Semi-Global Matching in Object Space

F. Bethmann [a, *], T. Luhmann [b]

Jade University of Applied Sciences Oldenburg, Ofener Straße 16/19, Oldenburg, Germany
[a] folkmar.bethmann@jade-hs.de, [b] thomas.luhmann@jade-hs.de

Commission III, WG III/1

KEY WORDS: image matching, semi-global matching, multi-image matching

ABSTRACT:

Semi-Global Matching (SGM) is a widespread algorithm for image matching which is used for very different applications, ranging from real-time applications (e.g. for generating 3D data for driver assistance systems) to aerial image matching. Originally developed for stereo-image matching, several extensions have been proposed to use more than two images within the matching process (multi-baseline matching, multi-view stereo). These extensions still perform the image matching in (rectified) stereo images and combine the pairwise results afterwards to create the final solution. This paper proposes an alternative approach which is suitable for the introduction of an arbitrary number of images into the matching process and utilizes image matching by using non-rectified images. The new method differs from the original SGM method mainly in two aspects: Firstly, the cost calculation is formulated in object space within a dense voxel raster by using the grey (or colour) values of all images instead of pairwise cost calculation in image space. Secondly, the semi-global (path-wise) minimization process is transferred into object space as well, so that the result of semi-global optimization leads to index maps (instead of disparity maps) which directly indicate the 3D positions of the best matches. Altogether, this yields to an essential simplification of the matching process compared to multi-view stereo (MVS) approaches. After a description of the new method, results achieved from two different datasets (close-range and aerial) are presented and discussed.

1. INTRODUCTION

Semi-Global Matching (Hirschmüller, 2005) has proven to be a powerful stereo matching algorithm which is used for a variety of applications and measurement tasks, ranging from close-range and real-time applications to aerial image matching. It has become widespread especially due to several advantages compared to other matching algorithms: It is very robust and reduces large outliers in low or non-textured areas while preserving edges and sharp object boundaries. It allows for the use of pixel-wise cost functions and is therefore able to resolve fine spatial structures on the object surface. Further on, it is almost independent of task dependent parameter settings and so it reduces efforts for the adaption of matching parameters for a special measurement task, avoids unsuccessful test runs and can be used in black box solutions. Finally, it can be implemented very efficiently in terms of computing time by using hierarchical matching strategies and techniques of parallelization on special hardware (GPU, FPGA) (Banz et al., 2010)(Buder, 2012)(Ernst & Hirschmüller, 2008)(Michael et al., 2013). All in all SGM can be regarded as a good compromise between highly accurate but less robust image matching techniques and robust but time-consuming global matching methods.

For a number of applications it is sufficient to use stereo cameras for image matching. This is especially true for many applications in computer vision (e.g. stereo cameras in assistance systems) where the need for real-time results is more important than high accuracies. On the other hand, various tasks focus on the accurate and complete 3D reconstruction of complex scenes (e.g. for aerial image matching, in fields of cultural heritage, archaeology, industrial measurements and so on). For these purposes, dense surface matching has been extended to so-called multi-baseline matching as proposed e.g. in Hirschmüller (2008) or multi-view stereo algorithms as proposed e.g. in Rothermel et al., (2013) and Wenzel et al. (2013). Multi-baseline matching performs stereo matching by SGM between a base image and all match images. Further on, invalid disparities are removed by consistency checks (left-right check) and all stereo matching results are combined by selecting the median value of all disparities for each pixel. Afterwards, the accuracy can be increased by calculating the weighted mean of all correct disparities, i.e. all disparities within an interval of e.g. 1 pixel around the median.

The multi-view stereo algorithm in Rothermel et al. (2013) performs stereo matching for all overlapping image pairs or at least for a selection of these. After removing outliers by left-right consistency checks an additional outlier elimination is performed by checking for geometric consistency in object space – considering uncertainty ranges that have been derived by error propagation. Finally, all corresponding image coordinates of each object point are used for triangulation to calculate the final 3D coordinates.

Both, multi-baseline matching as well as multi-view stereo approaches perform the matching in image pairs and do not allow for multi-image matching.

A method for multi-image matching by using facets in object space and simultaneous adjustment of DSM, orthophoto and parameters of a reflexion model was proposed by Wrobel (1987) and further on extended and modified by Weisensee (1992), Schlüter (2000) and Wendt (2002).

Another method for multi-image matching that uses Least-Squares Matching (LSM) was proposed by Grün (1985) and Grün & Baltsavias (1988). This method extends LSM by introducing epipolar- or collinearity constraints to the equation system so the search range is limited to epipolar lines.

