A Segment and Fusion-Based Stereo Approach

Frank Pagel
Fraunhofer Institute for Information and Data Processing
Autonomous Systems and Machine Vision
76131 Karlsruhe, Germany
frank.pagel@iitb.fraunhofer.de

Abstract

Most algorithms in stereo vision work on rectified images and therefore find the point correspondences row by row. So especially for standard block-matching algorithms periodic patterns are a problem in determining corresponding features reliably. This contribution describes a segment-based approach that allows the detection and removal of single outliers in an arbitrary dense disparity map and so improves the data quality. The first step is a matching of vertical edge segments in the images in a coarse to fine strategy. Then the segment information is taken into account. Even more, when using segments there is only need to calculate feature correspondences for a fraction of the image rows, which considerably reduces computation time. By fusing this information with the disparity map of the standard block matching algorithm a significant improvement of the resulting disparity map in the presence of periodic patterns can be reached.

1. Introduction

Stereo vision has been an active field of research for more than three decades. To date, stereo systems are commercially available as well as open-source implementations ([1], [2]). However, recent investigations ([3]) still revealed room for data quality improvement, namely in the presence of periodic patterns. Such structures in the scene can cause correspondence mismatchings. This happens quite often in real world and especially urban scenarios and may be a big problem for algorithms that work with the resulting 3D data.
The goal of the proposed algorithm is to increase the robustness of cheap and fast algorithms in the presence of periodic patterns. In this context robustness means the reduction of wrong disparity values in the resulting dense disparity map. Therefore a fusion-based strategy is used. A dense disparity map is calculated that is independent of the dense map of the block-matching algorithm. A feature-based optimization is performed to find correspondences of vertical edge elements in an image pyramid. By considering the whole edge segments outliers or unconfident disparities can be rejected. At a final step a fusion between the two disparity maps is done. The algorithm is designed to run in real-time.

The next section gives an overview of state-of-the-art stereo processing techniques and how they have been considered in this work. Section 3 presents a segment-based algorithm that addresses the problem of periodic structures. The vertical edge segment extraction is sketched, followed by the optimal matching approach. The disparity values between the segments are interpolated. Then a fusion step is performed where the initial dense disparity map is fused with the calculated sparse but robust disparity map. In section 4 some results on real data are shown.

2 Related Work

Without loss of generality one can assume that the stereo image pair is rectified ([4], [5]). This means that corresponding points are in the same image line, the so-called epipolar line, and hence the correspondence search can be reduced to a 1D search. Features on an epipolar line respective a row in the left image need to be matched to features in the same row in the right image.
In this paper the benefits of several stereo techniques are considered, such as dense or pixel-wise ([6], [7], [8]), sparse ([9], [10], [11]), local ([12], [13], [14], [11]) and global techniques ([7], [15], [8]).
[14] implements a block matching algorithm. Block matching is also the most popular algorithm used in commercial stereo systems because of its simplicity (see also [1]). Because the disparities are searched sequentially for each pixel or feature, these algorithms are highly error-prone to periodic patterns so that some disparity post processing has to be implemented.

978-0-7695-3651-4/09 $25.00 © 2009 IEEE
DOI 10.1109/CRV.2009.11

IEEE computer society

[11] calculates sparse depth maps using a pyramidal approach and vertical edges that is similar to the segment-based approach in this paper. Mismatches made in an early stage of the pyramid connot be removed during the processing procedure. However, [11] only generates sparse depth maps and does not explicitly consider the information along the whole segments, which is used in this paper to refine the results.
[16] is an approach especially designed for parallel processing architectures and multi camera stereo applications. It is feature based and creates clouds of 3D points. The topic of periodic structures has not been discussed so far for this system. As it is based on the backprojection of image features into planes one can only assume that periodic patterns may lead to ambiguities that need to be resolved.
[7] calculates a minimal cost path through a matrix where each entry relates to a distance measure between a pixel in the left image and a pixel in the right image via dynamic programming. [8] minimizes an energy function to find the globally optimal assignment for each pixel. Both dynamic programming and energy minimization result in dense depth maps and they consider the information of each pixel on the epipolar line. Although both algorithms should be able to handle periodic structures both algorithms are not considerable in real-time applications on a standard PC because of the high computational cost.
A probabilistic approach is given in [17], but as it has to estimate parameters in an expensive procedure it is also not executable under real-time conditions.
[15] used an optimization strategy that comes close to the proposed algorithm, except in this paper features are used instead of all pixels within an epipolar line to decrease computation time. The Hungarian Algorithm is used to find the minimal path through a cost matrix. As a result each pixel in the left row is assigned to a pixel in the right row so that the overall distance between the pixel assignments are minimal. Referred to the authors periodic structures can be well handled. Hence the information of the whole row is considered.
The approach of [18] seems to be quite promising, but no evaluation in the specific handling of periodic pattern has been done so far.

