2015 IEEE Intelligent Vehicles Symposium (IV)
June 28 - July 1, 2015. COEX, Seoul, Korea

A Multi-Block-Matching Approach for Stereo

Nils Einecke[1] and Julian Eggert[2]

Abstract— **Block-Matching stereo is commonly used in applications with low computing resources in order to get some rough depth estimates. However, research on this simple stereo estimation technique has been very scarce since the advent of energy-based methods which promise a higher quality and a larger potential for further improvement. In the domain of intelligent vehicles, especially semi-global-matching (SGM) is widely spread due to its good performance and simple implementation. Unfortunately, the big downside of SGM is its large memory footprint because it is working on the full disparity space image. In contrast to this, local block-matching stereo is much more lean. In this paper, we will introduce a novel multi-block-matching scheme which tremendously improves the result of standard block-matching stereo while preserving the low memory-footprint and the low computational complexity. We tested our new multi-block-matching scheme on the KITTI stereo benchmark as well as on the new Middlebury stereo benchmark. For the KITTI benchmark we achieve results that even surpass the results of the best SGM implementations. For the new Middlebury benchmark we get results that are only slightly worse than state-of-the-art SGM implementations.**

I. INTRODUCTION

Estimating depth information from a dual-camera is still one of the most versatile solutions for 3-D sensing with a wide application range in robotics [1], [2], [3], intelligent vehicles [4], [5], [6], [7] and also space science [8], [9]. The drawback of estimating depth with a stereo camera system, with respect to direct systems like lidar or time-of-flight, is the comparably high computational effort which is necessary to extract the depth information from the stereo images by finding correspondences. On the benefit side stereo cameras provide depth information with a very high spatial resolution which is a prerequisite for obstacle avoidance or path planning. Furthermore, the images provided by the cameras allow for other usage like object recognition or ego-motion estimation.

In the literature there are two main categories [10] of algorithms for finding the stereo correspondences: local and global methods. Local methods typically find correspondences by matching patches of one stereo image to the other image. In contrast, global methods typically optimize for an energy function that describes the best transformation of one image into the other. Usually this involves some smoothness or regularization terms in order to tackle NP-completeness for feasible processing. Apart from these two large groups there is one method that is located in between: semi-global-matching (SGM) [11]. On the one hand SGM is also based on an energy-functional, on the other hand it does not employ a fully global optimization but optimizations along one-dimensional paths. This semi-global optimization scheme has a depth accuracy that comes close to global stereo methods but with a much lower computational complexity. Due to this, SGM has become very popular, especially in the domain of intelligent vehicles [12], [7], [13]. One major drawback of SGM is its high memory footprint because it requires the full disparity space image (DSI[1]). This property makes it very challenging to bring SGM to low-energy hardware means like FPGA [5].

In contrast, local stereo methods based on block-matching are very easy to port to various hardware architecture because they need only a small part of the DSI at a time and the processing is embarrassingly parallel [14]. The downside of local methods is a generally lower accuracy and density of the resulting depth maps. In this paper, we will introduce a novel multi-block-matching (MBM) scheme that combines matching blocks of different size and shape in a probabilistic fashion. This scheme leads to a significant improvement over standard block-matching (BM) stereo while still preserving the low-memory and high-parallelization properties. Our experiments with this novel multi-block-matching (MBM) scheme on the KITTI [15] and the new Middlebury (version 3) [16] stereo benchmark show a major improvement with respect to standard BM stereo. For the KITTI data set the achieved performance surpasses even the best SGM algorithms while keeping almost the speed of standard BM stereo.

II. BLOCK-MATCHING STEREO PROBLEMS

The basic idea of standard block-matching (BM) stereo is very simple. By correlating image patches (called blocks or filters) between the left and right stereo images, correspondences between the images are found. The position difference of a correspondence is called disparity and is inversely coupled to the distance. There is a large bunch of cost functions [10] used as matching criteria; however, typically *sum of absolute difference* (SAD), *normalized cross-correlation* (NCC), *rank transform* (RT) [17] or *census transform* (CT) [17] are used. A general description of the BM cost functions $\hat{C}$ for a block-shaped neighborhood N_p of pixels around each image pixel p is given by:

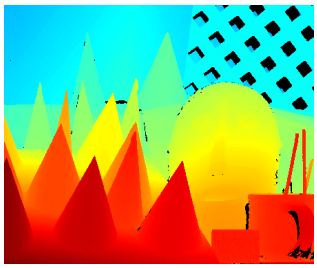
$$\hat{C}(p,d) = \sum_{q \in N_p} C(q,d), \tag{1}$$

[1]Nils Einecke is with the Honda Research Institute Europe GmbH, D-63073 Offenbach/Main, Germany nils.einecke@honda-ri.de

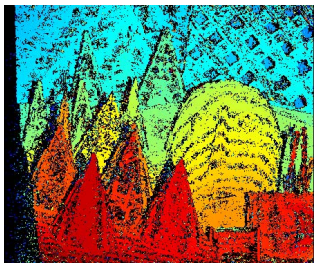
[2]Julian Eggert is with the Honda Research Institute Europe GmbH, D-63073 Offenbach/Main, Germany julian.eggert@honda-ri.de

[1]The DSI is the (virtual) three-dimensional memory structure that holds the cost values for all pixels and all disparities, i.e. its memory complexity is $\mathcal{O}(whd_n)$ where w is the image width, h is the image height and d_n is the number of disparities.