* Corresponding author

Both approaches, object-based matching with facets as well as LSM are using non-linear functional models within the adjustment and therefore need an approximate representation of the surface.
Within this paper, an approach for multi-image matching is proposed that uses a semi-global optimization strategy and is therefore applicable without any a priori knowledge about the object surface. Compared to SGM, the proposed method is mainly characterized by transferring the matching procedure from image into object space. This has several advantages compared to SGM in MVS.
Firstly, in pairwise matching the subsequent consistency checks can be performed more easily because the matching directly leads to 2.5D coordinates in object space. A transfer of disparity maps from image to object space is not necessary anymore and consistency has to be checked in Z-direction only.
Secondly, it is not required to rectify the images before the matching. Since in MVS every image has to be rectified several times for the use in different image pairs the new approach leads to a reduction of processing steps.
Thirdly, the whole voxel grid can be subdivided into smaller parts with the size of each part being easily adapted to the available memory space. Hence, the algorithm can process very large datasets even with standard hardware.
Finally, it is possible to correlate images not only pairwise but also to perform real multi-image matching which is not provided by existing MVS approaches.
Further advantages of the new approach will be discussed in the following sections. In chapter 3 the results of two different datasets (close-range object and aerial image setup) are discussed.

2. OBJECT-BASED MULTI-IMAGE SEMI-GLOBAL MATCHING (OSGM)

Within this chapter the method of object-based multi-image semi-global matching will be described in detail (section 2.2 to 2.9). In advance, a short review of SGM is given in section 2.1.

2.1 Review of SGM

The SGM method as originally described in Hirschmüller (2005) proposes an intelligent solution for the approximate minimization of global 2D energy functions as they are used e.g. within global image matching methods. SGM uses the following energy function:

$$E(D) = \sum_p C(p', D_P) + \sum_{q \in N_p} P_1 \cdot T[|D_p - D_q| = 1] + \sum_{q \in N_p} P_2 \cdot T[|D_p - D_q| > 1] \quad (1)$$

The first term of (1) contains the matching costs C between a pixel p' in image 1 and a potential corresponding pixel p'' in image 2 (at a specific disparity D_p). The second term adds a penalty P_1 for the current disparity D_P to the cost value C if the difference between D_P and the disparity D_q at a neighbouring pixel q is 1 (the function T returns 1 if $|D_p - D_q|=1$ and 0 in all other cases). The second term adds a larger penalty to the cost value C if the difference exceeds 1 (the function T returns 1 if $|D_p - D_q|>1$ and 0 in all other cases).
First step in SGM is the cost calculation to build up the structure $C(p', D_p)$ in equation (1) by calculating the matching costs between every pixel p' in the first image and all potentially corresponding pixel p'' in the second image. Using rectified image pairs the relationship between p' and p'' can be expressed by the parallax or disparity D with $p''(x''=x'-D, y''=y')$.

For calculating the matching costs different cost functions can be used, varying from very simple block matchers (e.g. differences of absolute intensity values (SAD)) to sophisticated pixel-wise approaches (e.g. mutual information as described in Hirschmüller (2005)). An analysis of different cost functions is not addressed in this paper but can be found e.g. in Hirschmüller and Scharstein (2007).
Second step in SGM is cost aggregation. The main idea of SGM is to utilize cost aggregation not in all directions (which would be necessary for a rigorous global solution) but in the direction of r=16 or at least r=8 paths L_r. Cost aggregation can be done recursively and separately for each path L_r with

$$L_r(p', D) = C(p', D) + \min(L_r(p'-r, D), L_r(p'-r, D-1) + P_1, L_r(p'-r, D+1) + P_1, \min_i L_r(p'-r, i) + P_2) - \min_k L_r(p'-r, k) \quad (2)$$

with $(p', D) = (x', y', D)$.
The positions of adjacent pixels are defined separately for each path with p-r:

$$L_r(p'-r, D) = L_r(x'-u, y'-v, D) \quad (3)$$

(e.g. with u=1, v=0 for a path in x'-direction).

The expression in (2) searches the minimum path costs inclusive possibly added penalties P_1 and P_2 at the position of the previous pixel in path direction (p-r) and adds this minimum to the cost value $C(p', D)$) at the current pixel p' and the disparity D. The last term of (2) subtracts the minimum path cost of the previous pixel to avoid very large values in L_r.
The results of the cost aggregation for 8 (or 16) paths can be fused with

$$S(p', D) = \sum_{r=1}^{8,16} L_r(p', D) \quad (4)$$

Then the final disparity D is derived from (4) by searching the minimum in for each pixel p' in $S(p', D)$. The final disparity is stored for each pixel p' leading to a dense disparity map $D(p')$.

2.2 Semi-global matching in object space

In the proposed method the object space has to be subdivided into a dense voxel raster in a first step. Each voxel may be a cube or a cuboid. The size of the cuboids (ΔX, ΔY, ΔZ) defines the resolution in object space and should be adapted to the mean ground sampling distance (so that for each pixel one 3D point is estimated) as well as to the spatial configuration of the images (base-to-height ratios).
For transferring the semi-global optimization procedure from image into object space the global energy function in (1) has to be modified to

$$E(Z) = \sum_{X,Y} C(X, Y, Z) + \sum_{q \in N_p} P_1 \cdot T[|Z - Z_q| = \Delta Z] + \sum_{q \in N_p} P_2 \cdot T[|Z - Z_q| > \Delta Z] \quad (5)$$

The first term in (5) contains the matching costs for each voxel of the raster and the second and third term add penalties P_1 and P_2 in case of differences in Z-direction between adjacent voxels.

