Real-time Dense Disparity Estimation based on Multi-Path Viterbi for Intelligent Vehicle Applications

Qian Long[1]
long@toyota-ti.ac.jp
Qiwei Xie[1]
qw_xie@toyota-ti.ac.jp
Seiichi Mita[1]
smita@toyota-ti.ac.jp
Hossein Tehrani[2]
hossein_tehrani@denso.co.jp
Kazuhisa Ishimaru[3]
kazuhisa_ishimaru@soken1.denso.co.jp
Chunzhao Guo[4]
czguo@mosk.tytlabs.co.jp

[1] Research Center for Smart Vehicles
Toyota Technological Institute
2-12-1 Hisakata, Tempaku, Nagoya, Aichi 468-8511, Japan

[2] DENSO CORPORATION
1-1, Showa cho, Kariya, Aichi, 448-8661 Japan

[3] NIPPON SOKEN Inc.
Nishio, Aichi, Japan

[4] Toyota Central R&D Labs., Inc.
Nagakute, Aichi, Japan

Abstract

This paper proposes a new real-time stereo matching algorithm paired with an online auto-rectification framework. The algorithm treats disparities of stereo images as hidden states and conducts Viterbi process at 4 bi-directional paths to estimate them. Structural similarity, total variation constraint, and a specific hierarchical merging strategy are combined with the Viterbi process to improve the robustness and accuracy. Based on the results of Viterbi, a convex optimization equation is derived to estimate epipolar line distortion. The estimated distortion information is used for the online compensation of Viterbi process at an auto-rectification framework. Extensive experiments were conducted to compare proposed algorithm with other practical state-of-the-art methods for intelligent vehicle applications.

1 Introduction

3D scene understanding plays an essential role for intelligent vehicle applications [17]. In these applications, passive stereo vision systems offer some significant advantages [35] to estimate depth information compared with active systems such as 3D LIDAR. In the past few years, much progress has been made towards solving the stereo matching problem [6, 23, 27, 32, 37, 38] and leads to increasingly wide applications for intelligent vehicles [14]. However, typical outdoor driving scenarios are still big challenges for stereo matching algorithms [18, 21]. In the KITTI website [18], many state-of-the-art researches can be found to overcome these challenges, such as PCBP-SS, StereoSLIC, and PCBP [36, 37], which are

based on Slanted-plane Markov Random Field model [4] and Superpixel segmentation [1]; wSGM [29], which is based on Semi Global Matching (SGM) [19] and Census transform [39]; ATGV [25], which is based on Total Generalized Variation (TGV) [8] and Census data term. It should be noticed that most top-ranked methods such as PCBP-SS, StereoSLIC, PCBP, wSGM, rSGM, and iSGM *et al*. at KITTI are based on SGM or using SGM results as initial value. In real-world applications, SGM-based algorithms are also the popular choices [14]. Specific hardware for SGM such as CPUs [15], GPUs [10], and FPGAs [6] has been implemented for these applications. Since currently SGM plays an important role in both state-of-the-art researches and practical usage , we mainly compare our algorithm with SGM in the experimental part.

To apply stereo vision in autonomous driving, we noticed that top-ranked algorithms as well as SGM itself have some problems for practical usage: *First*, the real-time or nearly real-time methods in the top 50 at KITTI only include ELAS [16], several varieties of SGM, and several methods based on Block Matching or SGM. However, 200ms/frame is a minimal requirement for autonomous driving and only a few methods can reach this requirement. *Second*, the information of small objects such as small poles at roadside, fallen objects or small animals on road area, thin fences, fire hydrants, and road curb *et al*. is crucial to applications [31]. However, all the top-ranked methods at KITTI tend to over smooth the disparity map and remove the small objects. *Third*, most stereo matching methods heavily rely on the assumption that input images have a known epipolar geometry [19], especially for the real-time ones. However, in real-world driving, this constraint may be slightly broken due to windshield distortion, thermal expansion, creep deformation and vibration *et al*. In this case, our experiments showed that SGM and ELAS generate poor results.

To solve the problems mentioned above, we propose a stereo matching algorithm named Multi-Path-Viterbi (MPV) to generate highly robust and accurate disparity map compared to state-of-the-art real-time stereo matching algorithms. Our MPV algorithm includes two parts: the first part estimates disparity by a Viterbi process [13] and the second part estimates epipolar line distortion by a convex optimization process. Two parts are combined into an online framework to do stereo matching and auto-rectification simultaneously in real-time.

The first part of algorithm has the following features: (i) We use a bi-directional Viterbi algorithm at total 4 paths to decode the matching cost space. Bi-directional idea can be found in the famous BCJR algorithm [2] to decrease the error rate. A hierarchical strategy is proposed to merge the 4 paths to further decrease the decoding error. (ii) We introduce Total Variation (TV) [9, 26] constraint into Viterbi path for approximately modeling 3D planes at different orientations to reach a similar effect as TGV [25] and Slanted-plane models [4]. (iii) The Viterbi nodes are spanned and interconnected from minimal to the maximum disparity because the disparity varies dramatically in the outdoor scenario. If the span level is n, normal Viterbi algorithm needs to perform $O(n^2)$ searching. We changed the search scheme and improved the complexity to $O(2n)$. (iv) We use structural similarity (SSIM) [33] to measure the pixel difference between left and right images at epipolar lines, instead of using Birchfield and Tomasi's pixel dissimilarity [30], sum of absolute differences [22], normalized cross correlation [24], mutual information [19], or census transform [25] *et al*.