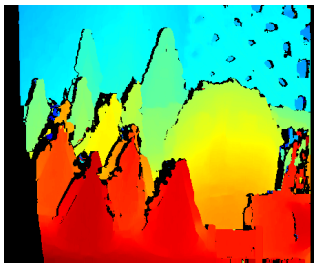
(a) cones scene



(b) ground truth depth



(c) 3x3 BM stereo



(d) 11x11 BM stereo

Fig. 1: The images in this figure demonstrate the trade-off between the fattening effect and the depth estimation noise for BM depth estimation. In (a) the *Cones* scene from the Middlebury online benchmark (version 2) is shown together with its ground truth depth map in (b). The depth maps (c) and (d) are the result of a 3x3 and a 11x11 BM stereo, respectively. While the small block-size leads to a very noisy map but sharp object contours, the larger block-size leads to a noise-free but very blurred depth map due to the fattening effect.

with the per pixel cost function C

$$C(p,d) = \Theta(I_1(p), I_2(p+d)), \tag{2}$$

where p is the pixel position in the first image I_1, d is the disparity and the second image is I_2. Θ describes the actual pixel cost metric, e.g. absolute difference of pixel intensities.

Unfortunately, the averaging of pixel costs C to get the block costs $\hat{C}$ introduces a problem known as fattening. This means that the disparity values in the disparity map get smeared. In most cases the foreground disparities are smeared over background disparities and thus this effect is sometimes called *foreground-fattening* but also known as *bleeding*. The fattening (bleeding) introduces a trade-off consideration for the size of the matching block. On the one hand, large blocks lead to stable and dense disparity maps with strong fattening. On the other hand, small blocks lead to noisy and sparse maps with very weak fattening. Fig. 1 shows an example where NCC stereo is applied to the well-known cones scene from the old Middlebury (version 2) benchmark [10] with two different block sizes. Comparing the ground truth data from Fig. 1b with the stereo results of a small 3x3 NCC block filter in Fig. 1c shows a lot of noise. To the contrary, the results using a larger 11x11 NCC block filter in Fig. 1d show no measurement noise but a map with strong fattening (smeared disparities).

The question is: Is there a way to do better? In the recent years adaptive support-weight block-matching [18] has become quite popular to tackle the fattening effect. Instead of building a straightforward averaging sum, the single pixel contributions are weighted in accordance with their (color-)similarity to the center pixel. Unfortunately, this technique is computationally very costly and good results were only achieved for colorful benchmark images. For outdoor scenes like in the KITTI benchmark no promising results with this technique have yet been published.

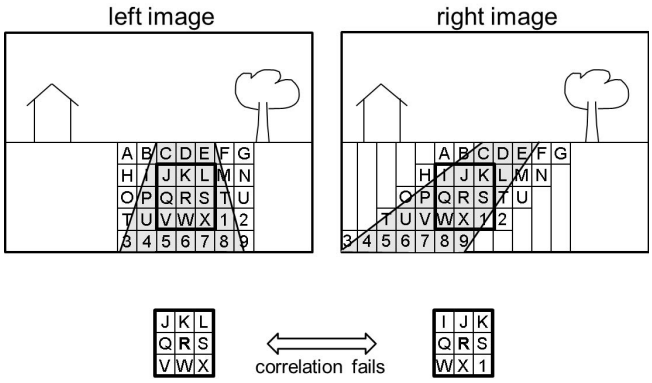


Fig. 2: This figure highlights the problem of the inherent constant disparity assumption of standard block-matching stereo. In real-world scenarios the disparity values within a matching block can vary drastically. One example are highly slanted surfaces as shown here. The non-constant disparity leads to strongly dissimilar corresponding image patches. This dissimilarity will lead to a failure of the correspondence search due to a bad correlation value.

Another way of tackling the fattening effect was presented in [19]. Here the matching block was subdivided into smaller blocks and only the best n sub-blocks were used to compute the average cost. With this best-blocks selection scheme the shape of the overall block is able to adapt to object borders at depth discontinuities. The only downside is the necessary sorting (or selection). For 9 (3x3) sub-blocks this is acceptable but for more sub-blocks the sorting might require a larger share of the overall processing time. Unfortunately, this technique has not been applied to modern benchmarks which makes it hard to judge its performance with respect to current methods.

Another problem of BM stereo is the inherent homogeneity assumption, i.e. that all pixels within one block have the same disparity. This assumption can be heavily violated in real-world scenes. In particular, intelligent vehicle scenarios are such cases. Due to the strong slant of the ground with respect to the forward-looking camera, pixels in neighboring image lines have strongly different disparities (a strongly different depth). As Fig. 2 demonstrates, this leads to low correlation values even for actually correct matches. There has been some work [20], [21] to tackle this problem with image pre-warping; however, this is only feasible for larger known scene parts like the street or house fronts.

III. MULTI-BLOCK-MATCHING STEREO

In this paper, we tackle the improvement of BM stereo from a more probabilistic point of view. Lets start with the following consideration. Assume you have a scene with slanted surfaces like the typical street scenario with houses or other upright obstacles as shown in Fig. 2. As explained above the correlation of the street will be difficult due to

the inherent homogeneity assumption. However, this is only true for squared matching blocks. Horizontally elongated matching blocks will have less problems with matching since the homogeneity assumption is not violated. The problem is that horizontally elongated matching blocks will lead to a very strong horizontal fattening and might ignore small vertical structures like poles or trees.

