2018 IEEE Intelligent Vehicles Symposium (IV)
Changshu, Suzhou, China, June 26-30, 2018

Real-time Stereo Disparity Quality Improvement for Challenging Traffic Environments

Yuquan Xu[1] Seiichi Mita[1] Hossein Tehrani[2] Hakusho Chin[2] Kazuhisa Ishimaru[3] Sakiko Nishino[3]

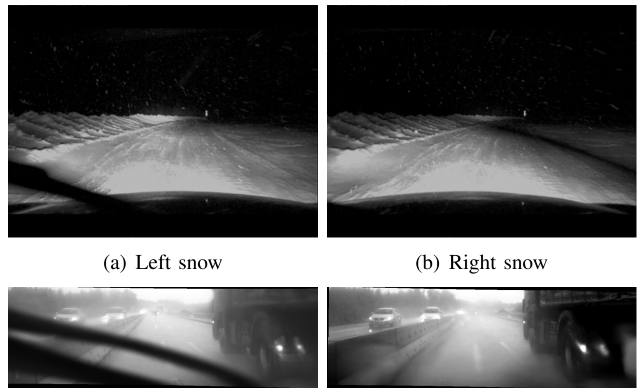
***Abstract*— The stereo vision system is one of the most important sensors for 3D reconstruction of the environment and scene understanding. It plays an essential role for the automated driving and ADAS applications. Although recently great progress has been made in this field both in the density and accuracy, the stereo vision in the challenging environment, such as rain, snow and low light conditions, are still open problems and need to be improved. To address this issue, we propose a novel real-time stereo vision algorithm, we transform the rough confidence score of the matching cost to the outlier probability rate and use it to guide the multi-path Viterbi method to estimate the disparity. Our proposed algorithm costs less than 40ms and improves the accuracy of stereo results in difficult scenarios. Real-world experiments show that the proposed method can significantly improve the disparity results in the various challenging environment.**

I. INTRODUCTION

The stereo vision simulates the binocular vision system of human eyes. It usually contains a pair of cameras which are mounted horizontally from each other (left and right camera). The aim of the stereo vision is to estimate the depth information of the scene by calculating disparities between left and right cameras. The disparity encodes the difference in horizontal coordinates of corresponding image points. The values in the disparity map are inversely proportional to the depth of the target at the corresponding pixel position.

Nowadays, the stereo vision has become important for the Advanced Driver Assistance Systems (ADAS) and automated vehicles, and much progress has been made in this field. Self-driving vehicles are on its way, but there are still a few problems to be solved. One of those is to let the automated vehicles handle the bad weather and different illumination conditions. Only a few studies aim to use the stereo vision in these challenging scenarios such as rain, snow, night and combinations thereof. The focus of this work is to improve the disparity results and to help perceiving the surrounding environment in challenging condition.

There is a basic assumption behind almost all the stereo vision algorithms that the corresponding image regions in the left and right images should be same or similar so that we can adopt different similarity metric and dynamic algorithms to find the correct disparities. However, in some challenging situations, this assumption can be violated and the causes can be broadly divided into two categories, noise and outliers.



(a) Left snow (b) Right snow

(c) Left rain (d) Right rain

Fig. 1. The outliers in the stereo pair. (a) and (b) are the stereo images from a snow and night scene. (c) and (d) are the stereo images from a rainy scene from the Stixel dataset [1]. We can see that the wipers in the left and right images are quite different which will cause great error in the disparity map

The noise is a common problem and the pictures captured by any kinds of cameras will inevitably contain varying degrees of noise especially in the low light conditions [2].

The outlier is another cause which broke the consistency of the stereo images. In this paper, the outlier means the left and right cameras see different objects. Although the stereo cameras are displaced close to each other and synchronized, there is still some difference between these two cameras when capturing close or fast objects such as wiper, flying snow and the water droplets on the windshield. In Fig. 1 we show the examples of outlier between left and right images and we can see that in the bad weather case, the left image and right image are no longer same/similar in several regions which will undoubtedly cause big errors in the stereo results.