The second part of algorithm has the following features: (i) A convex optimization equation is derived to estimate epipolar line distortion based on the output of Viterbi process. We summarize the properties of the epipolar line distortion caused by normal factors in intelligent vehicle applications. Based on these properties and inspired by the famous optical flow problem [20], we convert this distortion estimation problem to an optimization problem and employ the convex optimization theory [7] to solve it. (ii) The Viterbi process and convex

optimization are integrated into an online framework and two parts benefit each other without losing speed in this framework. It can automatically keep the epipolar line constraint to avoid the degradation of stereo matching results, which usually happens when other stereo matching methods being applied for vehicles.

Most of high ranked stereo matching algorithms are based on image segmentation. However, small objects and complicate scenarios are hardly well segmented by current image segmentation methods. Our method does not rely on any image segmentation or smoothing. It is sensitive to edge and has good performance for small objects.

Our algorithm is not only real-time but also has deterministic running time for every frame. For an image with n pixels and maximum m disparities, the time complexity of our algorithm is $O(nm)$. Unlike some segmentation-based or global-optimization-based methods, the running time of our algorithm is independent to the image content. For any 640x480 images with maximum 40 disparities, the running time is about 196ms with GTX TITAN GPU and Xeon E5-2620 CPU. This feature is helpful for process scheduling of real-time operating system and data synchronization of multi sensors as well as hardware implementation.

Unlike other auto-rectification or auto-calibration methods such as [11, 40], our auto-rectification framework does not estimate intrinsic and extrinsic matrix or the fundamental matrix but estimate the shifting through the normal of the epipolar line for every pixels. This nonparametric way makes our method be able to deal with translational, rotational and even nonlinear misalignment. Another benefit is that it is an online method. If there is a sudden change to the epipolar geometry, the system can be recovered after several hundreds of frames.

We did extensive experiments with our vehicle platform in urban and highway area. We used a 3D LIDAR (Velodyne, HDL-32E) as a ground truth to verify the accuracy of disparity maps. The results of long time driving courses in outdoor environment proved the robustness and accuracy of proposed method. Some results of outdoor experiments are presented in the paper.

2 Method

2.1 Matching Cost

Let I_0 and I_1 denote the rectified left and right images, (x,y) denote the coordination of pixel p in I_0, u denote the disparity, φ denote the $N \times N$ image patch located at $I_0(x,y)$, and ϕ denote the $N \times N$ image patch located $I_1(x-u,y)$. We use SSIM to measure the matching cost between φ and ϕ. Define $\varphi = \{\varphi_i | i = 1,2,\ldots,N^2\}$ and $\phi = \{\phi_i | i = 1,2,\ldots,N^2\}$, where φ_i and ϕ_i are pixels in the patches, and let μ_φ, σ_φ^2 and $\sigma_{\varphi\phi}$ be the mean of φ_i, the variance of φ_i, and the covariance of φ_i and ϕ_i, respectively. Approximately, μ_φ and σ_φ can be viewed as estimation of the luminance and contrast of φ. $\sigma_{\varphi\phi}$ measures the tendency of φ and ϕ to vary together and is an indication of structural difference. In [33], the luminance, contrast and structure similarity measures are given as follows:

$$l(\varphi,\phi) = \frac{2\mu_\varphi\mu_\phi + C_1}{\mu_\varphi^2 + \mu_\phi^2 + C_1}, \quad c(\varphi,\phi) = \frac{2\sigma_\varphi\sigma_\phi + C_2}{\sigma_\varphi^2 + \sigma_\phi^2 + C_2}, \quad s(\varphi,\phi) = \frac{\sigma_{\varphi\phi} + C_3}{\sigma_\varphi\sigma_\phi + C_3} \tag{1}$$

where C_1, C_2 and C_3 are small constants given by $C_1 = (K_1 L)^2$, $C_2 = (K_2 L)^2$, and $C_3 = C_2/2$ respectively. L is the dynamic range of the pixel values. $K_1 \ll 1$ and $K_2 \ll 1$ are two scalar

constants. The SSIM cost function is defined as

$$SSIM(p,u) = (1 - l(\varphi,\phi)^{\alpha} c(\varphi,\phi)^{\beta} s(\varphi,\phi)^{\gamma}) L/2 \tag{2}$$

where α, β, and γ are parameters to define the relative importance of the above three components.