To overcome this problem, our main idea is to combine matching blocks of different shape and/or size in a probabilistic manner in order to get the best results of each matching block. For doing so we regard the costs of the typical BM correlation search for a pixel p as a probability distribution over the disparities d. This means that the costs of a set of different blocks B can be integrated in a multiplicative fashion:

$$\tilde{C}(p,d) = \prod_{b\in B} \frac{\hat{C}_b(q,d)}{S_b(p)} \quad (3)$$

$$\hat{C}_b(p,d) = \sum_{q\in N_p^b} C(q,d) \quad (4)$$

$$S_b(p) = \sum_{d} \hat{C}_b(p,d), \quad (5)$$

where $\hat{C}_b$ is the matching cost of block b and S_b is the sum of all costs $\hat{C}_b(p,d)$ for a pixel over all disparities. The summed cost S_b is a normalization factor that turns the cost distributions into probability distributions. Unfortunately, S_b is different for each pixel and not known before the complete stereo computation is finished which would mean to either store the whole distribution or to compute stereo twice. Both options are not favorable because they either impede the high speed of BM stereo or drastically increase the otherwise low memory footprint.

Fortunately, it is not necessary to know $S_b(p)$. The typical way of extracting the final disparity D for a pixel p in BM stereo is to select the disparity with the best cost:

$$D(p) = \underset{d}{\operatorname{argmax}}\, \hat{C}(p,d) \quad (6)$$

This means that for MBM the final disparities are calculated by:

$$D(p) = \underset{d}{\operatorname{argmax}}\, \tilde{C}(p,d) \quad (7)$$

$$= \underset{d}{\operatorname{argmax}} \prod_{b\in B} \frac{\hat{C}_b(q,d)}{S_b(p)} \quad (8)$$

$$= \underset{d}{\operatorname{argmax}} \prod_{b\in B} \hat{C}_b(q,d) \quad (9)$$

Hence, the normalization of the distributions is not necessary and MBM stereo retains the favorable speed and memory properties of BM stereo.

But why should this probabilistic combination work at all? In the ideal case, the cost distribution will have a strong peak at the correct disparity where the patches of the two images are highly correlated. The costs for the other disparities can be regarded as noise since, ideally, unrelated image patches are compared. Furthermore, the cost will be reduced when

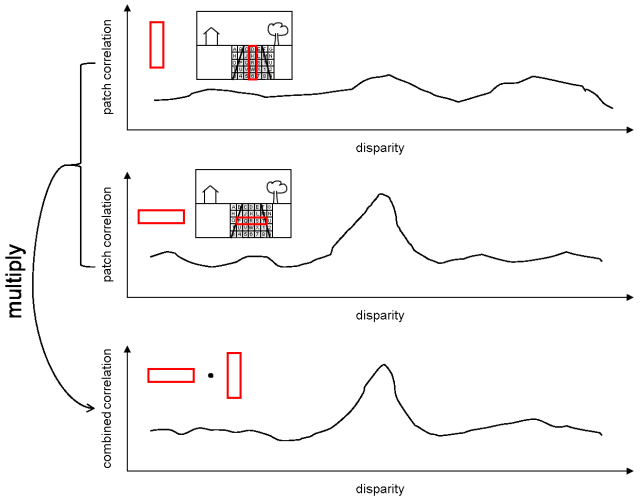


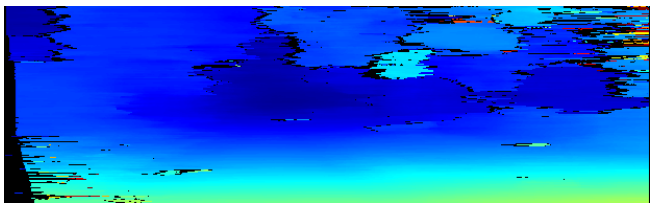
Fig. 3: This is a schematic visualization of the multi-block-matching (MBM) principle. In standard BM stereo the correlation values computed during the correspondence search can be regarded as a probability distribution for each pixel. Here the top two distributions are the correlation values for a vertical and a horizontal matching block on a slanted street surface (see also Fig. 2). While the horizontal block has a clear peak, the vertical block has very low correlation values because it encompasses pixels of different disparities. When combining both probability distributions multiplicatively the major peak of the better fitting matching block dominates the resulting cost distribution.

a block encompasses a depth border since only a partial match can be achieved (see also Fig. 2). Thus, at each pixel, the block fitting best to the scene structure will have the most prominent peak and dominate the overall cost after the probabilistic fusion. This principle is schematically outlined in Fig. 3 for the probabilistic integration of a vertical and a horizontal matching block for the street scenario discussed in Fig. 2. Since the disparity of the street changes with the image lines the vertical matching block encompasses scene structures of different depths thus leading to very low correlation values even for the correct disparity of the block's middle pixel. On the other hand, the horizontal matching block encompasses only scene parts of constant depth. Thus, it will have a very prominent peak at the correct disparity. Combining the cost distributions of both blocks will lead to a new combined cost distribution that has a high peak at the correct disparity of the middle pixel and a low match value at other disparities.