3 Robust Stereo Processing

For the following sections the images are assumed to be rectified so that the epipolar lines are colinear and corresponding image points are in the same image row ([4], [5]). The initial depth map that is used in this paper is the result of the edge-based block matching implementation of Konolige ([2], [1]), but any other depth map can be used, too, for the algorithm presented in this paper. The advantage of the Konolige algorithm is its low computational cost on a standard PC.
The algorithm proceeds as follows: First, edge segments are extracted and correspondences of edge elements are found on every rth row. Then the disparities for the whole edge segments are interpolated. The result is a sparse disparities map of edge segments that can be used to detect and remove outliers in the initial map. Finally, a fusion step is performed that takes the result of the inital, dense disparity map and the segments' information into account to yield a more robust disparity map (see fig. 1).

3.1 Correspondence Search

As the proposed algorithm should be able to handle VGA (640×480 Pixels) images in real-time, this approach focuses on strong image features. In this paper these are vertical edge segments with a large gradient so that correspondences can be found reliably. Regions between such features are considered as homogeneous or at least as regions with a slowly changing contrast. Depth values in these areas can only be estimated depending on the relative adjustment of strong features. Therefore the x-gradient of the images is calculated. A pixel $\mathbf{x}$ is assumed to be an edge pixel if and only if the gradient $g(\mathbf{x})$ is a local maxima or minima. To get the respective segment the edge pixels are concatenated with the three pixels in the row above and below. If there is more than one possible edge element to be connected to the segment, a new segment will be initialized. To reduce the influence of noise for the further processing segments with less than ϵ_S pixels are rejected.
The task of the correspondece search is to find an optimal matching of all features in the same epipolar line. Along an epipolar line a feature is given at the intersection of th epipolar line with an edge segment that passed the vertical connectivity test. A cost matrix is filled by calculating distance measures for all possible pairs of correspondences (see fig. 2). An optimal path through the cost matrix C is a minimum cost path and hence an optimal matching.
The Hungarian Method finds an optimal path in a cost matrix. This technique is equivalent to finding the maximum weighted matching on a bipartite graph([19], [20]). The algorithm has polynomial running time. The proposed approach is motivated by [15]. But instead of matching every pixel just single features are matched. The cost matrix C is filled with the results of a cost function $\rho(\mathbf{x}_L^i, \mathbf{x}_R^j)$ of the two sets of features $\mathbf{x}_L^i$ and $\mathbf{x}_R^j$ on the left and right epipolar line respectively with $i = 1, ..., N$ and $j = 1, ..., M$ and $C \in \mathbb{R}^{max(N,M)\times max(N,M)}$. The remaining matrix elements are set to ∞. The normalized correlation coefficient (NCC) between two image points is used which is defined as