2.2 Viterbi algorithm

We introduce TV constraint [26] in Viterbi path to constrain the disparity variation. As the matching cost is accumulated in the paths, the TV-constrained Viterbi is approximately equivalent to a full 2D convex optimization with TV term. Because TV constraint is applied to all the 4 paths independently, 3D planes at different orientation can be approximately modeled by at least one path. Therefore it can model the 3D objects with one or multiple slanted planes. TV constraint is useful to smooth some non-textured areas such as road or car body which are common in driving scenes but hard for stereo matching algorithms. Besides that, we also use the intensity gradient information to control the regularization level of TV constraint and make edges to be sharper. The TV constraint is expressed by defining the energy $E(u)$ on the disparity map u as follows:

$$E(u) = \sum_{p} SSIM(p,u) + \sum_{p' \in L_p} \varepsilon_{(p',u')\to(p,u)}, \quad \varepsilon_{(p',u')\to(p,u)} = \lambda e^{-|G|} |u - u'| \tag{3}$$

In the second term of $E(u)$, ε is the TV constraint modified by the gradient information G of image I_0. It penalizes all the disparity changes between p and p' which has disparity u' and belongs to p's neighbourhood L_p. λ is the parameter.

The problem of stereo matching can now be formulated as finding the disparity map u that minimizes the energy function $E(u)$. Viterbi algorithm can be used to approximate the optimum solution [28]. The Viterbi trellis in this case represents a graph of a disparity states for all pixels in one Viterbi path as shown in Fig. 1. Each node in this trellis represents assigned disparity to a pixel and each edge represents a possible disparity change between two adjacent pixels in the same Viterbi path. Let $e(p,u)$ denote the energy of node with pixel p and disparity u. We have:

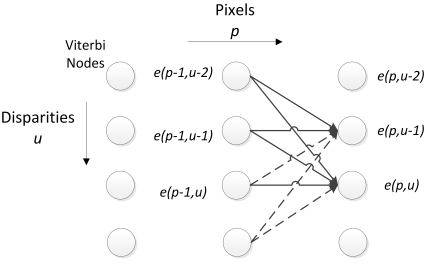


Figure 1: Trellis diagram for nodes and edges in a same Viterbi path.

$$\hat{u} = \arg\min_{u} E(u) \approx \arg\min_{u} \sum_{\text{all Viterbi paths}} e(p,u) \tag{4}$$

According to Viterbi algorithm [28]:

$$e(p,u) = \min_{u' \in L_u} \{ e(p-1,u') + \varepsilon_{(p-1,u')\to(p,u)} + SSIM(p,u) \} \tag{5}$$

Here L_u means the connected nodes from $p-1$ to (p,u). In normal Viterbi algorithm, node number of L_u is generally small and the total calculation for one pixel is $O(N(u) * N(L_u))$, where $N(\cdot)$ indicates the node number. However, in autonomous driving, dramatic disparity

variation is very common due to the large depth of field in outdoor scenes. Therefore, in our MPV algorithm, we set the L_u as all the possible Viterbi nodes. This setup can keep edge sharp for outdoor scenes. However, normal Viterbi algorithms need $O(N(u)^2)$ calculation for one pixel at this setup.

To increase the speed of Viterbi algorithm, we developed a technique to considerably decrease the number of comparison. The key idea is that we can store the comparison information at every node after finishing its calculation and use that information for the comparison at next node to decrease the comparison number. *E.g.*, at Fig. 1, we separate all the connections into up part and down part at $e(p,u-1)$ and $e(p,u)$ shown as solid lines and dash lines. For any disparity u' in all solid connections, if $(p-1,u')$ is the node at the minimum route to $e(p,u-1)$, then it is also the node at the minimum route to $e(p,u)$ except $(p-1,u) \rightarrow (p,u)$. Obviously, this observation establishes if the penalty ε is a monotonically increasing function to the disparity change. Therefore, at $e(p,u)$, we only need to compare route $(p-1,u') \rightarrow (p,u)$ and $(p-1,u) \rightarrow (p,u)$ for all solid connections. The same analysis can be applied to all dash connections. Similar idea can be found in [8, 12]. For our TV constraint case, above analysis can be simplified furthermore. After applying this trick, the total calculation for the whole image can be reduced to $O(N(u) * N(p))$ at our full connection Viterbi setup.

2.3 Path merging

We use 4 bi-directional (horizontal, vertical, and 2 diagonals) Viterbi paths on the matching space to provide good coverage of the 2D image. Horizontal directions have stronger constraints compared to other directions. In our approach, we use the results of horizontal directions as strong posterior information to calculate the optimum paths of other directions. We define four hierarchical levels and the costs of Viterbi nodes are updated based on the results of previous layer as shown in Fig. 2.