Therefore, the smoothness constraints of Semi-Global Matching effectuate a smoothing in the direction of a defined axis in object space. For classical 2.5D approaches this is the Z-direction of the global coordinate system. For 3D applications it is necessary to define a number of appropriate local coordinate systems and to assign a selection of appropriate images for matching to each local coordinate system. The matching is then performed within the local systems and all resulting point clouds have to be transformed into a global system afterwards.

2.3 Cost calculation in object space

In analogy to classical SGM the first step in OSGM is the calculation of the matching costs to build up the structure $C(X,Y,Z)$. Therefore, the centre of each voxel is re-projected into all images by using the collinearity equations and the grey (or colour) values of the corresponding image coordinates are used for cost calculation (Figure 1).

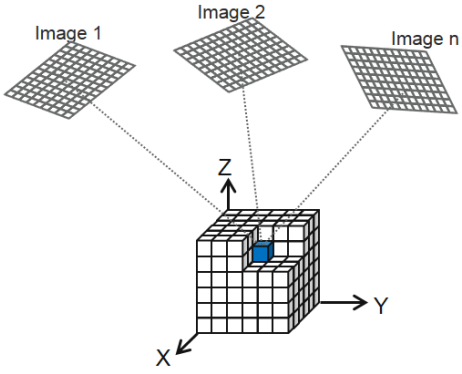

Figure 1. Calculation of matching costs for a voxel

If window-based cost functions are used (e.g. Census or normalized cross-correlation, NCC) the matching window is defined in object space parallel to a reference plane which is the X/Y-plane of the global (for 2.5D applications) or a local (for 3D application) coordinate system. In a next step the matching window is re-projected into all images and the grey or colour values underneath the re-projected windows are used for cost calculation. This is important because it leads to a high invariance against image rotations and (within limits) to different images scales. Therefore, images with larger baselines can be combined whereas stereo SGM typically uses image pairs with short baselines for pairwise matching.
Most of the common cost or similarity functions are designed for the calculation of the (dis)similarity between two signals (or respectively two images) and therefore are well-suited for pairwise image matching. Thus, for a combined cost calculation for n images it is necessary to think about sensibly extensions of cost or similarity functions for multi-image correlation. However, since pairwise image matching in multi-image bundles can be used for consistency checks and can therefore be regarded as an important tool for the reliable detection of occlusions and outliers, both strategies (pairwise image matching and combined multi-image matching) should be considered within the new approach.
Since the re-projection of the voxel centres or the matching window leads to sub-pixel coordinates within the images the SGM in object space leads directly to 3D points with sub-pixel accuracy. This is another advantage compared to the standard SGM in which sub-pixel accuracy is typically achieved by interpolating between neighbouring cost-values in disparity space, e.g. by quadratic curve fitting as suggested in Hirschmüller (2008).

2.4 Cost functions

Until now the focus was set on the development of the matching more than on implementing sophisticated cost functions.
A simple cost function which is often used for SGM is given by census Zabih and Woodfill (1994). Census is highly invariant against radiometric differences between the images and therefore leads to robust matching results. The cost parameter of census is given by the Hamming distance between two image windows. Hence, the maximum number of distinguishable cost values is equal to the maximum hamming distance h_{max} which depends on the window size (e.g. for a 5x5 window h_{max}=25).
Since changes of the centre coordinates of the voxels in Z-direction by small increments ΔZ lead to sub-pixel movements of the matching windows within the images, it is necessary to use a cost function that allows for the distinction between these sub-pixel movements. First investigations by using census have shown that it does not fulfil this requirement due to its limited resolution as described above.
Therefore, the normalized cross correlation (NCC) is used which is defined by

$$\rho_{fg} = \frac{\sigma_{fg}}{\sigma_f \cdot \sigma_g} = \frac{\sum (f_i - \bar{f})(g_i - \bar{g})}{\sqrt{\sum (f_i - \bar{f})^2} \cdot \sqrt{\sum (g_i - \bar{g})^2}} \tag{6}$$

In (6) σ_{fg} is the covariance between the grey values within the two image windows f and g and σ_f and σ_g are the variances of the grey values in the image windows. Since the coefficient ρ_{fg} is a measure of the similarity and SGM typically uses cost values for the description of the *dis*similarity, (6) is modified with

$$\rho = 1 - \rho_{fg} \tag{7}$$

The co-domain of the cost parameter ρ includes cost values between 0.0 (low matching costs, high similarity) and 2.0 (high matching costs, low similarity).

2.5 Cost aggregation in object space

For the minimization of (5) by adapting the semi-global approach the path-wise cost aggregation can be done in analogy to the standard SGM separately for every path L_r with

$$\begin{aligned} L_r(v,Z) = C(v,Z) + \min(& L_r(v-r,Z), \\ & L_r(v-r,Z-\Delta Z) + P_1, \\ & L_r(v-r,Z+\Delta Z) + P_1, \\ & \min_i L_r(v-r,i\cdot\Delta Z) + P_2)) \\ & - \min_k L_r(v-r,k\cdot\Delta Z) \end{aligned} \tag{8}$$

The expression in (8) is a modification of (2) in which v is used as substitution for the X,Y-coordinate of a voxel centre:

$$L_r(v,Z) = L_r(X,Y,Z) \tag{9}$$

The X,Y-position of adjacent voxels are defined separately for each path with v-r:

$$L_r(v-r,Z) = L_r(X - u\cdot\Delta X, Y - v\cdot\Delta Y, Z) \tag{10}$$

(e.g. with u=1, v=0 for path r=1, see Figure 4).