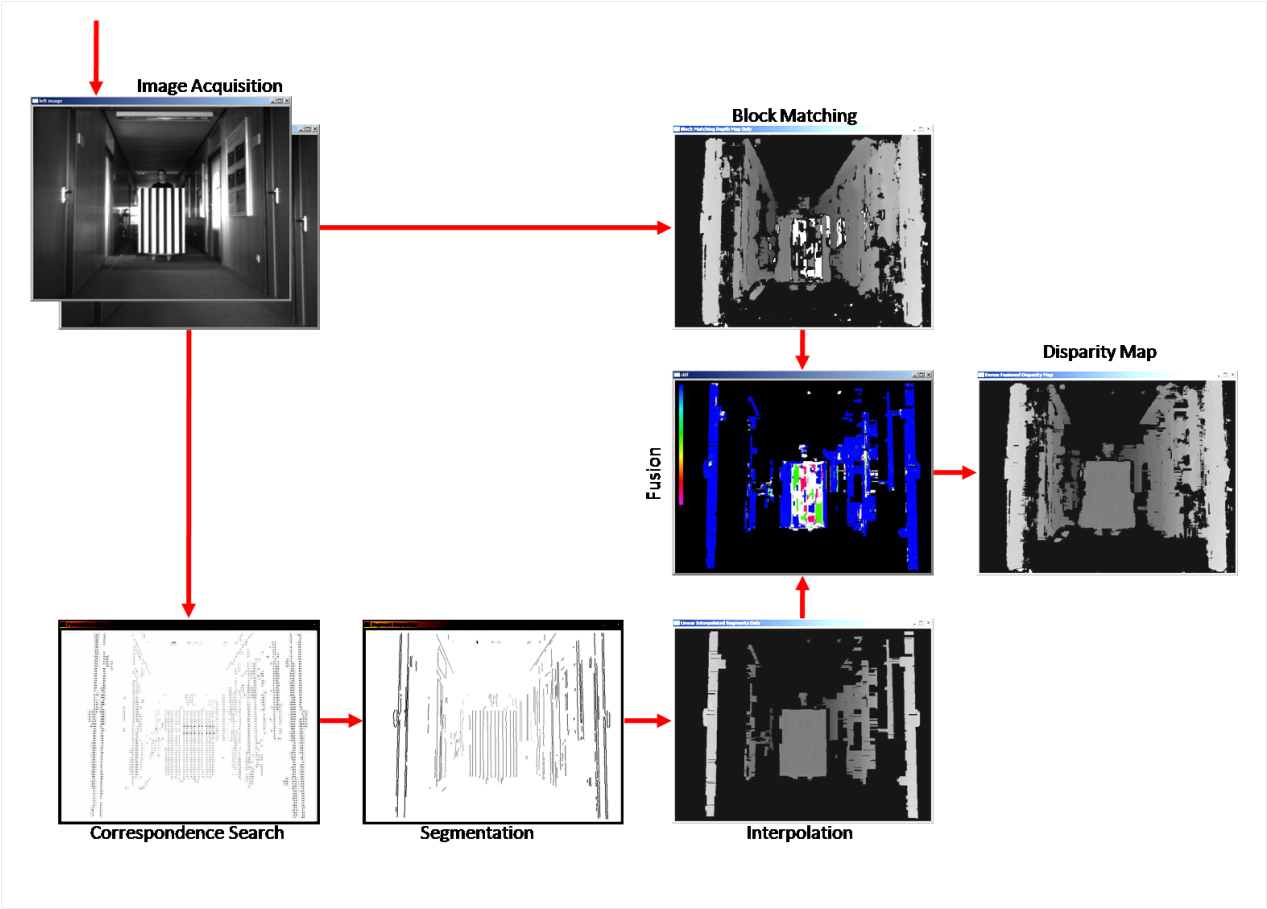

Figure 1. Fusion of the initial disparity map with the segment-based map.

$$\text{NCC}(\mathbf{x}_L, \mathbf{x}_R) = \frac{\sum_{\mathbf{i}\in\mathcal{U}}(I_L(\mathbf{x}_L+\mathbf{i})-\overline{I_L})(I_R(\mathbf{x}_R+\mathbf{i})-\overline{I_R})}{\sqrt{\sum_{\mathbf{i}\in\mathcal{U}}(I_L(\mathbf{x}_L+\mathbf{i})-\overline{I_L})^2\sum_{\mathbf{i}\in\mathcal{U}}(I_R(\mathbf{x}_R+\mathbf{i})-\overline{I_R})^2}} \quad (1)$$

where $\overline{I_L}, \overline{I_R}$ are the mean intensity values in $\mathcal{U}$ in the left and right images I_L and I_R. $\mathcal{U}$ is a $m \times m$ correlation mask centered at $\mathbf{x}_L$ respective $\mathbf{x}_R$. These image points are edge elements along the epipolar line that belong to a vertical edge segment. Experiments yielded good results with the NCC. But any other reasonable distance measure could be considered, too.

To avoid mismatches some restrictions are made. First of all the assumption is made that $x_L > x_R$. Furthermore, only disparities smaller than a given threshold are considered such that $x_L - x_R < \epsilon_{disparity}$. The costs for all pairs of points that violate these two conditions are set to infinity. The Hungarian Algorithm has no implicit ordering constraint e. g. like dynamic programming. It may happen in practice that correspondences are crossing, this means for two corresondence pairs $\{x_{1_L}, x_{1_R}\}, \{x_{2_L}, x_{2_R}\}$ that ($x_{1_L} < x_{2_L}$ and $x_{1_R} > x_{2_R}$) or ($x_{1_L} > x_{2_L}$ and $x_{1_R} < x_{2_R}$). But earlier experiments have shown that these crossings are very seldom and therefore can be neglected in this approach. Diagonal paths in the cost matrix are enforced by weighting adjacent matrix elements stronger depending on their diagonal, previous cost entry. Then the final matching is checked again and remaining crossings are solely removed.