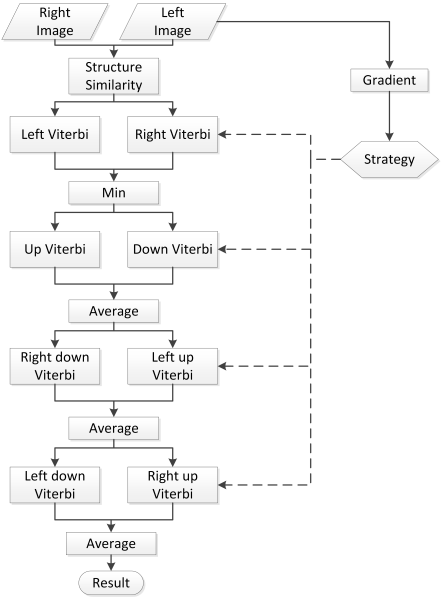


Figure 2: Hierarchical structure for the merging of multiple Viterbi paths.

In each layer, we apply bi-directional Viterbi algorithm according to Eq. (5). Then, we update the Viterbi node's energy by using optimum energy of the two opposite directions. For horizontal path, we use minimum function to sharpen edges, and for other paths we use average function to remove noises. After finishing one layer, the energy of Viterbi nodes of current layer is used as the initial value of the energy of Viterbi nodes at the next layer. More specific strategies can be applied to the path merging for every layer. *E.g.* we set a twice penalty to the left Viterbi in case of changing from small disparity to big disparity, which help to improve the performance at occluded area.

2.4 Auto-rectification

The basic idea of our method is based on the following simple observations to the distortion caused by the windshield: (1) the epipolar line distortion is fixed for a period of time; (2) the distortion is small; (3) the distortion is smooth; (4) if the distortion and disparity are both compensated then left image and right image should be approximately same.

Because the disparity map u can be regarded as pixel shifting and can be estimated by Viterbi process from I_0 and I_1, we can transform the right image to $\bar{I}_1$ as follows:

$$\bar{I}_1(x,y) = I_1(x-u,y) \tag{6}$$

Eq. (6) is also known as image warping. If there is no distortion and u is totally same as ground truth, $\bar{I}_1$ should be approximately same as I_0. Similarly, the epipolar line distortion also can be modeled as pixel shifting between left and right images. According to observation (2), the shifting at horizontal direction does not affect stereo matching. Therefore we only consider the pixel shifting at vertical direction, which breaks the epipolar line constraint of stereo matching. We define this vertical pixel shifting caused by distortion as vertical disparity and represent it as v. According to observation (4), we can write:

$$I_0(x,y) \approx \bar{I}_1(x,y+v) \tag{7}$$

Then the distortion estimation problem becomes how to estimate v by given I_0 and $\bar{I}_1$. Considering Eq. (7) and the observation (2) and (3), we can follow the same deviation to solve optical flow problem in [20] and get:

$$\hat{v} = \arg\min_v \int |\bar{I}_1(x,y+v) - I_0(x,y)|^2 + \lambda |\nabla v|^2 \,\mathrm{d}x\,\mathrm{d}y \tag{8}$$

where $|\cdot|$ is L_2 norm and λ is the parameter to control the smoothness weight of distortion.

We can solve the problem according to the convex optimization theory [7]: Rewrite I_0, $\bar{I}_1$, and v into vector form, apply 1st-order Taylor expansion to the first term of Eq. (8), and apply the anisotropic approximation [34] to the second term of Eq. (8) then we can write:

$$\hat{v} = \arg\min_v \{|\bar{I}_1(x,y+v_0) + \bar{I}_1^y \cdot (v-v_0) - I_0|^2 + \lambda |v_x|^2 + \lambda |v_y|^2\} \tag{9}$$

where $\bar{I}_1^y$ means y derivative of $\bar{I}_1$, v_0 is the initial value of v, v_x is x derivative of v, and v_y is y derivative of v. Rewrite the deviation as matrix form and transform Eq. (9) as follows:

$$\hat{v} = \arg\min_v \{|C_1 v + b|^2 + \lambda |C_2 v|^2 + \lambda |C_3 v|^2\} \tag{10}$$

where C_1 is $\bar{I}_1^y$, C_2 is the x deviation matrix, C_3 is the y deviation matrix, b is $\bar{I}_1(x,y+v_0) - \bar{I}_1^y v_0 - I_0$. We can solve Eq. (10) by solving the following equation:

$$\frac{\partial(|C_1 v + b|^2 + \lambda |C_2 v|^2 + \lambda |C_3 v|^2)}{\partial v} = 0 \tag{11}$$

Finally we have the equation:

$$(C_1^T C_1 + \lambda C_2^T C_2 + \lambda C_3^T C_3) v = -C_1^T b \tag{12}$$

Eq. (12) can be easily solved by least squares method. It should be noted that several iterations are necessary because the Taylor expansion is feasible only near v_0. Therefore v_0 is set to 0 at beginning of the iteration and is set to the result of previous iteration at other iterations.

We can calculate v for several continuous frames. According to observation (1), all the results should be approximately same. Therefore we can distinguish outliers and average all the inliers to improve the robustness. This process does not need to be run in real-time for all frames. Generally running once for several hundred frames is enough to follow the changing of v. After a v matrix is estimated, it can be used to compensate the next hundreds of images obtained by stereo cameras. The compensation process is similar as Eq. (6). The whole framework can be shown as Fig. 3.