The expression in (8) searches the minimum path costs including possibly added penalties P_1 and P_2 at the position of the previous voxel in path direction (v-r) and adds this minimum to the cost value $C(X,Y,Z)$ of the current voxel. The penalty P_1 is added if the difference in Z-direction between the current voxel and the adjacent voxel is equal to ΔZ (which is the height of one voxel) and P_2 is added if the difference in Z-direction is larger than ΔZ. The last term of (8) subtracts the minimum path cost of the previous voxel to avoid very large values in L_r. The paths of minimum costs are illustrated exemplarily in Figure 2.

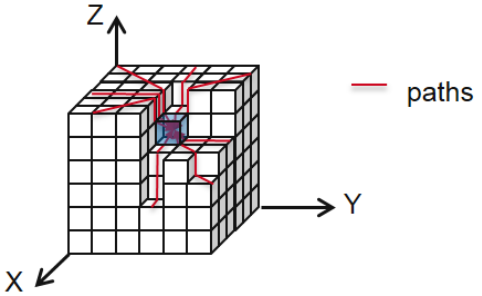

Figure 2. Paths of minimum costs

Analogue to (4) the results of the cost aggregation for 8 (or 16) paths can be fused with

$$S(v,Z) = \sum_{r=1}^{8,16} L_r(v,Z) \tag{11}$$

The matching result then is derived from (11) by searching the minimum in $S(v,Z)$ for each v:

$$\min_Z S(v,Z) \tag{12}$$

The final Z-coordinate for each voxel v is equal to the position Z where $S(v,Z)$ reaches a minimum. The final value is stored in an index map $Z(v)$ for each voxel v (instead of a disparity map $D(p)$).

2.6 Hierarchical computation

As stated above the spatial resolution of the voxel grid is ideally adapted to the (mean) GSD of the images to create dense 3D point clouds. Therefore, the voxel grid contains a very high number of voxels. To speed up the computation time for calculation and aggregation of costs it is reasonable to use a hierarchical approach (matching in image pyramids). The image pyramids are built up in a typical way by applying a low pass filter on the original images (Gaussian) and bisecting the resolution of the images from one pyramid level l to the next one (l+1). Accordingly, the resolution of the voxel raster has to be bisected as well by doubling the voxel sizes with $\Delta X_l = \Delta X_{l+1} \cdot 2$, $\Delta Y_l = \Delta Y_{l+1} \cdot 2$ and $\Delta Z_l = \Delta Z_{l+1} \cdot 2$.
The process of matching in image pyramids is shown in Figure 3:

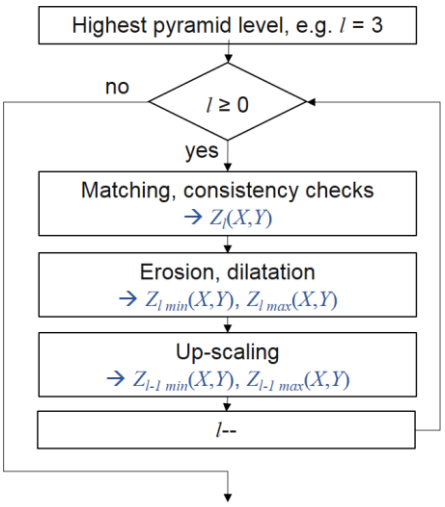

Figure 3. Hierarchical matching

The initial matching is performed in the highest level (low resolution) of the pyramid without any limitation of the search range. Afterwards, the matching result is analysed by deriving minimum and maximum Z-values by applying erosion and dilatation operators on the index map $Z_l(X,Y)$ for the current pyramid level l. For both, erosion and dilatation, windows with a fixed size of 7x7 elements are used for each pyramid level. Larger windows sizes extend the search ranges in case of high variations in Z-direction (e.g. in urban areas).