One popular strategy in robust stereo processing is the usage of image pyramids by subsampling the original images. An edge pyramid is used where the edge elements are propagated through the pyramid levels, similar to [11]. This step is important as the structure of periodic patterns is smoothed by subsampling the image. This helps to find correct correspondence matchings in a coarse level without having the difficulties one may usually have in the original images. The correspondences found in the coarse level are used as a boundary in the next level which means that the calculation of the minimal cost path has only to be executed for

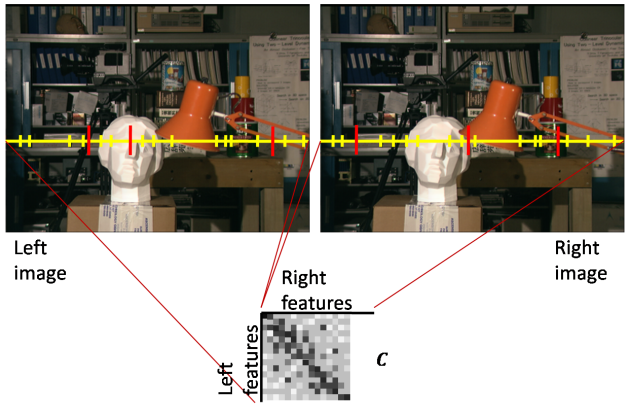

Figure 2. The Hungarian Method finds a minimal, bipartite cost matching in a cost matrix C. C is filled with distance measures between the features found on corresponding epipolar lines. The long marked features are features with no corresponding feature on the other epipolar line.

a small cost matrix. So, this approach trusts very strongly early matches in the pyramid. However, the next section shows how such mismatchings can be revealed using the segments' information.

3.2 Improving Robustness using Segments

For each vertical edge segment a disparity histogram is calculated where the disparity values $d \in \mathbb{N}$. Therefore the rows and their disparity values are counted along a single segment. To evaluate the histogram some parameters are needed: The most frequent disparity d_f and the corresponding number of rows q_{d_f} that contain this disparity, the total number of rows Q_{total} and the number of different disparity values $N_{different}$ on the segment. The distribution of the disparity values along the segment is assumed to be smooth. Next, the interval $I(d_f)$ on the histogram is defined. $I(d_f)$ contains d_f and its adjacent disparity values d_i with $d_i \neq 0$. Furthermore, the total number of disparity values inside $I(d_f)$, Q_I, the total number of disparity values outside $I(d_f)$, $\overline{Q_I}$, the number of different disparity values inside $I(d_f)$, N_I, and the number of different disparity values outside $I(d_f)$, $\overline{N_I}$ is defined. Using these parameters one can formulate the ratio

$$R_I = \frac{\overline{Q_I} \cdot N_I}{Q_I \cdot \overline{N_I}}. \tag{2}$$

The closer R_I is to zero the more unambigous is the disparity of a segment. Segments for which R_I is too big are neglected. Fig. 3 shows an example of how to calculate the values. If R_I is big enough all disparities on

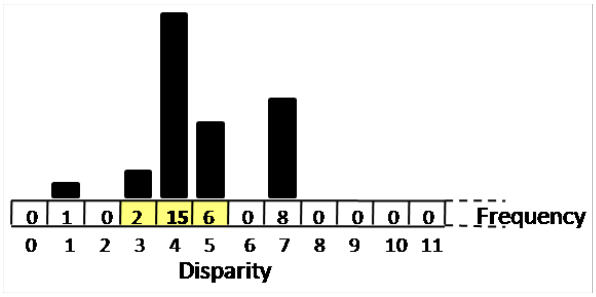

Figure 3. Example disparity histogram of a single segment. $d_f = 4$, $q_{d_f} = 15$, $I(d_f) = [3, 5]$, $Q_{total} = 32$, $Q_I = 23$, $\overline{Q_I} = 9$, $N_{different} = 5$, $N_I = 3$, $\overline{N_I} = 2$.

the histogram that are outside I are deleted on the segment. Then a linear interpolation between the remaining disparity values on the segment is performed. So, single outliers along a segment can activily be replaced. It becomes clearer now why there is no need to calculate the correspondences for every image line: By using the histogram, the local disparity distribution is approximated and can be used to make a reliable estimation of the disparities for the whole segment. So, obviously, when taking only every rth line, processing time decreases approximately by a factor of r. Fig. 4 shows the resulting segments from a sparse set of feature correspondences. Finally the correspondences in each line are refined to subpixel accuracy by fitting a quadratic curve to the image gradients:

$$\mathbf{x}^*(\mathbf{x}_0) = \frac{1}{2} \cdot \frac{g_{-1} - g_1}{g_{-1} - 2g_0 + g_1}. \tag{3}$$

