Efficient Stereo Matching for Moving Cameras and Decalibrated Rigs

Christian Unger, Eric Wahl and Slobodan Ilic

Abstract— **In vehicular applications based on motion-stereo using monocular side-looking cameras, pairs of images must usually be rectified very well, to allow the application of dense stereo methods. But also long-term installations of stereo rigs in vehicles require approaches that cope with the decalibration of the cameras. The need for such methods is further underlined by the fact that offline camera calibration is a costly and time-consuming procedure at vehicle production sites.**

In this paper we propose an approach for dense stereo matching that overcomes issues arising from inaccurately rectified images. For this, we significantly increase the search range for correspondences, but still preserve a high efficiency of the method to allow operation on platforms with highly limited processing resources.

We demonstrate the performance of our ideas quantitatively using well known stereo datasets and qualitatively using real video sequences of a motion-stereo application.

I. INTRODUCTION

Modern vehicles are often equipped with many different cameras. Famous examples include a front camera for advanced driver assistance and a rear camera for parking assistance.Lesser known examples are side looking cameras, which are usually integrated into the side mirrors or into the front bumper and help the driver at parking maneuvers or to observe crossing traffic (see Fig. 1).

The background of this paper are applications which are based on real-time motion-stereo using side-looking monocular cameras, for example [1], [2], [3]. In this case, we estimate depth by processing consecutive video frames using dense stereo methods. From these depth maps a reconstruction is computed, so that several applications can be realized, for example a parking assistant [2]. Since stereo matching is very demanding in terms of processing power, only highly efficient real-time methods are relevant.

In practice, an accurate rectification is of eminent importance when applying dense stereo methods to pairs of images. The reason for this lies in practical considerations to maximize the efficiency of stereo methods, where rectification usually transforms the epipolar geometry of both images in a way, such that epipolar lines are horizontal and *matched up*. This means, that after rectification the y-coordinate of corresponding image pixels is always constant and that the search-space for stereo-processing is heavily constrained. Therefore, an inaccurate rectification directly affects stereo matching. It is known that even slight inaccuracies of the epipolar geometry may result in significant degradation of the stereo matching performance.

In motion-stereo applications, the rectification of two consecutive camera frames must be estimated from available vehicle sensors, for example, from odometry using wheels and the levels of the dampers. However, practical experience shows that the accuracy of both odometry and damper-levels does not suffice for an accurate rectification, due to slippery or uneven ground.

Furthermore, future vehicles may be equipped with binocular front cameras, which implies the use of stereo algorithms in vehicles. For long-term installations of stereo rigs in vehicles, an adaption to decalibration issues is preferable, since there is very limited experience with vehicular stereo rigs over very long periods of time (e.g. 10 years). In these cases, the stability of the mounting concept (with respect to deterioration or deformation) and termal influences on material might have a huge impact on the accuracy of rectification. Moreover, camera calibration is costly, time-consuming and critical for the quality of serial production vehicles. From this point of view, methods are preferable that do not require an exhaustive calibration procedure, but work well with rough, approximate settings, that might, for example, be computed from CAD models.

In this paper we propose an algorithm to overcome issues arising from inaccurate rectification. We assume that the pair of images is approximately rectified and that the *epipolar deviation* of corresponding image points (i.e. distance from the epipolar line) is smaller than a predefined value. In our algorithm, we significantly increase the search range for correspondences and, although based on window-based block matching, still maintain a surprisingly high efficiency.

For this, we generalize and extend the concepts of the efficient disparity computation approach given in [4], which was originally designed for highly efficient disparity computation using accurately rectified image pairs. There, stereo matching is performed iteratively by alternating minimization and propagation phases at every pixel.

In the rest of the paper, we will first review related work, then present our method and finally show an exhaustive experimental evaluation.