To improve the stereo vision results for challenging environment, these two issues must be carefully considered . For the image noise, we utilize the non-local cost (NLC) method [2], which integrate the non-local means image denoising method [3] to the stereo matching cost computation and can improve the stereo vision results from noisy inputs. For the outlier issue, we compute the confidence score of the disparity based on the cost volume and then map it to the probability of the outlier. The confidence score is

[1]Yuquan Xu, Seiichi Mita are with the Research Center for Smart Vehicles, Toyota Technological Institute, 2-12-1 Hisakata, Tempaku, Nagoya, Aichi, 468-8511, Japan yuquan.xu86@toyota-ti.ac.jp, smita@toyota-ti.ac.jp

[2]Hossein Tehrani and Hakusho Chin are with the Corporate R&D 1, DENSO CORPORATION, 1-1, Showa cho, Kariya, Aichi, 448-8661 Japan , hossein_tehrani@denso.co.jp, hakusho_chin@denso.co.jp

[3]Kazuhisa Ishimaru and Sakiko Nishino are with the Research Division 2, SOKEN INC., Nishio, Aichi, 445-0012, Japan , kazuhisa_ishimaru@soken1.denso.co.jp, sakiko_nishino@soken1.denso.co.jp

a very important cue to determine whether the disparity result is good or not, and here we can use these score to roughly detect the outlier in the stereo images. Based on the calculated probability, we weight the matching cost to let the algorithm rely on the inlier parts and neglect the outlier parts. Subsequently, We use the MPV [4], [5], [2], [6] algorithm to estimate the disparity map of stereo images.

II. Related Work

The computation methods of matching cost for stereo vision include typical window-based metrics such as sum of absolute differences (SAD), normalized cross correlation (NCC) [7], census transform algorithm [8], mutual information, and recently deep learning based cost [9]. The cost aggregation algorithms contain unnormalized box filtering [10], [11], bilateral filter, non-local filtering, guided image filter[12] and cross scale cost aggregation[13]. To find the disparity from the cost, a simple winner take all (WTA) strategy can be used or other methods add explicit smoothness constraints to solve an optimization problem from the matching cost to improve the robustness. The optimization algorithms include dynamic programming [14], belief propagation [15], Viterbi [5] and graph cuts [16]. SGM [17] is a hybrid of local and global method that expands the one directional 1D scan-line optimization method to multidirectional 1D scan-line optimization. Recently, the end-to-end deep learning based depth estimation algorithms have been proposed [18] for stereo camera or monocular camera.

For the stereo vision for challenging environment, In [1], [19], [20], the temporal and scene priors are constructed from the disparity results with same camera in the good conditions and these priors are used to improve the disparity map in adverse weather conditions incorporated with SGM method. In [21] and [22], they both utilized some simple threshold based algorithm to reduce the noisy and outlier of disparities in the night scene, so their disparity results are very sparse. In this paper, we use the confidence measure to compute the outlier probability of the stereo vision. Hu et al. [23] gave a comprehensive summarize and comparison of many kinds of confidence metric.

III. Method

The whole flowchart of the proposed method can be seen in Fig. 2. To produce the smooth and accurate stereo vision result in the challenging environment, we utilize a coarse-to-fine framework. First, a Gaussian pyramid is built based on the input stereo images and we generate three layers and the downsampling rate between each layer is set to be 2. In each layer, we first generate the cost volume by the NLC method to reduce the noise effect in the stereo images. Then based on this cost volume, we compute a confidence score for each pixel and mapping this score to the probability of outliers and afterward the matching cost is weighted according to this probability. To compute the disparities from the cost, we make use of a modification of the multi-path Viterbi (MPV) algorithm followed by the weighted median filter as a post-processing step. Then, we apply upsampling to the disparity result in the coarser layer as the initial value for the finer layer and based on the initial value we limit the searching range of the finer layer to speed up the algorithm and improve the robustness.

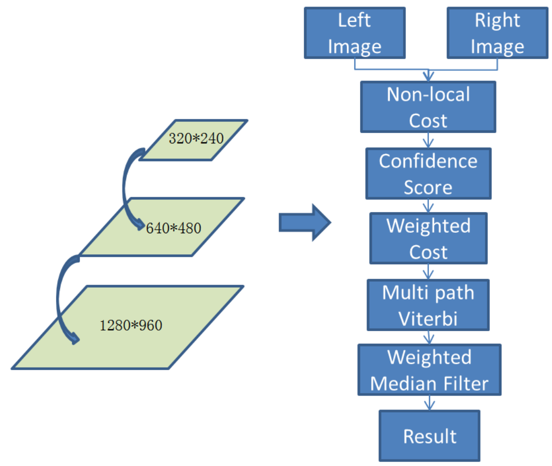


Fig. 2. The flow chart of the proposed algorithm. Left shows the coarse to fine frame work of the algorithm, we compute the disparity result in the coarser layer as the initial value of the finer layer. Right shows the detailed steps of the algorithm in each layer.

Next, we will present the details of each step of the proposed algorithm in the following sections.

IV. Non-local Matching Cost

For most existing stereo vision approaches, the first step is to compute the matching cost between the left and right images according to some kinds of similarity metric. In this paper, we adopt the NLC method to generate the cost volume which incorporates non-local idea to improve the robustness of stereo vision algorithm for noise. Specifically, we have two stereo images where I_l and I_r which denote the left and right image, respectively. We also denote N_i^l as a small square region centred at pixel i in I_l and N_j^r denotes a small square region centred at pixel j in I_r. To generate the cost volume, the distance between N_i^l and N_{i-d}^r is necessary, where $d \in \{0, 1, ..., D\}$ and $[0, D]$ is the search range of the disparities.