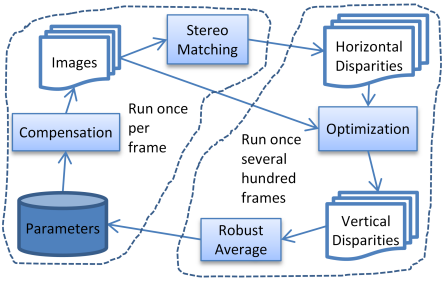


Figure 3: Framework of automatic online rectification.

3 Experiments

We did four kinds of experiments to evaluate the proposed method. *First*, we compare our algorithm with an open source SGM algorithm: OpenCV semi-global block matching (SGBM) at some ideal images. Here, we only show the comparison for small objects and non-texture area because of limited space. *Second*, we evaluate our algorithm at the classical Middlebury benchmark [27]. *Third*, we compare our algorithm with SGBM and ELAS at the KITTI benchmark [18]. *Finally*, we test the proposed algorithm in our experimental autonomous vehicle at real driving environments.

Our algorithm has very few parameters and the parameters except maximum disparities do not need to be changed for almost all scenarios. This is one of merits of our algorithm. In all of the following experiments, we set the window size of SSIM as 5x5 pixels and other parameters of SSIM as its original paper. The weight of TV constraint is set to 10 and the maximum disparity is set according to the scenarios. The parameters of SGBM and ELAS are set according to KITTI website. For the sake of convenience, we denote our multi-path-Viterbi stereo matching algorithm as MPV.

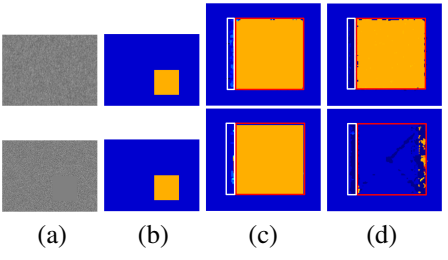
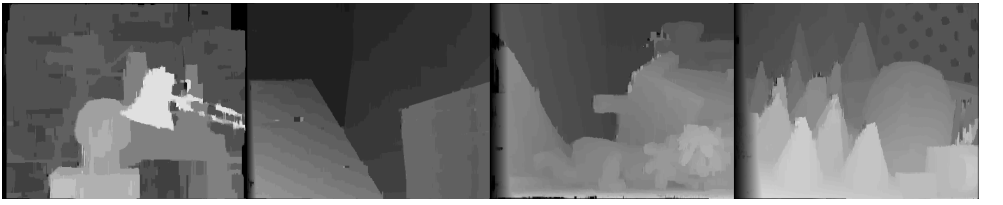


Figure 4: Random-dot stereogram test. (a): Random-dot stereogram; (b): Ground truth; (c): MPV result; (d): SGBM result. Row 1: Simulation for small objects; Row 2: Simulation for non-texture area. Red rectangle indicates the border of ground truth. White rectangle indicates the occluded area.

1. We use random-dot stereogram to generate ideal images and compare the result with SGBM as shown in Fig. 4.

The first row in Fig. 4 simulates the ideal situation with rich texture and can be considered as an extreme case of scenario with full of small objects. In this case, segmentation-based method cannot improve the result. According to Fig. 4, our method has better edge performance than SGBM. On the other hand, SGBM has better performance at the occluded area because our method currently does not apply any post processing such as left-right consistency check to remove errors at the occluded area. However, occluded areas are usually small for outdoor scenarios and only obstacles near the



(a) Tsukuba (b) Venus (c) Teddy (d) Cones

Figure 5: Middlebury Results.

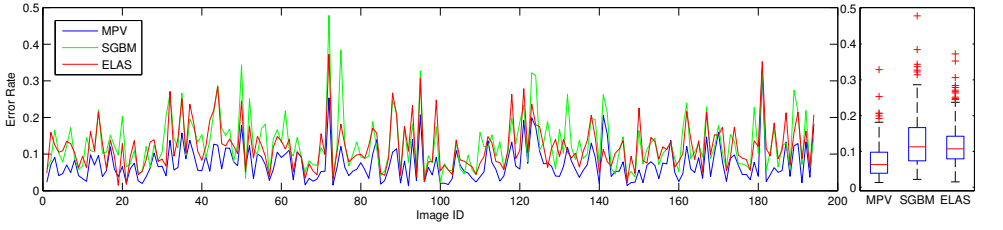


Figure 6: KITTI Results.

vehicle can generate large occluded area in the image. When it occurs, detecting the obstacle and its precise distance is more important than generating smooth disparities at the occluded area. Therefore we mainly rely on the merging strategy of Viterbi paths to handle occlusion instead of time consuming post processing methods. On the other hand, if two GPUs are equipped, the left-right consistency check can be easily applied to detect occlusion without losing speed. The second row in Fig. 4 simulates the ideal situation without any texture and can be considered as an extreme case of scenario with a large non-texture area. In this case, many algorithms including SGBM fail due to the local minimum problem. According to Fig. 4, the performance of our method is much better than SGBM.