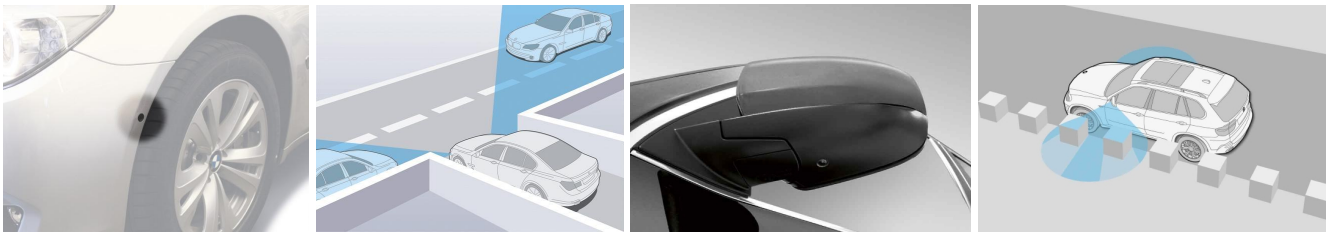
II. RELATED WORK

Binocular stereo matching is a well explored direction [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], but to our knowledge, all of these methods require an accurate rectification of the images. However, relaxing the epipolar constraint immediately leads to optical flow methods [15], [16]. While real-time GPU implementations exist, most of

This work is supported by the BMW Group.

C. Unger and E. Wahl are with the BMW Group, Munich, Germany Firstname.Lastname@bmw.de

S. Ilic is with the Technische Universität München, Garching b. München, Germany Slobodan.Ilic@in.tum.de



Camera at the front bumper Camera at the side mirror

Fig. 1. Real-Time Motion-Stereo for automotive driver assistance: camera on the vehicle observe the lateral space. If the vehicle moves, depth is inferred via motion-stereo.

the approaches that compute a dense flow field on the CPU are far from real-time. Further, many methods are designed to recover small displacements and do not directly address the problem of "small epipolar deviations". In this paper, we focus on an efficient method for standard CPUs that directly addresses the problem of large horizontal and small vertical displacements. In particular, our method is an efficient formulation using block matching and is therefore different from differential optical flow methods like [16].

While also reconstruction algorithms [17], [18], [19], [20], [21], [22], [23] rely on some knowledge about camera positions, the calibration may be done by estimating the epipolar geometry from a sparse set of feature points [24], [25], [26] using epipolar or trilinear constraints. However, the extraction of feature points, their matching and the projective warping of the pair of images for rectification is also relatively time-consuming and only works well if enough correctly matched feature points are available. In practice, these interest points are a strong limitation for our real-time motion-stereo application, since the presence of them cannot always be ensured.

In other applications, *online calibration* of stereo rigs is applied [27]. But also in these works, usually a set of sparse correspondences is required to determine or refine calibration parameters. In this work we focus on determining dense correspondences directly.

III. METHOD

We assume that the positions of the two cameras are estimated inaccurately using odometry information and that the two camera images are rectified based on these estimated positions. The unknown imprecision of the assumed camera locations results then in a distortion of the epipolar geometry of the rectified camera images. Therefore, correspondences will not lie on the estimated epipolar lines, and have to be searched within a certain corridor near the epipolar line. In the following, we call the distance from the epipolar line the *epipolar deviation*.

A. Stereo Matching with Epipolar Deviations

Our approach generalizes and extends the concepts proposed in [4].

1) Definitions: For every pixel location $\mathbf{p} = (x, y)^T$ we search for a displacement vector $\mathbf{d} = (d, v)^T$, where d is the displacement in x-direction (i.e. the disparity) and v the displacement in y-direction. For the dissimilarity of image pixels, we use a matching cost summed over a support region:

$$E(\mathbf{p}, \mathbf{d}) = \sum_{u=-w}^{w} \sum_{v=-w}^{w} C\big(\mathbf{p}+(u,v)^T, \mathbf{p}+\mathbf{d}+(u,v)^T\big) \quad (1)$$

where $C(\mathbf{p}_L, \mathbf{p}_R)$ is a cost function, for example the absolute intensity difference (AD) of the pixel $\mathbf{p}_L$ of the left image and the pixel $\mathbf{p}_R$ of the right image. All these two-dimensional displacement vectors are stored in a disparity map $\mathcal{D}(\mathbf{p})$.