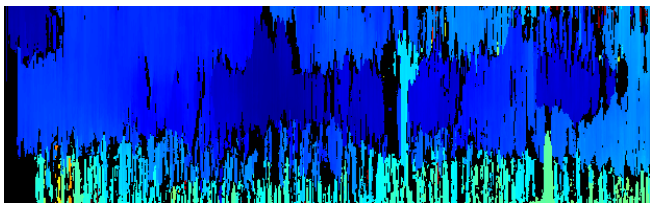
Similarly, when having a vertical structure like a pole the horizontal matching block would have a very flat distribution and the vertical matching block would have a high peak at the correct disparity. This means that the probabilistic combination of both block correlations leads to some kind of block filter steering. When the vertical block matches best the vertical block distribution will dominate the overall cost distribution and when the horizontal block matches best the horizontal block distribution will dominate the overall cost distribution.

Of course the above considerations are based on an idealization of the real world. Foremost, the world is structured and thus the cost values beyond the correct match will not

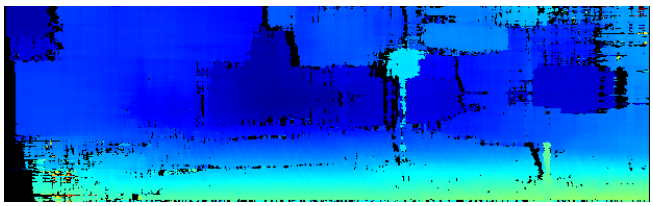
(a) KITTI training scene no. 106



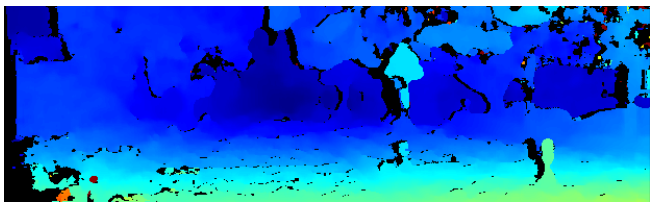
(b) horizontal block 61x1



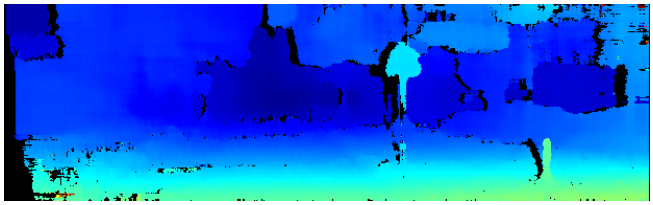
(c) vertical block 1x61



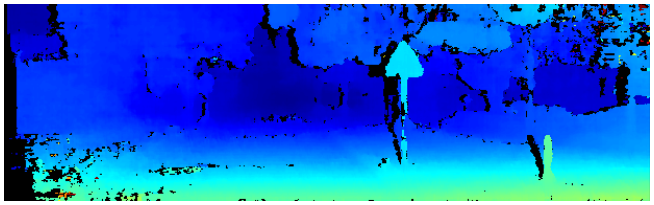
(d) horizontal•vertical block



(e) square block 11x11 (standard block-matching stereo)



(f) horizontal•vertical•square block



(g) multi-block-matching stereo

Fig. 4: In (a) the training scene no. 106 from the KITTI training test set is shown. Using a horizontal block for BM stereo (b) leads to good results for the ground but bad results for vertical structures like the poles. For a vertically shaped block (c) the opposite occurs. Combining both blocks (d) by integrating the matching cost distributions (see Fig. 3) leads to a much better result but with unfavorable streaking artifacts. Standard BM stereo with square blocks (e) does not exhibit such artifacts. A combination of the square block correlation costs with the combined horizontal and vertical block matching costs (f) leads to a significant improvement. For the proposed multi-block-matching approach (g) the matching cost of a horizontal block, a vertical block and two square blocks of different size are combined.

be random or necessarily flat. Thus, the probabilistic integrations could in principle fail. Nevertheless, our experiments in the next section show that this kind of multi-block cost combination is very effective.

Fig. 4 exemplarily shows the proposed integration for the KITTI training scene no. 106. The sub-figures 4b and 4c show the results for a thin horizontal block (61x1) and a thin vertical block (1x61), respectively. As pointed out earlier, such block shapes are very dangerous to use due to unfavorably strong fattening effects. It can easily be observed in Fig. 4b that the horizontal block indeed strongly improves the depth estimation of the ground surface, but it also completely erases the vertical ground pole on the lower right. In contrast, the vertical block (Fig. 4c) is able to capture the ground pole and the street sign pole but completely fails for most of the ground surface.

However, when combining the matching cost correlations of the two blocks, the result is a beneficial combination of the strengths of both block shapes (see Fig. 4d). Here the ground surface is estimated quite well and most parts of the two poles are also captured. Unfortunately, the overall result is not completely satisfying. When comparing the combined depth map with the results of a standard BM with a squared block, as shown in Fig. 4e, one can see that the depth map of the combined vertical and horizontal block is very noisy and exhibits some unfavorable streaking artifacts. Also the object shapes are not captured so nicely. For example the triangle street sign has a more circular shape in the combined depth map while standard BM with a squared matching block captures the triangle shape of the sign much better.