3.3 Fusion of the two Depth Maps

First of all the disparity map is filled with the values from the initially calculated, dense map D_1. This map is the result of the standard block-matching algorithm. So, the task is to detect the uncertain parts of the map and replace it with either more senseful values or no values at all, to get the final disparity map D. For this purpose a second dense depth map D_2 is created by simply interpolating horizontally and linearly between two adjacent edge segments if the x-distance $\|\Delta x\| < \epsilon_{\Delta x}$ and the difference of the disparity values $\|\Delta d\| < \epsilon_{\Delta d}$. A pixel $\mathbf{x}$ of the disparity map is meant to be *uncertain*, if

$$\|D_1(\mathbf{x}) - D_2(\mathbf{x})\| > 1. \tag{4}$$

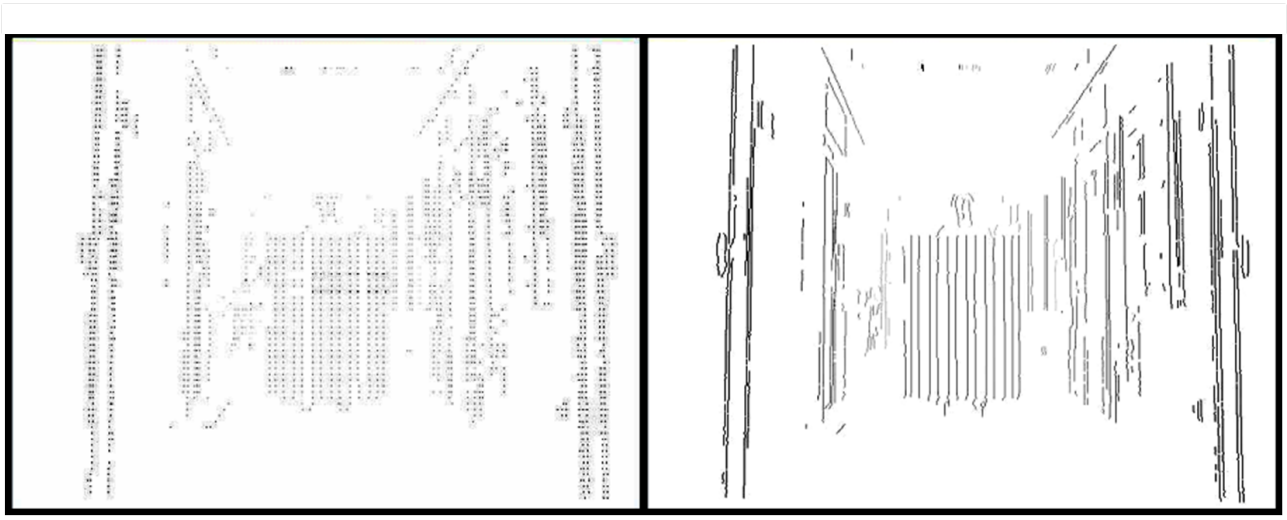

Figure 4. Correspondence images for a periodic pattern. Left: Found disparities for every 5th row. Right image: The resulting segments after histogram analysis. Notice that wrong correspondences in the left image (dark points on the periodic pattern) can be caught in the segment representation.

So, the pixel values of the final disparity map are filled with

$$D(\mathbf{x}) = \begin{cases} D_1(\mathbf{x}), & \mathbf{x} \text{ is } certain \\ 0, & \Delta d > \epsilon_{\Delta d} \\ D_2(\mathbf{x}), & \Delta d \leq \epsilon_{\Delta d} \quad \text{and} \quad \Delta x < \epsilon_{\Delta x} \\ 0, & else \end{cases} \tag{5}$$

Fig. 5 shows an outline of the proposed algorithm.

4 Experimental Results

The proposed algorithm ran on an Intel 2GHz machine. A VGA image pair was computed in $\sim 1.5s - 2s$ without any specific software optimization. The benefit of this approach becomes clear in the presence of periodic pattern. So the behaviour of both algorithms, block matching and the presented segment-based algorithm, was tested with real images of periodic patterns. The Middlebury Stereo Vision Data Base [21] was also used to give some objective results (see fig. 7).