2) Hierarchical Iteration: We run our algorithm in multiple resolutions, starting at a low, coarse resolution. In every resolution, we perform a minimization procedure, which computes an estimated disparity map. The use of different image scales reduces ambiguity in textureless regions and allows the recovery of large displacement vectors. Even though it is disadvantageous for thin foreground objects, the advantages outweigh the drawbacks in practice.

The image pyramid is created by halfing the image dimensions. At every resolution, the minimization uses the upscaled disparity map from the previous, lower resolution as a starting point.

$$\mathcal{D}^{(k+1)}(2x + i, 2y + j) = 2\mathcal{D}^{(k)}(x, y) \quad (2)$$

with $i, j \in \{0, 1\}$. In the very beginning, we initialize all displacements to $(0, 0)^T$.

3) Optimization Procedure: One of the central ideas of the method is that at every pixel location, a steepest descent is performed. This means that at every pixel, the displacement vector is modified using the *minimization* step. In general however, the minimization will stop at local, suboptimal minima. To alleviate this problem, a *propagation* is introduced, so that at every pixel, the displacement vectors of adjacent pixels are evaluated.

a) Minimization Step: Let the current displacement vector at $\mathbf{p}$ be $\mathbf{d}_0 = \mathcal{D}(\mathbf{p}) = (d_0, v_0)^T$. The mapping for the iteration is then given as:

$$\mathbf{d}_{n+1} = (d_{n+1}, v_{n+1})^T := \operatorname{argmin}_{\mathbf{d} \in M} E(\mathbf{p}, \mathbf{d}) \quad (3)$$

with the modified vectors

$$M := \left\{ \begin{pmatrix} d_n + i \\ v_n + j \end{pmatrix} \middle| \, i, j \in \{-1, 0, 1\}, i^2 + j^2 \leq 1 \right\} \quad (4)$$

If $\mathbf{d}_{n+1} = \mathbf{d}_n$ the iteration is stopped and the disparity map is updated. In practice, we perform the iteration at all pixels of the image.

b) Propagation Step: In the propagation at every pixel displacement vectors from surrounding pixels are evaluated and the disparity map is updated immediately:

$$\mathcal{D}(\mathbf{p}) \mapsto \operatorname{argmin}_{\mathbf{d} \in N(\mathbf{p})} E(\mathbf{p}, \mathbf{d}) \tag{5}$$

with the neighbouring displacement vectors $N(\mathbf{p})$ (with $\mathcal{D}(\mathbf{p}) \in N(\mathbf{p})$). At this step, displacement vectors may be spread through their local neighbourhood. In practice, we alternate minimization and propagation steps for a few iterations until convergence is achieved (2-3 repeatitions from experience).

B. Epipolar Geometry and Reconstruction

From the computed correspondences, the fundamental matrix can be estimated [24] to determine the epipolar geometry and a corrected rectification or a reconstruction using known techniques [17], [19].

If the pair of images is rectified, the updated disparity map for the rectified images must be derived. Let $\mathbf{p}_R = \mathbf{p}_L + \mathbf{d}$ be a correspondence and $\mathtt{H}_L$ and $\mathtt{H}_R$ be the rectifying homographies for the left and right frame. For every entry $\mathcal{D}'(\mathbf{p}'_L)$ of the updated disparity map we use the inverse mapping $\mathbf{p}_L = \mathtt{H}_L^{-1}(\mathbf{p}'_L)$ to compute:

$$\mathcal{D}'(\mathbf{p}'_L) = \mathtt{H}_R\left(\mathbf{p}_L + \mathcal{D}(\mathbf{p}_L)\right) - \mathbf{p}'_L \tag{6}$$

Please note that this formula uses inhomogenous vectors, with $\mathtt{H}_L$ and $\mathtt{H}_R$ as projective functions. In practice, this step and the rectification is relatively time-consuming and for our application it is sufficient to simply ignore the small vertical displacements.