The problem is that at certain image positions neither the horizontal nor the vertical block fit the scene shape very well. Since both block shapes are quite elongated, both will violate the homogeneity assumption because they both have background information in their block field. One solution

could be to use smaller blocks. However, as the results in the next section will show this is not the best way to go. It is much better to combine the two blocks with another block, namely a squared one. The result of further combining the horizontal and the vertical block with a 11x11 squared block is depicted in Fig. 4f.

The additional 11x11 square block removes a lot of noise and most of the streaking artifacts. As the experiments latter show the best combination we currently found is to further include a small 3x3 block into the calculation because this reduces the fattening to a large extent and leads to good object contours. This multi-block-matching stereo approach of a thin horizontal block, a thin vertical block, a medium size square block and a small square block is show in Fig. 4g. It can be seen that thin poles are captured quite accurately. Also when comparing the results to the standard BM stereo (Fig. 4e) one can see that the street surface is estimated much smoother without the typical bumpy structure of standard BM stereo.

IV. EXPERIMENTAL RESULTS

For evaluating our novel multi-block-matching (MBM) stereo approach we used two online benchmarks: the KITTI stereo benchmark [15] with road scenes and the new Middlebury stereo benchmark version 3 (beta) [16] with a small set of very diverse scenes. We also tested several BM stereo cost functions: *sum of absolute difference* (SAD), *rank transform* (RT) [17], *census transform* (CT) [17] and *summed normalized cross-correlation* (SNCC) [22]. In all experiments standard block-matching post-processing steps were employed: left-right check, small region removal (<200 pixel), parabolic fitting for sub-pixel accuracy, background fill-in and two small median filters for noise removal (1x9 and 9x1).

It has to be noted here, that the costs were modified a bit in order to be usable in the proposed scheme as a probability:

$$\mathrm{SAD}_{\mathrm{mod}} = 255 - \mathrm{SAD} \quad (10)$$

$$\mathrm{RT}_{\mathrm{mod}} = 255 - \mathrm{RT} \quad (11)$$

$$\mathrm{CT}_{\mathrm{mod}} = 64 - \mathrm{CT} \quad (12)$$

$$\mathrm{SNCC}_{\mathrm{mod}} = 1 + \mathrm{SNCC} \quad (13)$$

This means that all costs need to be maximized and SNCC is bound between [0,2]. The costs were not normalized to yield 1 for a best match because as discussed in section III the typical max selection renders such a normalization unnecessary.

The first tests were conducted with the 193 KITTI training images to investigate different combinations of block sizes and shapes. Table I gives an overview of these results for the cost functions SAD, RT, CT and SNCC. In this analysis the rank transform is applied using a 11x11 neighborhood and the census transform using a 7x7 neighborhood. Please note that the neighborhoods for image transformations are independent of the block-sizes used for block-matching. Similarly, for the two-stage SNCC cost function the NCC size is kept fixed with a 3x3 size, i.e. the sizes in the table refer to the summation stage of SNCC. The results for the different cost functions are quite consistent. As already observed in the example in Fig. 4, using only the vertical and horizontal block leads to worse results than using standard BM. This is mainly due to strong streaking artifacts. A further combination of the horizontal and vertical block with a standard square block leads to substantial improvement over the combined horizontal and vertical result but is still worse than standard BM. The final combination with an additional small 3x3 square block results in a significant improvement over the standard BM approach[2].

TABLE I: This table gives a short overview of the performance of MBM stereo for different cost functions and different block combinations. Please note that these results were generated using the KITTI training data. In the first row the results for a standard block-matching are shown. The second row is MBM with just one vertical and one horizontal block. The third row shows the results for an additional square block and the fourth row shows the MBM results of a vertical, a horizontal and two differently sized square blocks. The final row shows some more experimental results where the vertical and horizontal block correlations are combined with a max operator instead of a multiplication. Currently, this final combination gives the best result for MBM.

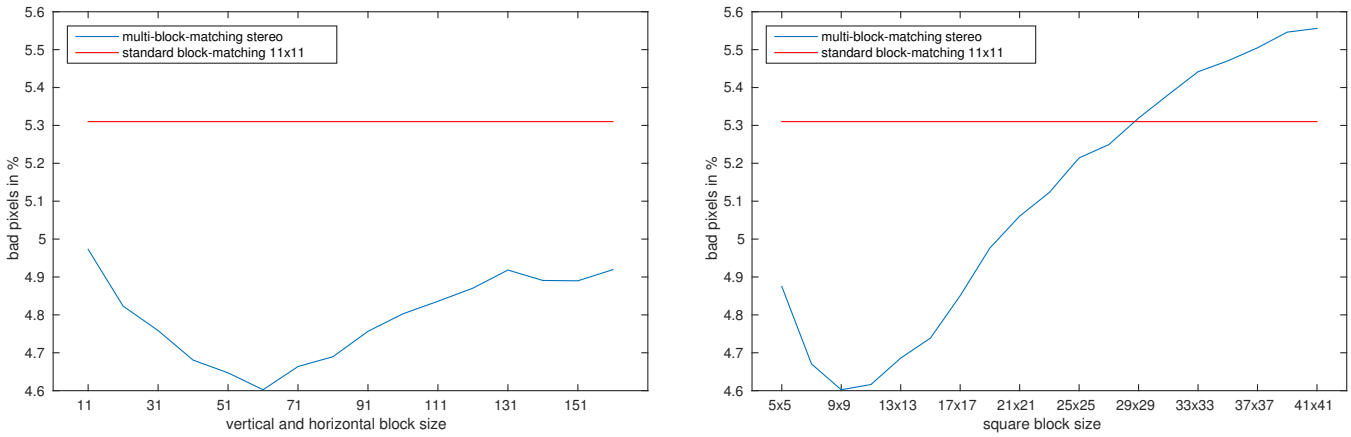
Block Combination	SAD	RT	CT	SNCC
11x11	22.39%	6.19%	5.52%	5.31%
1x61, 61x1	31.94%	9.97%	7.42%	7.60%
1x61, 61x1, 11x11	25.98%	6.35%	5.49%	5.42%
1x61, 61x1, 11x11, 3x3	24.01%	5.70%	4.99%	4.87%
max(1x61, 61x1), 11x11, 3x3	19.57%	5.47%	4.70%	4.62%