2. We evaluate our algorithm by Middlebury benchmark and get Table 1 and Fig. 5. Table 1 shows the pixel error rate with threshold of 1 pixel. The final average error rate is 9.64%. According to Middlebury website, our method has similar result with other real-time methods. It should be noticed that our algorithm currently does not include any pre and post processing such as image smoothing, peak filter, left-right consistency check, intensity consistent disparity selection, and discontinuity preserving interpolation *et al*. Pre or post processing has significant effects especially for Middlebury images. It usually needs to be carefully tuned to handle specific structured environments and tends to lose robustness when situation changing. Especially for volatile scenarios in autonomous driving, normal post processing will generate error or lose important information at some situations.

Image	nonocc	all	disc
Tsukuba	3.54	5.26	15.9
Venus	1.16	2.57	11.6
Teddy	7.92	17.0	18.8
Cones	4.64	14.6	12.7

Table 1: Middlebury Results. nonocc: non-occluded region; all: all region; disc: regions near discontinuities.

3. We did an objective evaluation at the KITTI datasets. We use the KITTI training dataset which includes total 194 images and use the development kit in KITTI website to do the evaluation. The error rates for every image compared with SGBM and ELAS can be found at Fig. 6. The left part of the figure shows the error rate for every image in KITTI

training dataset. The right part shows the box plot of all results. Our method has 7.38% average error rate compared to SGBM's 12.88% and ELAS's 11.99%. Both the box plot and average indicate that our MPV method obviously has better performance than SGBM and ELAS. Compared with SGBM, one of the most fundamental differences is that our method has several hierarchical steps to merge the cost of multiple paths. On the contrary, SGM or its derivatives use sum of the cost at different directions such that Total Cost $=$ $\sum(\text{Direction}_1 + \text{Direction}_2 + \ldots)$. The hierarchical steps in MPV form a deep structure and can help to remove the errors generated in previous paths. Furthermore, these steps weight horizontal direction in essential. This is a good priori for the scenarios with big road area and can help to generate correct road surface in disparity map.

In Fig. 9, we compared our methods to the top-ranked algorithm PCBP-SS in KITTI for generating disparities of small poles. Our method can generate better shape for objects and can keep the small pole in disparity map very clearly compared with PCBP-SS. It verifies the simulation of small objects as shown in Fig. 4.

4. We evaluated our method on our experimental autonomous car with the stereo camera being installed outside as shown in Fig. 7. The selected stereo camera is Bumblebee BBX3-13S2C-38. We compared the distance measured by precise laser rangefinder and the distance calculated by our stereo matching algorithm. The difference between these two distances can be found in Fig. 8. The legend shows formulas which are used to calculate distance from disparity. The formula of red line is determined by the parameters which come from camera calibration. The formula of green line is determined by parameters which come from curve fitting. Theoretical error is given by the hardware specification of Bumblebee camera. The fluctuation of red and blue lines is caused by the rounding error of disparity values. The real error is mainly dominated by theoretical error and also influenced by calibration and stereo matching methods. According to Fig. 8, our stereo matching algorithm does not bring extra system errors to the results.

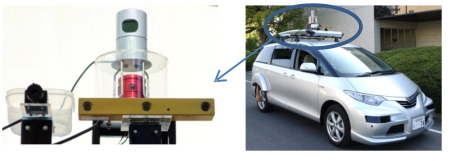


Figure 7: Our experimental car and the stereo camera.

We also tested our methods with the stereo camera being horizontally installed behind the windshield. After the camera was installed, an offline calibration and rectification had been conducted to initialize our auto-rectification framework. After a period of driving, small shifting and distortion between epipolar lines would occur. In this situation, most stereo matching algorithms will generate lots of errors without manually calibration and rectification again. Our method generates much better result in real-time with the online rectification framework. As shown in Fig. 10, column (a) is images captured by stereo cameras behind windshield, column (b) is results of our algorithm before online rectification, and column (c) is our final results after online rectification. It is clear that the errors caused by distortion are removed. Real driving videos including featured cases and typical failure cases can be found in the supplementary material.

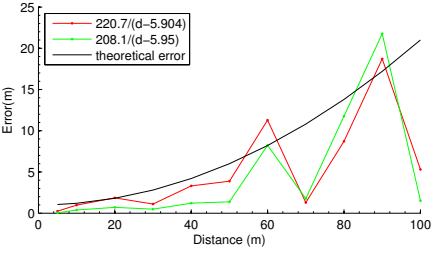


Figure 8: Distance precision.