C. Recommendations for the Implementation

To improve running times, the set of displacement vectors M can be slightly reduced:

$$M := \left\{ \begin{pmatrix} d_n \\ v_n \end{pmatrix}, \begin{pmatrix} d_n + 1 \\ v_n \end{pmatrix}, \begin{pmatrix} d_n + 1 \\ v_n + 1 \end{pmatrix}, \begin{pmatrix} d_n + 1 \\ v_n - 1 \end{pmatrix} \right\} \tag{7}$$

In this case, the horizontal component (i.e. disparity) is never decreased. Then, the number of vectors to evaluate in the propagation step can also be reduced, by storing the maximally tested disparity for every pixel: only those displacements are evaluated whose disparity is larger than the stored maximum.

Further, the quality of disparity maps can be slightly improved, if only the horizontal dimension of the images is scaled in the image pyramid. However, this strongly reduces the maximally recoverable vertical displacement.

IV. RESULTS

We compare our proposal to the method originally introduced in [4], which does not account for deviations of epipolar geometry. All in all, we show the performance of three different methods:

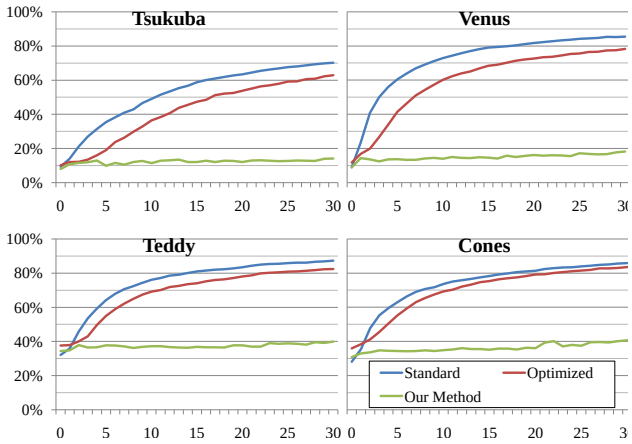


Fig. 2. The overall disparity-error (y-axis) of different stereo methods for growing values of the epipolar deviation (x-axis): the errors are percentages of disparities that differ by more than 1 from the ground truth. We tested the standard approach [4], our method (section III-A) and the optimized variant of our proposal (section III-C).

1) **Standard**: Our implementation of [4], which does not account for epipolar deviations.
2) **Our Method**: Our implementation of the concepts presented in section III-A, which can handle very large deviations.
3) **Optimized**: Our implementation including the improvements given in section III-C, which is optimized for fast running times and small epipolar deviations.

We evaluate using modified stereo datasets of the well known Middlebury benchmark [7] and show qualitative results on real world sequences acquired with a vehicle. To simulate the effects of a distorted epipolar geometry, we transform the right image of every dataset of [7] with a homography which does not modify the x-coordinate of transformed points. By keeping the left camera frame unchanged, we can still use the provided ground truth disparity maps without modification. At every value for the epipolar deviation v_{max}, we transformed the right image of every dataset with a randomly parameterized homography such that the epipolar deviation v of every pixel fulfills $-v_{max} \leq v \leq v_{max}$. To determine the overall disparity-error, we ran the algorithms and compared the estimated disparity maps to the ground truth.

A. Large Deviations

Fig. 2 shows the overall disparity-error (the percentage of disparities that differ by more than 1 from the ground truth) of the standard approach [4], our method (section III-A) and the optimized variant (section III-C) for growing values of the maximum epipolar deviation (x-axis). It can be seen that for the standard approach the error grows very quickly, whereas in our method, the error grows only slightly. The error of the optimized variant also grows fast, but an improvement at very small deviations is visible.

Fig. 4 shows disparity maps of the standard approach [4] and our method for different values of the epipolar