In a further exploratory analysis we used a pixel-wise max selection for the thin horizontal and vertical block correlations costs (see bottom row of Table I). This turned out to give another significant performance boost. Unfortunately, only correlation costs of blocks with the same amount of pixels are comparable. So this max selection cannot be applied in general. However, there might be some potential for further improvement which has to be investigated in future work.

In order to find out how stable the multi-block-matching results are, we varied the block-sizes in a wide range. Fig. 5 depicts results for the SNCC cost measure together with the performance of the standard BM result of SNCC with 11x11 patch size. In Fig. 5a the performance of MBM for different sizes of the horizontal and vertical block filter are plotted. These show that the optimal size for these two thin blocks is about 61, i.e. 61x1 for the horizontal block and 1x61 for the vertical block. The second plot in Fig. 5b depicts the performance of MBM for different sizes of the larger square filter, i.e. the 3x3 block was kept fix. This plot shows a clear optimum at 9x9 block size.

The best block sizes (61x1, 1x61, 9x9, 3x3) found for MBM with the KITTI training data set were then applied to the KITTI testing data set to upload the result for the KITTI online table (`http://www.cvlibs.net/datasets/kitti/eval_stereo_flow.php`). Table II gives a short overview of the comparison in the KITTI online table from December 2014. This overview shows that our

[2]For SAD there is no improvement. One reason could be that SAD typically needs different block-sizes than the normalizing cost measures.



(a) changing vertical and horizontal block size

(b) changing medium square block size

Fig. 5: These plots analyze the multi-block-matching stereo more in detail by varying the block sizes. (a) In the left plot the block sizes of the thin vertical and horizontal blocks are changed but the square block sizes are kept fixed. This plot shows a clear tendency for improvement for all tested block sizes with an optimum at 61, i.e. 61x1 for the horizontal and 1x61 for the vertical block. (b) The right plot depicts the performance for changing the size of the medium square block. The results clearly demonstrate an optimum around a size of 9x9.

TABLE II: Snapshot of the online KITTI stereo benchmark table from December 2014. It shows the top two ranking methods at that time: *CVPR 1186* [23] and *MC-CNN* [23] both being anonymous at that time; plus some selected methods: *Particle Convex Belief Propagation* PCBP [24], *SceneFlow* [25], *SGM with road scene adaption* rdSGM [23], *Weighted SGM* wSGM [26] and *summed normalized cross-correlation* SNCC [22]; together with the proposed *multi-block-matching stereo* MBM. Please note that currently (December 2014) rdSGM and wSGM are the best SGM methods in the KITTI benchmark.

Rank	Method	Out-Noc	Out-All	Avg-Noc	Avg-All	Time
1	CVPR 1186	2.47%	3.27%	0.7 px	0.9 px	265s
2	MC-CNN	2.61%	3.84%	0.8 px	1.0 px	100s
...						
12	PCBP	4.04%	5.37%	0.9 px	1.1 px	5min
13	**MBM**	**4.35%**	**5.43%**	**1.0 px**	**1.1 px**	**0.18s**
14	SceneFlow	4.36%	5.22%	0.9 px	1.1 px	150s
...						
17	rdSGM	4.91%	6.07%	1.2 px	1.3 px	10s
18	wSGM	4.97%	6.18%	1.3 px	1.6 px	6s
...						
27	SNCC	5.40%	6.44%	1.2 px	1.3 px	0.11s
...						

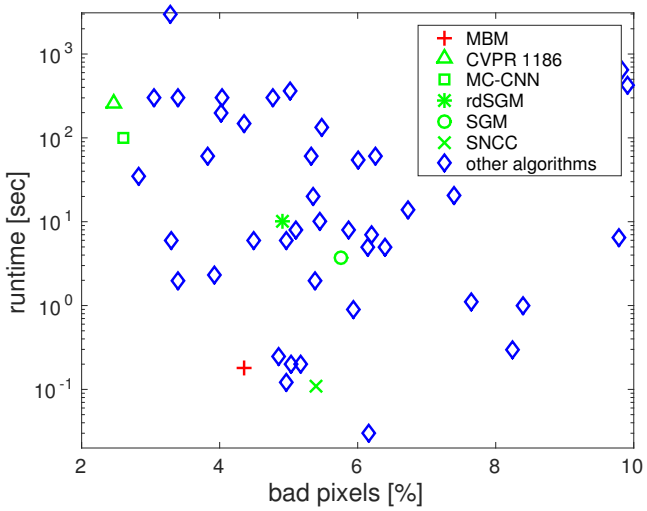


Fig. 6: This plot compares the performance of all 65 algorithms in the KITTI benchmark in December 2014 against their runtime. This shows that our proposed multi-block-matching scheme (MBM) provides a good trade-off between speed and accuracy. Please note the logarithmic scale of the runtime.