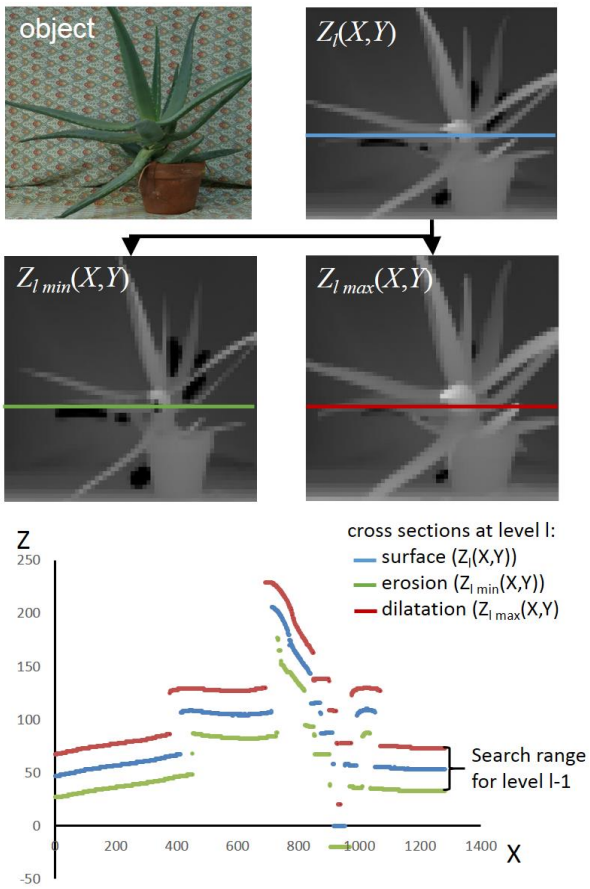

Figure 4. Limitation of search ranges

Further on, the results of erosion and dilatation are shifted in Z-direction with a constant offset of $-2 \cdot \Delta Z_l$ and $+2 \cdot \Delta Z_l$,

respectively, to avoid too strong limitations of the search range in areas of local maxima and minima on the object surface. Finally, the results of erosion and dilatation for a pyramid level *l* are up-scaled for the next lower pyramid level *l*-1, where they are used as limitations of the search range. Figure 4 shows exemplarily the results of erosion and dilatation as well as the derived limitation of the search range for a cross-section of the object surface.

2.7 Consistency checks

The new approach allows for pairwise matching as well as for multi-image matching. In pairwise matching the cost structure $C_i(X,Y,Z)$ has to be built up for each image pair *i* (with $i=n\cdot(n-1)/2$ and *n*=number of images). Afterwards, for each pair a matching result is generated by semi-global optimization in object space and all pairwise results can be used for consistency checks, aiming for the detection of occlusions and the elimination of outliers. Therefore, for the matching result of each pair uncertainty ranges are defined (in Z-direction, see Figure 5) and the number *c* of all results from other image pairs that are lying within an uncertainty range is determined. If *c* is exceeds a threshold *t* the matching is assumed to be consistent and the final result is set equal to the mean of all Z-values within the uncertainty range.
Figure 5 illustrates the procedure. The results of matching in pair 3 and 4 are within the uncertainty range of the matching in pair 1 and the result of pair 2 is outside. If, for example, *t* is set to 2 the results of matching in pair 1, 3 and 4 would be assumed to be consistent. The final Z-coordinate is then set to the mean Z-value of pair 1, 3 and 4.
The threshold *t* can be adapted automatically related to the number of images *i* in which the current voxel is visible, e.g. $t = i / 2$). The uncertainty range can be estimated by simple error propagation using a priori knowledge about base-to-height ratios and image matching quality.

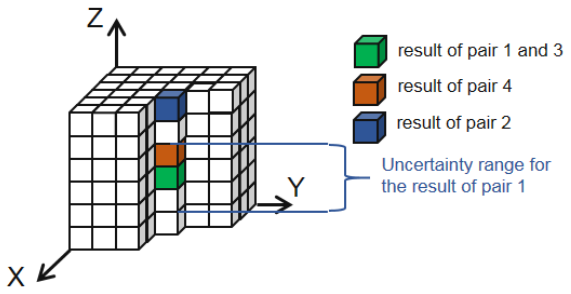

Figure 5. Consistency checks

The approach generally allows for the estimation of more than one Z-value for each X-Y-coordinate. For each pairwise matching result the consistency to all other results is checked and the consistency criterion could be fulfilled for different Z-values. This will be the case in partly occluded areas where voxels in different heights are identified, which are not visible in all image pairs.

2.8 Image selection for block-wise matching

For processing larger image bundles it is helpful to sort the images in advance and to assign them to different segments of the surface. We use a relatively simple approach to complete this task.

First step is to subdivide the voxel grid into small blocks (see Figure 6). The maximum size of a block is derived from the available memory space as well as from the amount of overlap of the images.

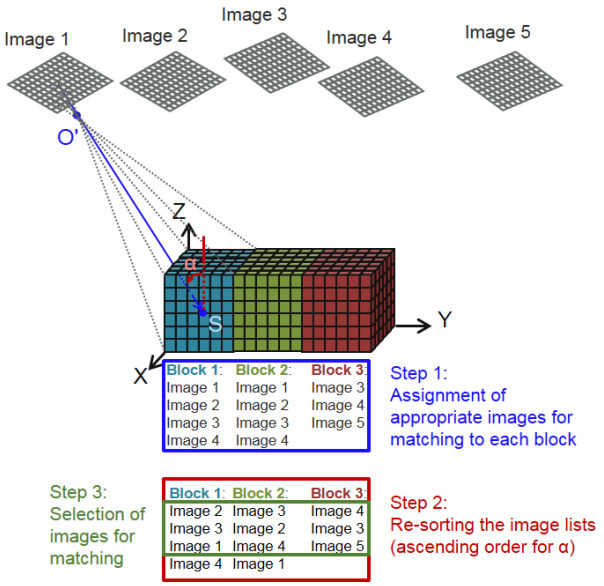

Figure 6. Selection of images for block-wise matching

In a second step the eight corners of each block are re-projected into all images. Every image in which a block is entirely visible is assigned to this block, resulting in a list of images for each block which can be used for matching.
Since the robustness of matching decreases with increasing baselines and the probability of occlusions may increase as well, it is sensible to prioritize the images within the lists for each block. For this, the vectors between the projection centres (O') and the centre point of the block (S) and the angles α between these vectors and the reference axis for SGM (the Z-axis) are calculated. Afterwards, the image lists are re-sorted in ascending order for α.
The re-sorted lists gives by tendency a priority to image pairs with short baselines. If the number of images for matching should be limited – which is be useful for the processing of highly overlapping image data – the first images of the lists can be selected for matching (Step 3 in Figure 6).
Afterwards, each block is processed separately, considering overlaps between the blocks.