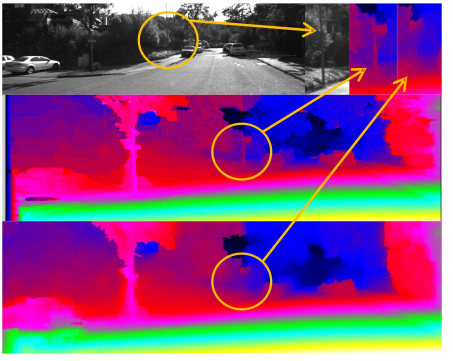
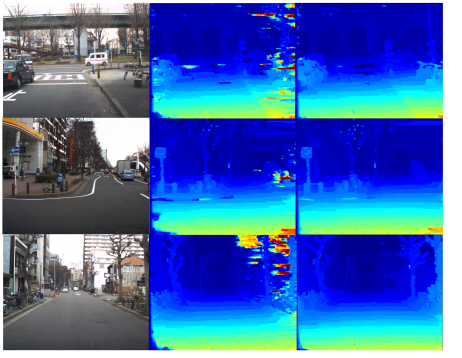


Figure 9: KITTI results. Row 1: Test image 18; Row2: Our Result; Row 3: PCBP-SS Result.



(a) Images (b) Before (c) After

Figure 10: Auto-rectification results.

4 Conclusion

This paper proposes an accurate and highly robust real-time stereo matching for Advanced Driving Safety Systems or autonomous driving. We have evaluated our algorithm with comparison to the well-known SGBM in both ideal images and real driving experiments and achieved an improvement of 5.5% to SGBM's pixel error rate at the KITTI training dataset. The experiment results have shown that the proposed method has less local minimum problems compared to SGBM and can accurately estimate the depth at pixel level for detailed structures of outdoor environments. Furthermore, real-world testing has shown that the proposed online auto-rectification framework can significantly reduce the performance degradation due to nonlinear epipolar line distortion or shifting caused by vehicle windshield or long-term driving.

Acknowledgements: This work was supported by Research Center for Smart Vehicles of Toyota Technological Institute and DENSO CORPORATION.

References

[1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Susstrunk. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 34(11): 2274–2281, 2012.

[2] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv. Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate. *IEEE Transactions on Information Theory*, 20(2): 284–287, 1974.

[3] Christian Banz, Sebastian Hesselbarth, Holger Flatt, Holger Blume, and Peter Pirsch. Real-time stereo vision system using semi-global matching disparity estimation: Architecture and FPGA-implementation. In *International Conference on Embedded Computer Systems: Architectures, Modeling and Simulation (SAMOS)*, pages 93–101. Ieee, July 2010. ISBN 978-1-4244-7936-8. doi: 10.1109/ICSAMOS.2010.5642077.

[4] Stan Birchfield and Carlo Tomasi. Multiway cut for stereo and motion with slanted surfaces. In *IEEE International Conference on Computer Vision (ICCV)*, volume 1, pages 489–495. IEEE, 1999.

[5] Michael Bleyer, Christoph Rhemann, and Carsten Rother. PatchMatch Stereo-Stereo Matching with Slanted Support Windows. In *British Machine Vision Conference (BMVC)*, volume 11, pages 1–11, 2011.

[6] Gunilla Borgefors. Distance transformations in digital images. *Computer vision, graphics, and image processing*, 34(3):344–371, 1986.

[7] Stephen P Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[8] Kristian Bredies, Karl Kunisch, and Thomas Pock. Total Generalized Variation. *SIAM Journal on Imaging Sciences*, 3(3):492–526, 2010.

[9] Antonin Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical imaging and vision*, 20(1-2):89–97, 2004.

[10] Ines Ernst and Heiko Hirschmuller. Mutual Information based Semi-Global Stereo Matching on the GPU. *Advances in Visual Computing*, (December):1–3, 2008.

[11] Olivier D. Faugeras, Q-T Luong, and Stephen J. Maybank. Camera self-calibration: Theory and experiments. In *European Conference on Computer Vision (ECCV)*, pages 321–334. Springer, 1992.

[12] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient Belief Propagation for Early Vision. *International Journal of Computer Vision (IJCV)*, 70(1):41–54, May 2006. ISSN 0920-5691. doi: 10.1007/s11263-006-7899-4.

[13] G David Forney Jr. The viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278, 1973.

[14] Uwe Franke, David Pfeiffer, Clemens Rabe, Carsten Knoeppel, Markus Enzweiler, Fridtjof Stein, and Ralf G. Herrtwich. Making Bertha See. *IEEE International Conference on Computer Vision (ICCV)*, 2013.

[15] Stefan K. Gehrig and Clemens Rabe. Real-time semi-global matching on the CPU. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 85–92. IEEE, 2010.

[16] Andreas Geiger, Martin Roser, and Raquel Urtasun. Efficient Large-Scale Stereo Matching. In *Asian Conference on Computer Vision (ACCV)*, 2010.

[17] Andreas Geiger, Martin Lauer, and Raquel Urtasun. A generative model for 3D urban scene understanding from movable platforms. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1945–1952. Ieee, June 2011. ISBN 978-1-4577-0394-2. doi: 10.1109/CVPR.2011.5995641.

[18] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3354–3361. IEEE, 2012.

[19] H. Hirschmuller. Stereo Processing by Semiglobal Matching and Mutual Information. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 30(2):328–341, February 2008. ISSN 0162-8828. doi: 10.1109/TPAMI.2007.1166.