novel multi-block-matching (MBM) approach achieves high performance that even surpasses the currently best SGM implementations, rdSGM [23] and wSGM [26]. Although there are some global stereo approaches having a better performance, MBM has a much lower computational and memory complexity. To get a better impression of these results, Fig. 6 plots runtime vs. performance for all 65 algorithms in the KITTI benchmark in December 2014. Please note, that the runtime of MBM was measured on a single core of a 3.1GHz i5 processor while most of the other algorithms in KITTI employ graphic cards or multiple cores.

To see how MBM stereo works in a more diverse scene setup we applied it to the new Middlebury online benchmark version 3 [16] (`http://vision.middlebury.edu/stereo/eval3/`) which at the time of writing (December 2014) was still in a beta phase but already publicly available. Since the images in this benchmark are much larger we changed to block setup to (3x3, 21x21, 1001x1, 1x1001). The results in Table III demonstrate, that MBM is able to strongly improve the average disparity error compared to standard BM stereo. In contrast to the KITTI results, SGM performs slightly better than our proposed MBM. One reason for this different result could be that the Middlebury scenes are more diverse which could mean that SGM generalizes a bit better than MBM. Another influence could stem from the background fill-in post-processing. Since occlusion holes are much wider in the Middlebury benchmark, due to the much larger images, the potential error when filling disparities in is also larger. This has to be analyzed more in detail in future work. Nevertheless, the overall performance of MBM is very good and at least close to the performance of SGM but sometimes even superior.

TABLE III: Results and comparison using the training data of the new Middlebury online benchmark version 3 (beta). The results for the algorithms *semi-global-matching* SGM [11], *summed normalized cross-correlation* SNCC [22], and Cens5 [19] are from the online table in December 2014. The results for *multi-block-matching* MBM stereo are from a temporal upload which is not yet published in the online table.

method	resolution	avgerr	bad 0.5	bad 1.0	bad 2.0
SGM	H	4.83px	51.8%	28.2%	17.6%
MBM	**H**	**4.96px**	**54.4%**	**30.8%**	**19.8%**
SGM	F	5.82px	52.3%	31.7%	22.1%
SNCC	H	6.96px	50.8%	29.8%	20.4%
Cens5	H	7.25px	55.4%	34.2%	23.2%

A. Computational Considerations

At first sight it might seem a high effort to compute multiple block-matchings for stereo calculation. A naïve estimation of the runtime could lead to the conclusion that for four blocks you need four times the runtime of standard BM stereo. Fortunately, that is not the case because the additional runtime is only in the accumulation phase of BM stereo. Typically, in BM for each disparity the stereo images are shifted, then the pixel-wise costs are calculated and then costs are integrated via a box filter (summation filter). These box filters are very cheap to compute with a constant (size independent) runtime and very small constant memory. Instead of having one box filter in the accumulation phase, four have to be computed for MBM. This is only a minor increase in runtime for the additional three box filters and the pixel-wise multiplication or max operation of the cost distributions.

V. CONCLUSIONS

In this paper, we have presented a novel multi-block-matching (MBM) stereo approach that boosts standard BM stereo performance significantly while keeping the lean algorithmic properties. The basic idea behind MBM is to not only use one block for finding the left-right correspondences via block-matching but to use multiple blocks of different size and shape. By regarding the resulting cost distributions over the disparities as a probability distribution, the single block correlations can easily be integrated into one cost via multiplication.

We did extensive tests on the state-of-the-art stereo benchmarks KITTI and the new Middlebury version 3. Our experiments demonstrate that indeed MBM stereo leads to a large improvement when choosing the right combination of blocks with different shape and size. We found that, the combination of a thin horizontal block, a thin vertical block, a medium sized squared block and a small sized squared block results in the best performance improvement. Furthermore, we found that in some cases a pixel-wise max selection leads to better results than the pixel-wise multiplication of the costs. Altogether, we could show with our experiments that the novel MBM is able to compete with state-of-the-art stereo methods while retaining the very fast runtime of standard BM stereo.

There are several questions remaining for future work. Firstly, the performance of combined blocks is not completely understood yet. Some combinations lead to a vast improvement while others lead to much worse results. Even worse is that the performance does not necessarily increase monotonically with the amount of blocks used. Removing just one block of a combination of blocks might lead to results worse than standard BM, while adding this block to standard BM does not lead to a perceivable improvement. Secondly, the integration itself leaves some room for further investigations. From the theoretical point of having probability distributions we started with using a multiplicative integration. However, the experiments showed that sometimes a pixel-wise max selection is better. It is very likely that other integration schemes might be even more beneficial. We hope that also others investigate this new technique for block-matching stereo to help getting a better understanding of the underlying principles and to bring this method to its performance limits.