2.9 Parallel computing

The current version of the semi-global matching in object space is implemented by using multi-threading on CPU which leads to acceptable computing times even for large datasets. As mentioned before, a further increase of performance can be achieved by implementing the algorithm on highly paralleling hardware (GPUs).
For parallelization each block is subdivided into a number of smaller blocks corresponding to the number of available processor kernels. Further on, cost calculation can be done simultaneously for all blocks on the different kernels.
For cost aggregation each path is processed in one thread which is feasible because eight path directions are used and can be assigned to the eight kernels of the CPU.

3. RESULTS

Within this chapter results of OSGM are presented and discussed. Two different datasets have been processed: a close-range dataset and a set of aerial images of an urban area.

3.1 Close-range dataset

The close-range dataset contains 38 images capturing a small clay sculpture (figure 7). The size of the sculpture is about 80mm (height), 100mm (length) and 60mm (width). It has a slightly natural textured surface and continuous curved areas as well as fine geometrical structures in regions of the head and the back.

Figure 7. Test object

For the purpose of comparison the object was measured using a fringe projection system from multiple viewing directions. The results of all views are transformed into a global coordinate system by a set of coded targets for the orientation of the scanner (Figure 8). Later on, these coded targets are used also for orientation of the images so that both, the matching and the scanning result, are present in the same coordinate system. The accuracy of the scanner is specified with 20-50µm, depending on the properties of the surface and the resolution is 40µm (specification of the manufacturer). The scanning result leads to a surface representation (TIN) of about 630.000 triangles.

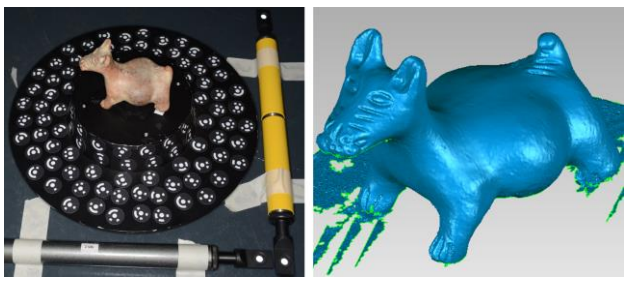

Figure 8. Reference point field (left) and result of fringe projection measurements (right)

The images for matching were captured with a calibrated Nikon D2x camera from different viewing directions. The mean distance to the object is about 550mm which leads to a mean ground sample distance of about 0.3mm. The spatial configuration of the image bundle is illustrated in figure 9. Finally, the matching has been performed by using all 38 images. The voxel size was set to $\Delta X=\Delta Y=\Delta Z=0.3$mm. As cost function the normalized cross correlation as described in section 2.3 and 2.4 and small matching windows with 5x5 elements were used.

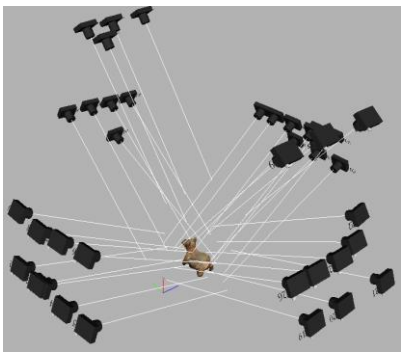

Figure 9. Spatial configuration of the image bundle

The point cloud achieved from matching consists of 110.000 points, computing time was approximately two minutes. The matching result is illustrated in Figure 10:

Figure 10. Result of OSGM, colour coded point cloud

The comparison between matching result and fringe projection measurement focuses on the evaluation of the matching accuracy. Figure 11 shows the colour-coded 3D deviations (shortest distance between each point to the TIN of the fringe projection measurement, calculated by Geomagic):

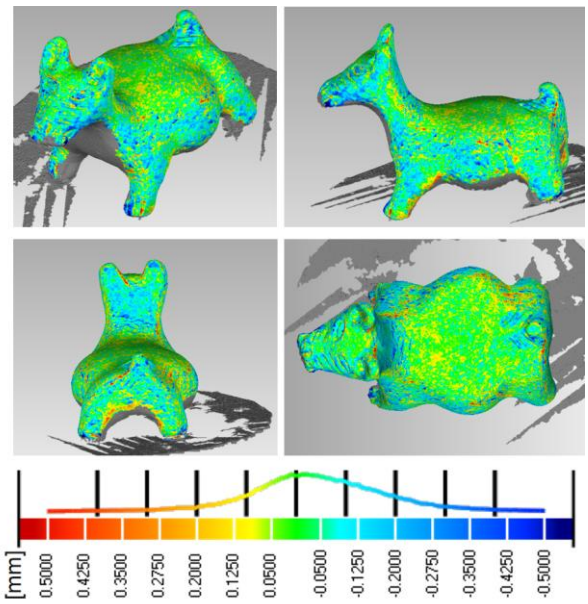

Figure 11. Comparison to fringe projection measurement

The standard deviation is about 0.16mm which corresponds to approximately half of the resolution and therefore can be regarded as a reasonably result. The fine structures in regions of the head (small grooves) are well resolved without smoothing which is due to the small matching windows. In some areas (e.g. at the front feet) the deviations reach up to 0.5 mm which cannot be fully explained until now. Altogether, the matching shows a very good result.