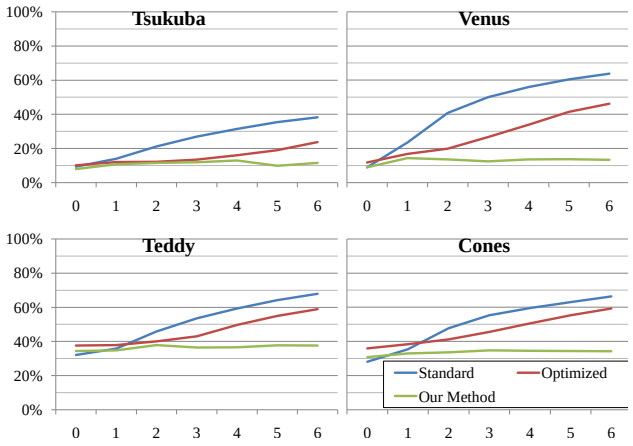


Fig. 3. The overall error (y-axis) of different stereo methods for small values of the epipolar deviation (x-axis): the errors are percentages of disparities that differ by more than 1 from the ground truth. We tested the standard approach [4], our method (section III-A) and the optimized variant of our proposal (section III-C).

deviation. The degradation of the disparity maps of the standard approach is clearly visible from the appearance of wrongly estimated disparities. In contrast, our method is also qualitatively very robust against epipolar deviations.

B. Small Deviations

For small epipolar deviations (up to a few pixels), the *optimized* variant of our method presented in section III-C is interesting. Fig. 3 shows the overall error of the methods for growing values of the maximum epipolar deviation.

The optimized variant is slightly worse than our original proposal. In practice however, it gives a good compromise between quality and processing time, if the expected maximum epipolar deviation is less than three pixels.

C. Real Sequences

We selected some particular video sequences acquired with a monocular side-looking camera on the vehicle integrated into the front-bumper for motion-stereo applications. In these examples, the pairwise rectification of camera images was not accurate due to floor unevenness. In Fig. 5, we present an undistorted camera image and disparity maps computed using the standard approach and our proposals. This figure intuitively reflects our practical experience that the optimized variant is a very good compromise between speed and accuracy, and is sufficient in almost all situations.

D. Execution Times

We tested our single-threaded implementations on a standard mobile computer (with a Intel Core2 Duo P8700 CPU with 2.53 GHz and 2 GB RAM). We used SIMD instructions to obtain maximum performance. Tab. IV-D lists the running times of our implementations and additionally the running times of traditional correlation-based stereo, which does not include the efficient disparity computation algorithm proposed in [4]. We also ran the approaches on a sequence from the vehicle. In this case, we used sub-sampled images

TABLE I

RUNNING TIMES IN MILLISECONDS OF OUR IMPLEMENTATIONS.

	Tsukuba	Teddy	Real (320)	Real (213)
Standard	31	54	19	8
Our method	192	388	134	58
Optimized	86	153	54	19
Traditional	31	141	54	17

with resolutions of **320**x240 or **213**x160, and a disparity range of 48 or 32 respectively.

The generic version (*Our method*) presented in section III-A introduces a relatively high overhead, when compared to the standard approach. However, the optimized variant comes only with a moderate computational overhead in difficult scenes. Interesting is the comparison to traditional correlation-based stereo: the additional cost is neglible in difficult scenarios. In *Tsukuba*, the maximum disparity is very small (only 10) and no advantage is taken of the efficient disparity computation.

V. CONCLUSION

In this paper, we propose a novel dense stereo matching method, whose performance is not sacrificed by inaccurately rectified images. We achieve this by significantly increasing the search range for correspondences. To preserve high efficiency, we use a generic minimization and propagation scheme embedded in a hierarchical setup. We further present an optimized variant that allows real-time operation only using CPUs.

We evaluate our proposal using famous stereo datasets and real imagery from a motion-stereo application. Our paper shows clearly that our method appeals through simplicity, good results and efficiency.

ACKNOWLEDGMENTS

We would like to thank Daniel Scharstein and Richard Szeliski for providing stereo images with ground truth data. This work is supported by the BMW Group.

REFERENCES

[1] E. Wahl, T. Strobel, A. Ruß, D. Rossberg, and R.-D. Therburg, "Realisierung eines parkassistenten basierend auf motion-stereo," in *16. Aachener Kolloquium*, 2007.

[2] E. Wahl and R.-D. Therburg, "Developing a motion-stereo parking assistant at bmw," *MATLAB Digest*, 2008.

[3] K.-T. Song and H.-Y. Chen, "Lateral driving assistance using optical flow and scene analysis," in *Proceedings of IEEE Intelligent Vehicle Symposium*, 2007, pp. 624–629.

