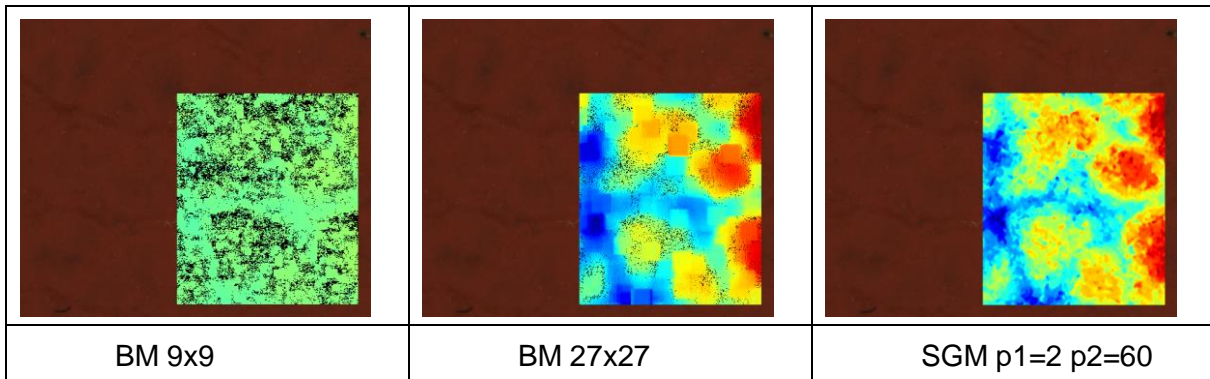- ***Reduction of height reconstruction artefacts on highly textured regions:***

For the following scan image data of insulation tubes, BM with small window size results in height reconstruction artefacts that can be accounted for by increasing the matching window size (from 9x9 px to 27x27 px). However, resulting from that is an unwanted low-pass filter effect on the reconstructed image disparity, removing the fine local height deviations on the insulation tubes. The measurement is therefore faulty. To illustrate this effect more clearly, we have plotted disparity profiles corresponding to the image regions marked in red:

		BM 9x9
		BM 27x27
		SGM p1=3 p2=100

---

[1] H. Hirschmüller, "Stereo Processing by Semi-Global Matching and Mutual Information," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 2, pp. 328–341, February 2008

- ***Surface with weakly textured terrain:***

In the following illustration of scan data of a 3D textured technical fabric, the false color overlay indicates the reconstructed image height. Black values in the BM result image indicates pixel positions for which height could not be reconstructed due to missing texture information. By using a larger block matching window size, the amount of pixel that cannot be reconstructed is reduced considerably but blocking artefacts typical for BM are introduced. Using SGM, detailed image height is reconstructed without artefacts.

BM 9x9	BM 27x27	SGM p1=2 p2=60

2.   A brief comparison of BM and SGM

The main difference between BM and SGM is the way a matching cost is computed at one hand, and the way the cost volume is processed on the other hand. The cost in BM is computed with normalized cross-correlation, where our implementation of SGM uses a special Census transformation.

The cost volume entries contain the costs for every pixel in the stereo image and each possible disparity value in the disparity search range. Cost volume processing (ie. to identify the disparity) in BM is realized by selecting for every pixel the disparity with minimum cost, without considering the surrounding neighborhood.

In SGM, the cost volume is processed along multiple symmetric concentric paths for each image pixel. Each path sums the cost for reaching a pixel with a certain disparity and the overall costs is computed as a sum over all paths. The SGM cost function contains a regularization parameter to penalize global discontinuities (ie. a smoothness constraint) and a parameter to penalize small disparity steps, for instance occurring on slanted surfaces. By making use of this regularization, prior knowledge about the surface structure can be incorporated, which makes the algorithm very robust.

3.   System requirements

Additional to the current requirements (see CS-3D-API manual chapter 2), the following is recommended to run the new method:

- GPU with CUDA compute capability version at least 3.5
- GPU with at least 3GB RAM

The GPU memory usage scales linearly with the search range and the width of the disparity image, and 3 GB may not be enough for some large data computation settings.

4. Using SGM in CS-3D-API

A new configuration parameter called "matchingMethod" has been introduced that allows to switch the 3D reconstruction method between block matching and SGM.

To alter this configuration parameter via API, you first need to get a copy of the active config3DApi-Object (chapter 6.3) and set the attribute "matchingMethod" to the following values:

- BLOCK_MATCHING
- SEMI_GLOBAL_MATCHING

To alter this configuration parameter via configuration file, you can set parameter matchingMethod = BLOCK_MATCHING or matchingMethod = SEMI_GLOBAL_MATCHING

If this parameter does not exist in the configuration file, you can add it under section [C3D-API].

The rest of the calculation procedure is the same for block matching and SGM.

5. Parameters

There are three parameters which the user can adjust to influence the result and speed of 3D reconstruction with SGM method.