3.2 Aerial images

The second dataset consists of a selection of 10 aerial images, captured above an urban area by a DMC camera. Ground sample distance is 10cm and the images are overlapping by 60% in both directions. In the first instance, a squared area of about 500m x 500m was selected for the matching.

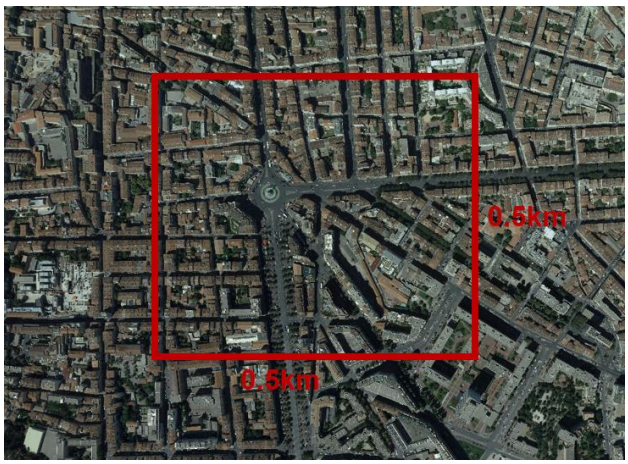

Figure 12. Test area for matching

The matching was initialized with a voxel size of $\Delta X=\Delta Y=\Delta Z=10$cm according to the GSD. As cost function within semi-global matching, normalized cross correlation is used, and the window size is set to 5x5 elements again. The matching generates a dense point cloud with about 25 million points.

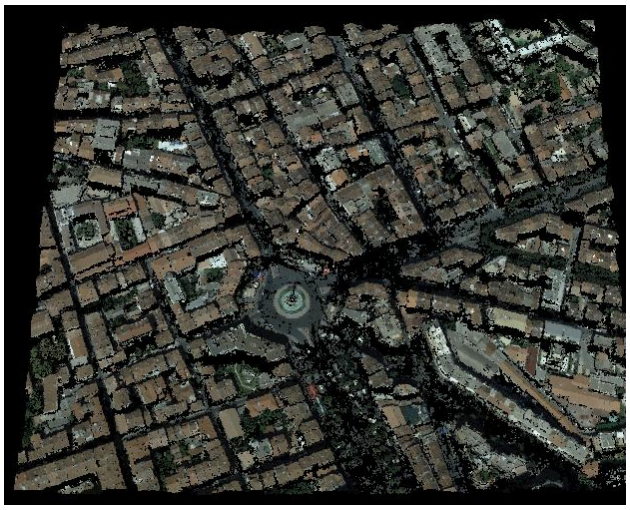

Figure 13. Dense point cloud, 25 million points

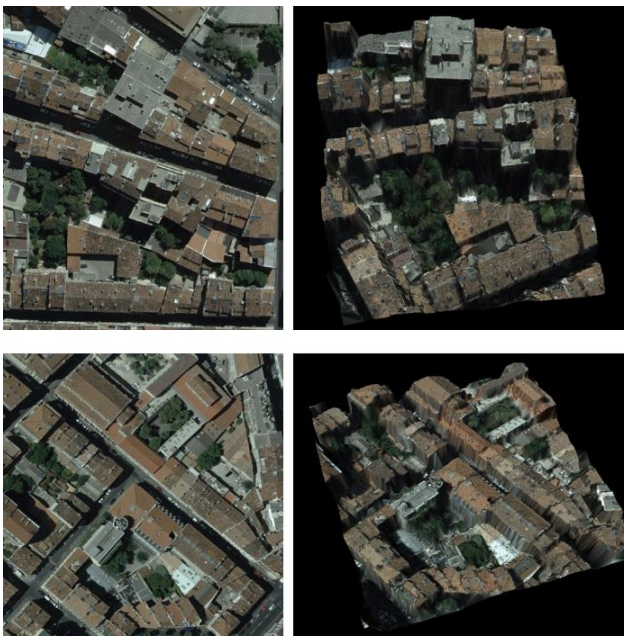

Figure 14. Details of a TIN derived from unfiltered point cloud (right) and corresponding image sections (left)

The 3D visualisation in figure 14 (right hand side) shows a TIN which has been derived from the unfiltered point cloud. On the left hand side the corresponding image sections are illustrated.
Obviously, the surface has been captured with lots of details (structures on the roofs, e.g. like dormers) and changes in height are modelled very well (e.g. borders of the buildings).
For an extensive evaluation of the quality of the matching result a comparison to high resolution LiDAR data is planned. A combined dataset with high resolution image data and high resolution LiDAR data will be available soon.