Although the Tsukuba image may seem less intuitively after the fusion, the data itself is less noisy (but also sparser). Whereas the limbs in the Tree image look finer in the fusioned disparity map. The biggest improvement could be reached with periodic pattern. As the block matching obviously fails within such regions the fusion maps show much more reliable results. This is quite important for outdoor applications as especially in urban environments periodic pattern just like window frontages or tree rows may occur very often. Fig. 7 shows four examples of images and the processing stages of the algorithm. In fig. 8 the block matching-based input disparity map is compared with the fusion result from the presented algorithm. It can be seen that a lot of noisy disparity estimations could be either removed or corrected. Even if not every wrong estimated disparity could be removed, one can see a significant improvement.

Fig. 6 shows clearly the benefit that can be reached by fusing the block matching disparity map with the presented segment interpolated map. The upper histogramm shows systematically big peaks. This is an expected behaviour of the algorithm because of the periodic confusion (see [3]). So in this area it is difficult to determine the real distance of the periodic pattern in the scene. The histogram of the fusioned depth map has only one significant peak around 17 and one more small peak at 48 as a result of a correspondence mismatching.

5 Conclusion

An approach was presented to reduce the number of erorneous disparities caused by periodic patterns in the stereo images. Therefore a combinatorial algorithm was chosen to find a minimum cost matching of all features along two coresponding epipolar lines. The segments were analized to remove remaining outliers and to yield a confident but not necessarily dense disparity map. By fusing this disparity map with a map resulting from a standard block matching algorithm, a meaningful improvement of the data quality could be shown especially for vertical, periodic structures.

1. Get undistorted, rectified images
2. Calculate initial, cheap disparity map
3. Build image pyramid
4. Calculate gradient image and extract edges at local maxima
5. Find vertical edge segments by concatenating the edge elements
6. Propagate the edge elements through the pyramid
7. for level $l = L - 1...0$, for every $r(l)$th Line:
   - Fill cost matrix
   - Find minimal path using the Hungarian Method
   - Remove crossings
   - Use correspondences as search borders for the next level
8. Calculate and analyze disparity histogram for every edge segment
9. Interpolate along segments and remove outliers
10. Calculate subpixel disparities for each correspondence using the gradient image
11. Fusion of the interpolated segment disparity map and the initial disparity map

Figure 5. Algorithm outline.

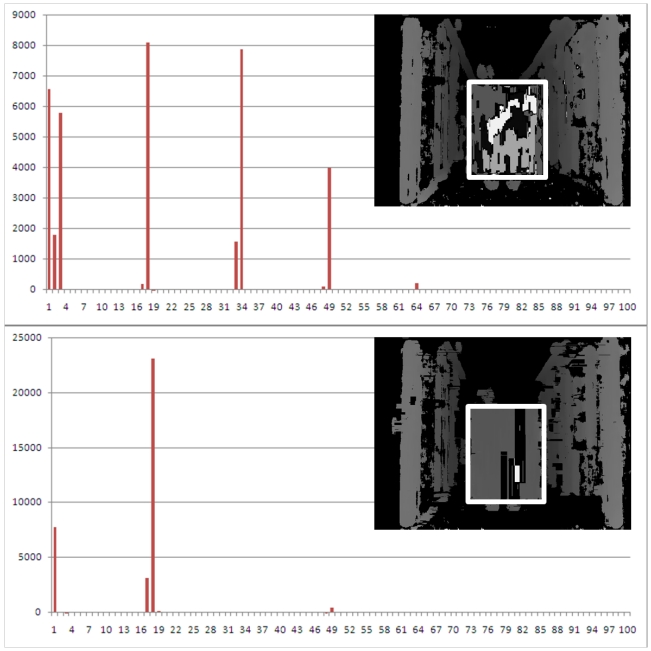

Figure 6. Disparity histograms within the selected region with the periodic pattern (disparities on the abscissa). Top: Block Matching algorithm. Bottom: Fusion result.

The approach works as an add-on for precalculated dense depth maps and runs on standard PC hardware with reasonable computational time. More test series have to be done with periodic pattern to verify the robustness in a large set of scenarios. Therefore a detailed evaluation scheme like in [3] should be used to make not only qualitive but also quantitive statements. There are plans to implement this algorithm in the near future on a hardware and software optimized platform to yield a significant computational speedup.

Acknowledgments

The author would like to thank the Federal Office of Defense Technology and Procurement (BWB).

References

[1] [Online]. Available: http://www.ai.sri.com/∼ konolige/svs/svm.htm

[2] [Online]. Available: http://www.intel.com/ technology/computing/opencv/

[3] F. Pagel, B. Elias, B. Giesler, and D. Willersinn, "Analysis of binocular stereo systems," in *Proc. of the 5th International Workshop on Intelligent Transportation*, 2008.