[20] Berthold K. P. Horn and Brian G. Schunck. Determining optical flow. In *Artificial Intelligence*, volume 17, pages 185–203. International Society for Optics and Photonics, 1981.

[21] Reinhard Klette, Norbert Kruger, Tobi Vaudrey, Karl Pauwels, Marc van Hulle, Sandino Morales, Farid I Kandil, Ralf Haeusler, Nicolas Pugeault, Clemens Rabe, and Markus Lappe. Performance of Correspondence Algorithms in Vision-Based Driver Assistance Using an Online Image Sequence Database. *IEEE Transactions on Vehicular Technology*, 60(5):2012–2026, 2011. ISSN 0018-9545. doi: 10.1109/TVT.2011.2148134.

[22] Kurt Konolige, Willow Garage, and Menlo Park. Projected Texture Stereo. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 148–155, 2010.

[23] Xing Mei, Xun Sun, Mingcai Zhou, Shaohui Jiao, and Haitao Wang. On building an accurate stereo matching system on graphics hardware. In *IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 467–474, Barcelona, November 2011. IEEE. ISBN 978-1-4673-0063-6. doi: 10.1109/ICCVW.2011.6130280.

[24] Rene Ranftl, Stefan Gehrig, Thomas Pock, and Horst Bischof. Pushing the limits of stereo using variational stereo estimation. In *IEEE Intelligent Vehicles Symposium (IV)*, number 1, pages 401–407, Alcala de Henares, June 2012. Ieee. ISBN 978-1-4673-2118-1. doi: 10.1109/IVS.2012.6232171.

[25] Rene Ranftl, Thomas Pock, and Horst Bischof. Minimizing TGV-based Variational Models with Non-Convex Data Terms. In *International Conference on Scale Space and Variational Methods in Computer Vision (SSVM)*, June 2013.

[26] Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.

[27] Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision (IJCV)*, 47(1-3):7–42, 2002. doi: 10.1023/A:1014573219977.

[28] Thai Tran Son and Seiichi Mita. Stereo Matching Algorithm Using a Simplified Trellis Diagram Iteratively and Bi-Directionally. *IEICE Transactions on Information and Systems*, E89-D(1):314–325, 2006. doi: 10.1093/ietisy/e89âĂŞd.1.314.

[29] Robert Spangenberg, Tobias Langner, and Raúl Rojas. Weighted Semi-Global Matching and Center-Symmetric Census Transform for Robust Driver Assistance. In *International Conference on Computer Analysis of Images and Patterns (CAIP)*, pages 34–41. Springer, 2013.

[30] Jian Sun, Nan-ning Zheng, and Senior Member. Stereo Matching Using Belief Propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 25 (7):1–14, 2003.

[31] Sebastian Thrun. Winning the DARPA Grand Challenge. In *Knowledge Discovery in Databases: PKDD 2006*, volume 4213 of *Lecture Notes in Computer Science*, page 4. Springer Berlin Heidelberg, 2006. ISBN 978-3-540-45374-1. doi: 10.1007/11871637_4.

[32] Tinne Tuytelaars and Luc J Van Gool. Wide Baseline Stereo Matching based on Local, Affinely Invariant Regions. In *British Machine Vision Conference (BMVC)*, volume 412, 2000.

[33] Zhou Wang, Alan Conrad Bovik, Hamid Rahim Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, April 2004. ISSN 1057-7149.

[34] Manuel Werlberger, Werner Trobin, Thomas Pock, Andreas Wedel, Daniel Cremers, and Horst Bischof. Anisotropic Huber-L1 Optical Flow. In *British Machine Vision Conference (BMVC)*, pages 108.1–108.11. British Machine Vision Association, 2009. ISBN 1-901725-39-1. doi: 10.5244/C.23.108.

[35] John Iselin Woodfill, Gaile Gordon, and Ron Buck. Tyzx deepsea high speed stereo vision system. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, page 41. IEEE, 2004.

[36] Koichiro Yamaguchi, Tamir Hazan, David Mcallester, and Raquel Urtasun. Continuous Markov Random Fields for Robust Stereo Estimation. In *European Conference on Computer Vision (ECCV)*, 2012.

[37] Koichiro Yamaguchi, David Mcallester, and Raquel Urtasun. Robust Monocular Epipolar Flow Estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[38] Qingxiong Yang, Liang Wang, Ruigang Yang, Shengnan Wang, Miao Liao, and David Nister. Real-time Global Stereo Matching Using Hierarchical Belief Propagation. In *British Machine Vision Conference (BMVC)*, volume 6, pages 989–998, 2006.

[39] Ramin Zabih and J Woodfill. Non-parametric local transforms for computing visual correspondence. In *European Conference on Computer Vision (ECCV)*, pages 151–158, Stockholm, Sweden, May 1994. Springer. doi: 10.1007/BFb0028345.

[40] Frederik Zilly, Marcus Müller, Peter Eisert, and Peter Kauff. Joint estimation of epipolar geometry and rectification parameters using point correspondences for stereoscopic tv sequences. 2010.