- **heightRangeMode: defines the search range mode for height values**

    User can choose between fixed height range or dynamic height range.

    With fixed height range, the internal disparity search range is fixed to 128, starting from dStart, and the output height map is truncated by dEnd. Thus, the maximum height search range is 128 pixels. If the input height range (dEnd-dStart) is larger than 128, input dEnd is ignored and set to dStart+128 automatically.

    With flexible height range, any height range is accepted for calculation, and height range is dEnd - dStart. To alter this parameter via API, you can set the attribute "heightRangeMode" to the following values:

    - FIXED_RANGE
    - DYNAMIC_RANGE

    To alter this configuration parameter via configuration file, you can set parameter heightRangeMode = FIXED_RANGE or heightRangeMode = DYNAMIC_RANGE. If this parameter does not exist in the configuration file, you can add it under section [C3D-API].

- **p1: penalty value for small disparity change (i.e. 1 pixel disparity)**

    Using a lower penalty for small changes permits an adaptation to slanted or curved surfaces. p1 is in range between 0 and 31. Default value is 7.
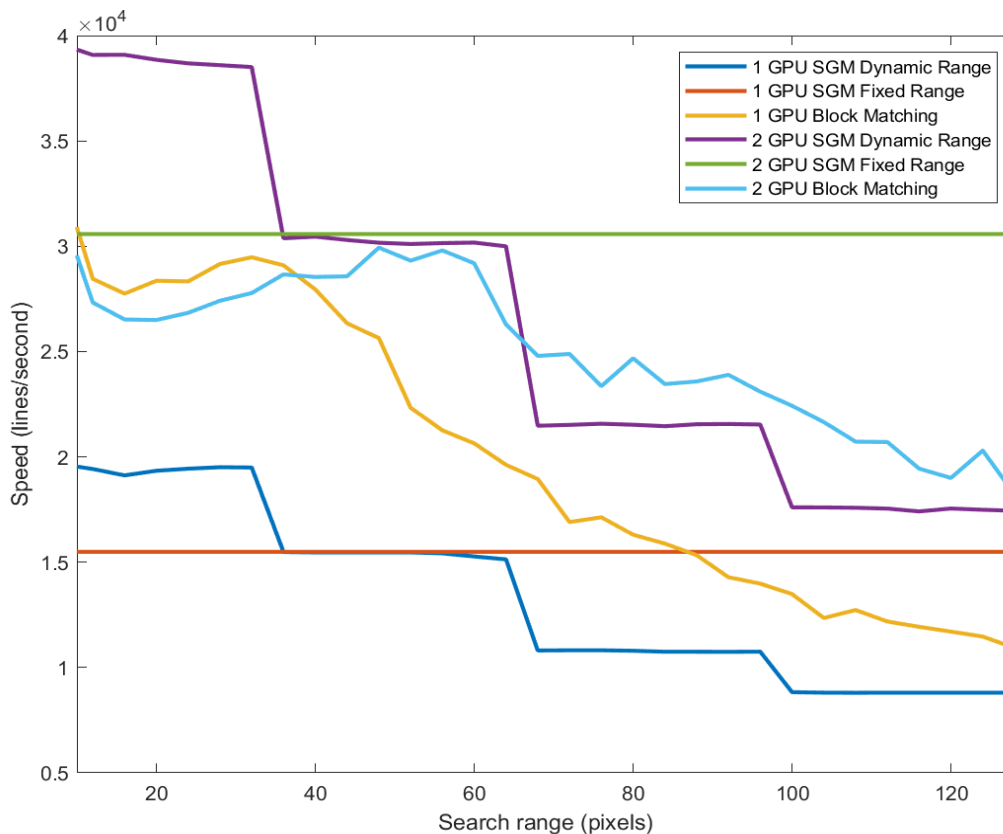
- **p2: penalty for all larger disparity changes (i.e. larger than 1 pixel disparity)**

Penalizes discontinuities in height. Such discontinuities are often visible as intensity changes. It always has to be ensured that p2 ≥ p1. p2 is in range between 0 and 127. Default value is 100.

6.   Performance

The performance of different matching methods and parameters for compact camera and dual camera on our test system is illustrated in the following plot.

Compact camera:

The system has following specifications:

- Operating System: Windows 10 Pro, 64-bit
- Processor: Intel i9, CPU @ 3.10GHz
- CPU RAM: 64 GB
- Number of GPUs: 2
- GPU1: GeForce GTX 1080 Ti
- GPU2: GeForce GTX 1080 Ti
- Driver version: 431.60

We can see that with SGM method, using 2 GPUs clearly outperforms using only 1 GPU and is highly recommended. With large search range (larger than 64 for example), fixed height range mode outperforms dynamic height range mode.

Note that performance is very platform dependent and may vary considerably in other systems.