[4] M. Pollefeys, R. Koch, and L. Van Gool, "A simple and efficient rectification method for general motion," in *Proc. of the 7th IEEE International Conference on Computer Vision*, 1999, pp. 496–501.

[5] A. Fusiello, E. Trucco, and A. Verri, "Rectification with unconstrained stereo geometry," in *British Machine Vision Conference*. BMVA Press, 1997, pp. 400–409.

[6] D. Scharstein, R. Szeliski, and R. Zabih, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, pp. 7–42, 2002.

[7] S. Birchfield and C. Tomasi, "Depth discontinuities by pixel-to-pixel stereo," *International Journal of Computer Vision*, vol. 35, pp. 1073–1080, 1999.

[8] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate eenergy minimization via graph cuts," Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, Tech. Rep., 2003.

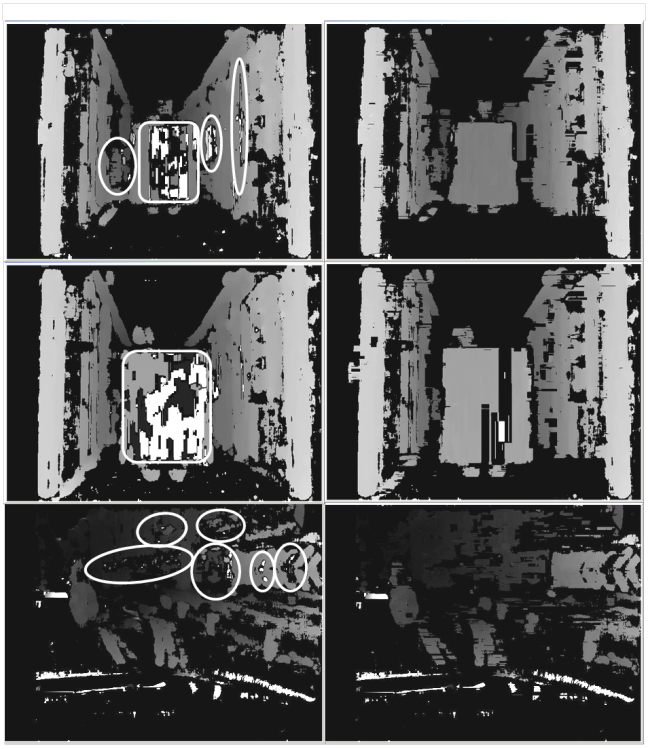

Figure 8. Left: The initial depth map calculated with a state-of-the-art block matching algorithm. Right: The fusioned result. Top: 14cm periodic pattern at 4m distance. Middle: 8cm periodic pattern at 3.5m distance. Bottom: Urban scene. A camera with focal length f=510 pixels and 9cm baseline was used.

[9] R. T. Collins and R. T. Collins, "Daca76-92-c-0041. a space-sweep approach to true multi-image matching," 1995.

[10] G. Van Meerbergen, M. Vergauwen, M. Pollefeys, and L. Van Gool, "Gool. a hierarchical symmetric stereo algorithm using dynamic programming," *International Journal of Computer Vision*, vol. 47, pp. 275–285, 2002.

[11] U. Franke and A. Joos, "Real-time stereo vision for urban traffic scene understanding," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2000.

[12] L. Falkenhagen, "Hierarchical block-based disparity rstimation considering neighbourhood constraints," in *Proc. of the International Workshop on SNHC and 3D Imaging*, 1997, pp. 115–122.

[13] A. Fusiello and V. Roberto, "Efficient stereo with multiple windowing," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society Press, 1997, pp. 858–863.

[14] T. Dang, C. Hoffmann, and C. Stiller, "Fusing optical flow and stereodisparity for object tracking," in *Proc. of the 5th IEEE International Conference on Intelligent Transportation Systems*, 2002, pp. 112–117.

[15] G. Fielding and M. Kam, "Applying the hungarian method to stereo matching," in *Proc. of the IEEE Conference on Decision and Control*, 1997, pp. 549–558.

[16] R. T. Collins, "A space-sweep approach to true multi-image matching," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 1996, pp. 358–363.

[17] L. Cheng and T. Caelli, "Bayesian stereo matching," *Comput. Vis. Image Underst.*, vol. 106, no. 1, pp. 85–96, 2007.

[18] H. Hirschmller, "Accurate and efficient stereo processing by semi-global matching and mutual information," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005, pp. 807–814.