4. SUMMARY AND OUTLOOK

The presented modification of SGM is mainly characterized by transferring the process of cost calculation and path-wise cost aggregation from image into object space. Instead of estimating dense disparity maps, index maps are generated which directly indicate the best matches in 3D space.
The new approach was tested under laboratory conditions by using a test object with reference data of a fringe projection measurement as well as for a set of areal images. The tests show very promising results. The new method maintains the benefits of SGM (e.g. robustness in non-textured areas, good results at sharp object boundaries) and adds several advantages.
In opposite to most multi-baseline or multi-view stereo approaches the new approach works without rectified images and therefore reduces the efforts for pre-processing (no need for image rectification) and for post-processing (no need for the fusion of disparity maps). Further on, the new method allows for the integration of more than two images into the matching process and is therefore suitable for real multi-image correlation. All in all, the new algorithm has a clearly simplified structure compared to SGM in multi-view stereo approaches.
Further developments will focus on the implementation of pixel-wise cost functions to fully exploit the advantages of SGM. For an extensive evaluation of the matching quality especially for aerial images, new datasets containing high resolution images as well as LiDAR data will be processed.
Since the structure of the proposed method separates the process of cost calculation from special properties of image

sensors, extensions for the integration of other sensors (e.g. aerial or satellite based) should be considered. In addition, the integration of colour- or multi-spectral information into the matching process could be helpful for stabilizing the matching process.

ACKNOWLEDGEMENTS

The research has been supported by the Lower Saxony program for Research Professors, 2013-2016.

REFERENCES

Banz, C., Hesselbarth, S., Flatt, H., Blume, H. and Pirsch, P., 2010. Real-time stereo vision system using semi-global matching disparity estimation: Architecture and fpga-implementation. In: IEEE Conference on Embedded Computer Systems: Architectures, Modeling and Simulation.

Buder, M., 2012. Dense realtime stereo matching using a memory efficient Semi-Global-Matching variant based on FPGAs. In: Kehtarnavaz, N., Carlsohn, M. (Eds.): Real-Time Image and Video Processing 2012. SPIE Proceedings Vol. 8437.

Ernst, I. and Hirschmüller, H., 2008. Mutual information based semi-global stereo matching on the gpu. In: ISVC, Vol. LNCS 5358, Part 1, Las Vegas, NV, USA, pp. 228–239.

Gruen, A. W., 1985: Adaptive least-squares correlation – a powerful image matching technique. South African Journal of Photogrammetry, Remote Sensing and Cartography, 14 (3): pp. 175-187.

Gruen, A. W., Baltsavias, E. P., 1988: Geometrically constrained multiphoto matching. Photogrammetric Engineering and Remote Sensing, 54 (5): pp. 633 – 641.

Hirschmüller, H., 2005. Accurate and Efficient Stereo Processing by Semi-Global Matching and Mutual Information. Proc. IEEE Conf. Computer Vision and Pattern Recognition, vol. 2, pp. 807-814, June 2005.

Hirschmüller, H. and Scharstein, D. 2007. Evaluation of Cost Functions for Stereo Matching. Computer Vision and Pattern Recognition. IEEE Conference on, pp. 1–8.

Hirschmüller, H., 2008. Stereo processing by semi-global matching and mutual information. IEEE TPAMI 30(2), pp. 328–341.

Hirschmüller, H. and Scharstein, D., 2009. Evaluation of stereo matching costs on images with radiometric differences. IEEE TPAMI 31(9), pp. 1582–1599.

Michael, M., Salmen, J., Stallkamp, J., Schlipsing, M. 2013. Real-time Stereo Vision: Optimizing Semi-Global Matching. 2013 IEEE Intelligent Vehicles Symposium (IV) June 23-26, 2013, Gold Coast, Australia.

Rothermel, M., Wenzel, K., Fritsch, D., Haala, N. 2012. SURE: Photogrammetric surface reconstruction from imagery. Proceedings LowCost3D Workshop 2012, 04th – 05th Decembre 2012, Berlin.

Weisensee, M., 1992. Modelle und Algorithmen für das Facetten-Stereosehen. Dissertation. Deutsche Geodätische Kommission bei der bayrischen Akademie der Wissenschaften, Reihe C, Heft Nr. 374.

Wendt, A., 2008. Objektraumbasierte simultane multisensorale Orientierung. Dissertation. Deutsche Geodätische Kommission bei der bayrischen Akademie der Wissenschaften, Reihe C, Heft Nr. 613, Munic.

Wenzel, K., Rothermel, M., Fritsch, D., Haala, N. 2013. Image acquisition and model selection for multi-view stereo. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XL-5/W1, pp. 251-258.

Wrobel, B., Digitale Bildzuordnung durch Facetten mit Hilfe von Objektraummodellen. In: BuL 55 (1987), Nb. 3, pp. 93-101.

Zabih, R. and Woodfill, J. 1994. Non-parametric local transforms for computing visual correspondance. In *Proc. ECCV*, pages 151–158.