[4] C. Unger, S. Benhimane, E. Wahl, and N. Navab, "Efficient disparity computation without maximum disparity for real-time stereo vision," in *BMVC*, 2009.

[5] O. Faugeras, B. Hotz, H. Mathieu, T. Viéville, Z. Zhang, P. Fua, E. Théron, L. Moll, G. Berry, J. Vuillemin, P. Bertin, and C. Proy, "Real time correlation-based stereo: algorithm, implementations and applications," INRIA, Tech. Rep. RR-2013, 1993.

[6] A. Brunton, C. Shu, and G. Roth, "Belief propagation on the gpu for stereo vision," in *Canadian Conference on Computer and Robot Vision*, 2006, pp. 76–81.

[7] D. Scharstein, R. Szeliski, and R. Zabih, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Comput. Vis.*, vol. 47, pp. 7–42, 2002.

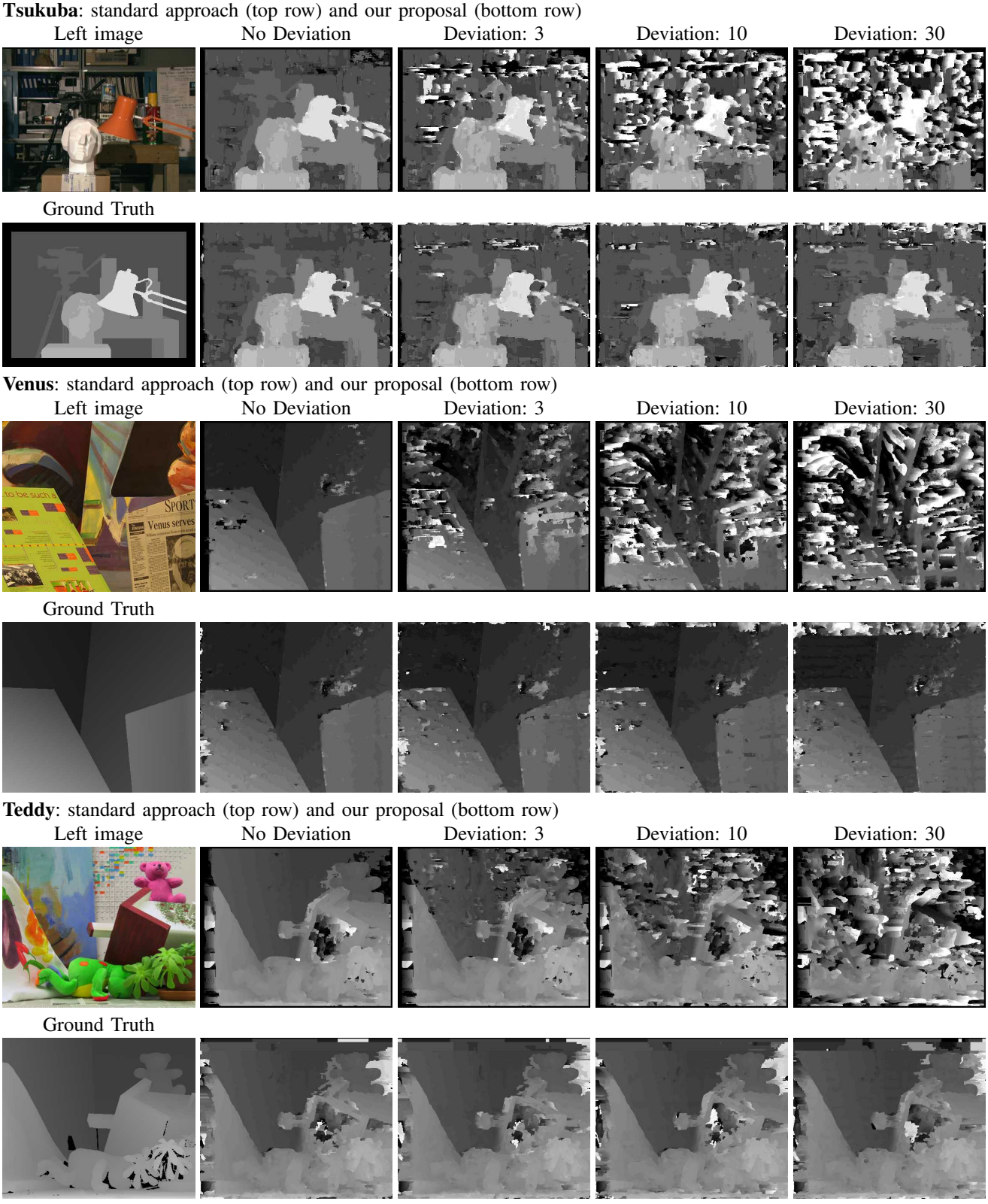


Fig. 4. Disparity Maps of [4] and our proposal at different epipolar deviations. In these cases, the epipolar deviation of at least one pixel was 0, 3, 10 or 30. In each block, the top row is the result of [4] and the bottom row shows the result of our method.