[19] E. Lawler, *Combinatorial Optimization. Networks and Matroids*. Holt, Rinehart and Winston, 1976.

[20] C. Papadimitriou and K. Steiglitz, *Combinatorial Optimization. Algorithms and Complexity*. Prentice Hall, 1982.

[21] [Online]. Available: http://cat.middlebury.edu/ stereo/data.html

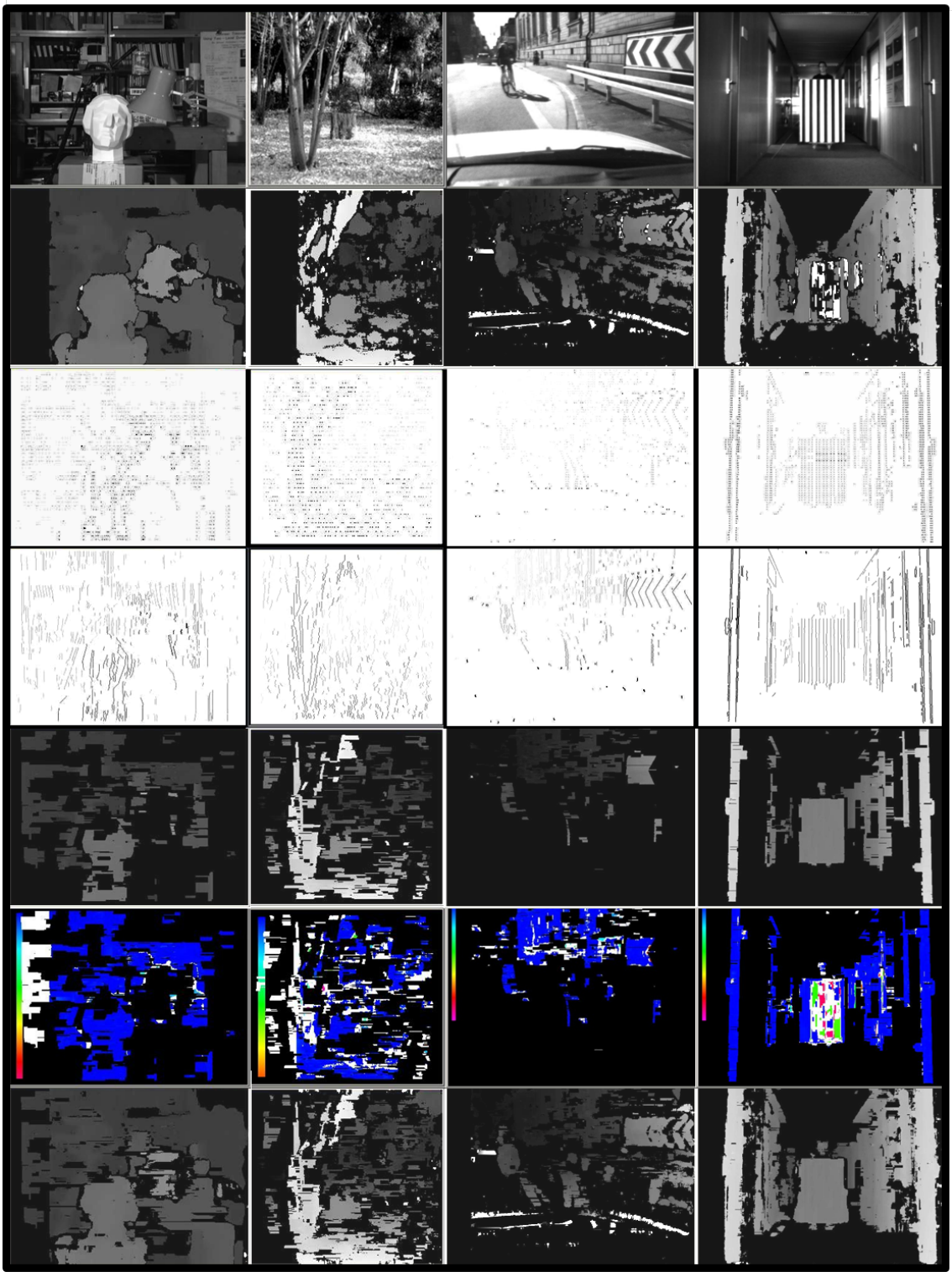

Figure 7. From left to right: Tsukuba, Tree (both from [21]), urban scene and periodic pattern . From top to bottom: Left camera images, disparity maps from [1], linewise feature disparities, segment disparities, interpolation between the edge segments, certainty maps, fusion maps.