REFERENCES

[1] B. Bäuml, F. Schmidt, T. Wimbck, O. Birbach, A. Dietrich, M. Fuchs, W. Friedl, U. Frese, C. Borst, M. Grebenstein, O. Eiberger, and G. Hirzinger, "Catching flying balls and preparing coffee: Humanoid rollin'justin performs dynamic and sensitive tasks," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2011, pp. 3443–3444.

[2] N. Einecke, M. Mühlig, J. Schmüdderich, and M. Gienger, "Bring it to me - Generation of behavior-relevant scene elements for interactive robot scenarios," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 3415–3422.

[3] J. Leitner, S. Harding, M. Frank, A. Frster, and J. Schmidhuber, "Transferring spatial perception between robots operating in a shared workspace," in *IROS*, 2012, pp. 1507–1512.

[4] F. Oniga, S. Nedevschi, M. M. Meinecke, and T. B. To, "Road surface and obstacle detection based on elevation maps from dense stereo," in *Proceedings of the 10th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2007, pp. 859–865.

[5] S. K. Gehrig, F. Eberli, and T. Meyer, "A real-time low-power stereo vision engine using semi-global matching," in *Proceedings of the 7th International Conference on Computer Vision Systems (ICVS)*, 2009, pp. 134–143.

[6] M. Bertozzi, L. Bombini, A. Broggi, M. Buzzoni, E. Cardarelli, S. Cattani, P. Cerri, A. Coati, S. Debattisti, A. Falzoni, R. I. Fedriga, M. Felisa, L. Gatti, A. Giacomazzo, P. Grisleri, M. C. Laghi, L. Mazzei, P. Medici, M. Panciroli, P. P. Porta, P. Zani, and P. Versari, "VIAC: An out of ordinary experiment," in *Proceedings of the Intelligent Vehicles Symposium (IV)*, 2011, pp. 175–180.

[7] U. Franke, D. Pfeiffer, C. Rabe, C. Knoeppel, M. Enzweiler, F. Stein, and R. G. Herrtwich, "Making Bertha See," in *IEEE ICCV Workshop Computer Vision for Autounomous Vehicles*, 2013, pp. 1–10.

[8] M. Kaiser, T. Kucera, J. Davila, O. S. Cyr, M. Guhathakurta, and E. Christian, "The STEREO mission: An introduction," *Space Science Reviews*, vol. 136, pp. 5–16, 2008.

[9] J. Maki, A. Culver, R. Murdock, O. Pariser, M. Powell, N. Ruoff, and the MSL Science Team, "Mars science laboratory navcam/hazcam operations and results," in *44th Lunar and Planetary Science Conference*, 2013, Abstract #1236.

[10] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, no. 1-3, pp. 7–42, April-June 2002.

[11] 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, 2008.

[12] S. Hermann and R. Klette, "Iterative semi-global matching for robust driver assistance systems," in *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2012, pp. 465–478.

[13] T. Scharwächter, M. Schuler, and U. Franke, "Visual guard rail detection for advanced highway assistance systems," in *Proceedings of the Intelligent Vehicles Symposium (IV)*, 2014, pp. 900–905.

[14] I. Foster, *Designing and Building Parallel Programs*. Addison-Wesley, 1994.

[15] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 3354–3361, http://www.cvlibs.net/datasets/kitti/eval_stereo_flow.php.

[16] D. Scharstein and H. Hirschmüller, "Middlebury stereo evaluation - version 3," 2014, http://vision.middlebury.edu/stereo/eval3/.

[17] R. Zabih and J. Woodfill, "Non-parametric local transforms for computing visual correspondence," in *Proceedings of Third European Conference on Computer Vision (ECCV)*, 1994, pp. 151–158.

[18] K.-J. Yoon and I. S. Kweon, "Adaptive support-weight approach for correspondence search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, pp. 650–656, 2006.

[19] H. Hirschmüller, P. R. Innocent, and J. M. Garibaldi, "Real-time correlation-based stereo vision with reduced border errors," *International Journal of Computer Vision (IJCV)*, vol. 47, no. 1-3, pp. 229–246, 2002.

[20] N. Einecke and J. Eggert, "Stereo image warping for improved depth estimation of road surfaces," in *Proceedings of the Intelligent Vehicles Symposium (IV)*, 2013, pp. 189–194.

[21] B. Ranft and T. Strauß, "Modeling arbitrarily oriented slanted planes for efficient stereo vision based on block matching," in *Proceedings of the IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2014, pp. 1941–1947.

[22] N. Einecke and J. Eggert, "A two-stage correlation method for stereoscopic depth estimation," in *Proceedings of the International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, 2010, pp. 227–234.

[23] "Anonymous KITTI submission."

[24] K. Yamaguchi, T. Hazan, D. McAllester, and R. Urtasun, "Continuous markov random fields for robust stereo estimation," in *Proceedings of Third European Conference on Computer Vision (ECCV)*, 2012, pp. 45–58.

[25] C. Vogel, K. Schindler, and S. Roth, "Piecewise rigid scene flow," in *International Conference on Computer Vision (ICCV)*, 2013, pp. 1377–1384.

[26] R. Spangenberg, T. Langner, and R. Roja, "Weighted semi-global matching and center-symmetric census transform for robust driver assistance," in *International Conference on Computer Analysis of Images and Patterns (CAIP)*, 2013, pp. 34–41.