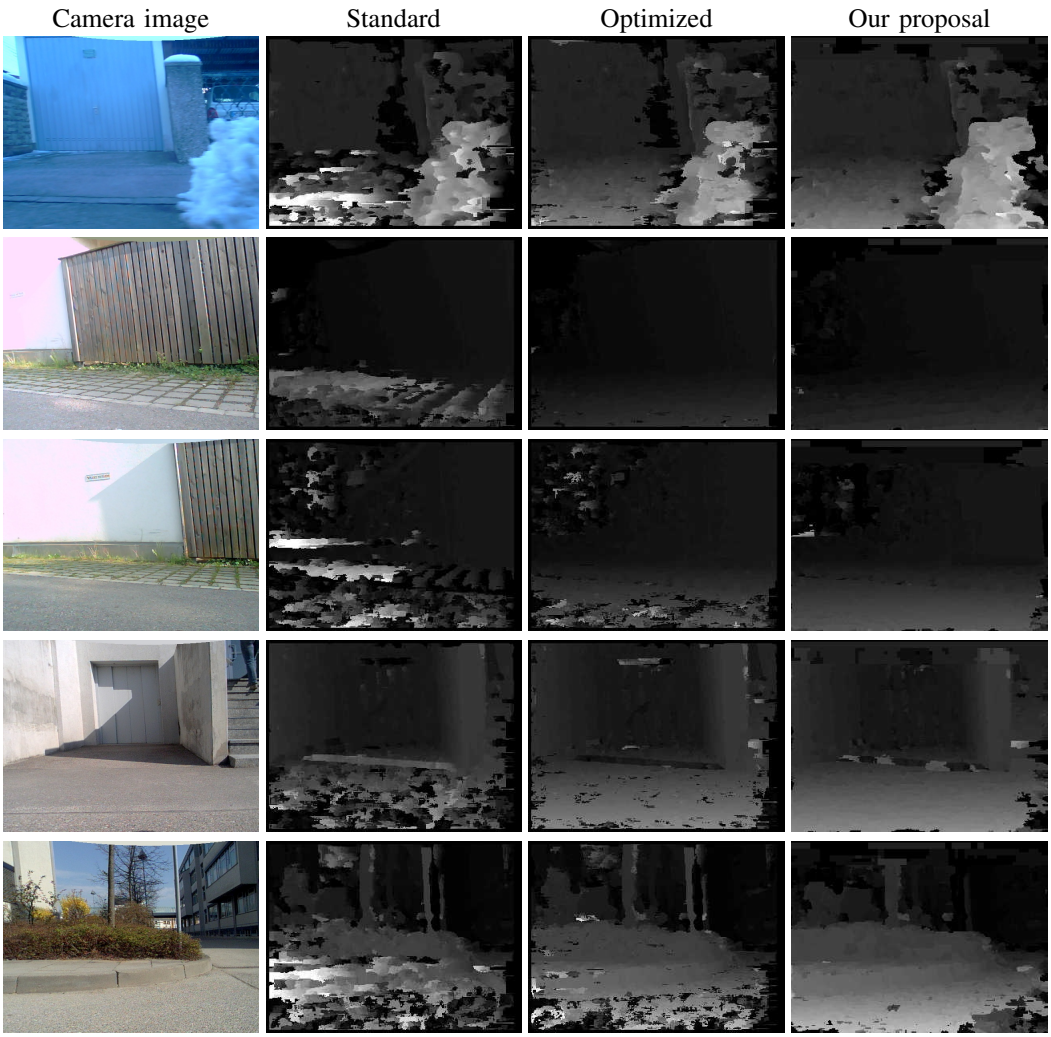


Fig. 5. Disparity Maps of [4], our optimized variant and our proposal on real sequences.

[8] H. Hirschmuller, "Accurate and efficient stereo processing by semi-global matching and mutual information," in *CVPR*, 2005, pp. 807–814.

[9] H. Hirschmüller, P. R. Innocent, and J. Garibaldi, "Real-time correlation-based stereo vision with reduced border errors," *Int. J. Comput. Vis.*, vol. 47, no. 1-3, pp. 229–246, 2002.

[10] H. Hirschmüller and D. Scharstein, "Evaluation of cost functions for stereo matching," in *CVPR*, 2007, pp. 1–8.

[11] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient belief propagation for early vision," *Int. J. Comput. Vis.*, vol. 70, no. 1, pp. 41–54, 2006.

[12] L. Wang, M. Liao, M. Gong, R. Yang, and D. Nister, "High-quality real-time stereo using adaptive cost aggregation and dynamic programming," in *3DPVT*, 2006, pp. 798–805.

[13] W. van der Mark and D. M. Gavrila, "Real-time dense stereo for intelligent vehicles," *IEEE Trans. Intell. Transport. Syst.*, vol. 7, no. 1, pp. 38–50, 2006.

[14] A. Klaus, M. Sormann, and K. Karner, "Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure," in *ICPR*, 2006, pp. 15–18.

[15] S. Baker, S. Roth, D. Scharstein, M. J. Black, J. Lewis, and R. Szeliski, "A database and evaluation methodology for optical flow," *Computer Vision, IEEE International Conference on*, vol. 0, pp. 1–8, 2007.

[16] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artif. Intell.*, vol. 17, no. 1-3, pp. 185–203, 1981.

[17] A. Azarbayejani and A. P. Pentland, "Recursive estimation of motion, structure, and focal length," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, pp. 562–575, June 1995.

[18] R. T. Collins, "A space-sweep approach to true multi-image matching," in *CVPR*, 1996, p. 358.

[19] R. Koch, M. Pollefeys, and L. V. Gool, "Robust calibration and 3d geometric modeling from large collections of uncalibrated images," in *DAGM*, 1999, pp. 413–420.

[20] P. Merrell, A. Akbarzadeh, L. Wang, J.-M. Frahm, R. Yang, and D. Nistér, "Real-time visibility-based fusion of depth maps," in *ICCV*, 2007, pp. 1–8.

[21] G. Zhang, J. Jia, T.-T. Wong, and H. Bao, "Consistent depth maps recovery from a video sequence," *PAMI*, vol. 31, no. 6, pp. 974–988, 2009.

[22] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski, "A comparison and evaluation of multi-view stereo reconstruction algorithms," in *CVPR*, 2006, pp. 519–528.

[23] C. Zach, "Fast and high quality fusion of depth maps," in *3DPVT*, 2008.

[24] P. H. S. Torr and D. W. Murray, "The development and comparison of robust methods for estimating the fundamental matrix," *Int. J. Comput. Vis.*, vol. 24, pp. 271–300, 1997.

[25] R. Hartley, "Lines and points in three views and the trifocal tensor," *IJCV*, 1997.

[26] D. Nistér, "Frame decimation for structure and motion," in *in: 3D Structure from Images-SMILE 2000, LNCS*. Springer-Verlag, 2001, pp. 17–34.

[27] T. Dang, C. Hoffmann, and C. Stiller, "Continuous stereo self-calibration by camera parameter tracking," *IEEE Transactions on Image Processing*, vol. 18, no. 7, pp. 1536–1550, 2009.