To compute the non-local cost, the top M similar patches of N_i^l in the left image are detected and used to denoise the patch N_i^l. Afterwards, for the right image, we directly use the same relative position of M_i to denoise all the candidate patches N_{i-d}^r as shown in Eq. (1),(2) and (3)

$$\tilde{N}_i^l = \sum_{k \in M_i} \omega(i,k) N_k^l \tag{1}$$

$$\tilde{N}_{i-d}^r = \sum_{k \in M_{i,d}} \omega(i-d,k) N_k^r \tag{2}$$

$$\omega(i,k) = \frac{1}{Z(i)} exp(-\frac{|N_i - N_k|^2}{h_1^2}) exp(-\frac{|i-k|^2}{h_2^2}) \tag{3}$$

where $Z(i)$ is the normalized factor, $\tilde{N}_i^l$ denotes the denoised patch of N_i^l, M_i is a set of positions of the most similar patches found in the neighborhood of N_i^l, $M_{i,d}$ represents the related positions generated by M_i, $\omega(i,k)$ is in the light of non-local means method [3], respectively.

We use Structural Similarity Index (SSIM) to measure the matching cost between $\tilde{N}_i^l$ and $\tilde{N}_{i-d}^r$.

V. Confidence Priors

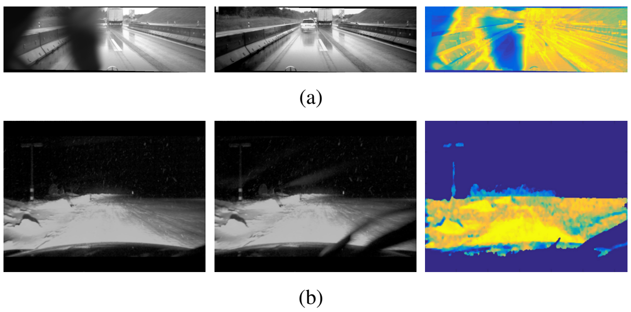


Fig. 3. The outlier weight parameter from the confidence score. The color represents the value (yellow=high, blue=low)

Once we generate the cost volume of the left and right images, we can use stereo matching algorithms to estimate the disparities from the volume. Nowadays, almost all the stereo matching algorithm will accumulate the cost value of different regions of the image to generate a robust disparity result and remove the noise. It will cause the ambiguous or wrong cost value to affect the disparity results of other parts. It is not difficult to imagine that if there are outliers in the stereo images, the matching cost in these areas will be wrong and the stereo algorithm will provide unexpected results not only in these areas but also other affected regions. As a result, we propose a weighting step after building the cost volume to reduce the artifacts caused by outliers.

The stereo confidence metrics are usually used as a threshold to remove unreliable disparities. Here, we use the confidence score as a cue of outliers and use this information to reduce the affect of outliers. In total, we focus on two confidence metrics. First of all, we choose peak-ratio naive (PKRN) from [23], which performed promising result on both indoors and outdoors scenario. We also choose the matching score measure (MSM) which is the simplest confidence measure but suitable to the outlier case.

The MSM, and PKRN metric are:

$$C_{MSM} = -c_1 \tag{4}$$

$$C_{PKRN} = \frac{c_2 + \sigma}{c_1 + \sigma} \tag{5}$$

where c_1 is the minimum cost and c_2 is the second minimum cost and σ is to avoid the singularity case and set to be 5 for the SSIM cost in our experiment. Afterwards, we mapping these two confidence score to the probability by:

$$P_{MSM} = e^{C_{MSM}/N_1} \tag{6}$$

$$P_{PKRN} = \frac{1}{1 + e^{-N_{PKRN}/N_2}} \tag{7}$$

where N_1 and N_2 are two normalized parameter which should be set based on the camera and scene, in our experiment we find $N_1 = 10$ and $N2 = 1$ can produce good results on most samples. The final probability of outlier P_o is simply computed by a product of P_{MSM} and P_{PKRN} by:

$$P_o = P_{MSM} * P_{PKRN} \tag{8}$$

Although the P_o is not guaranteed to be a probability function, we consider P_o as a pixel-wise weighting function multiplied to our the cost volume which means we reduce the cost value in the outlier area of the image and the stereo algorithm is more reliable to the inlier area . Fig. 3 shows the example of P_o in the different adverse scene. We can see that we are able to set small value in the outlier of wiper in the image and keep the details of the objects in the images.

VI. Multi-path Viterbi Algorithm

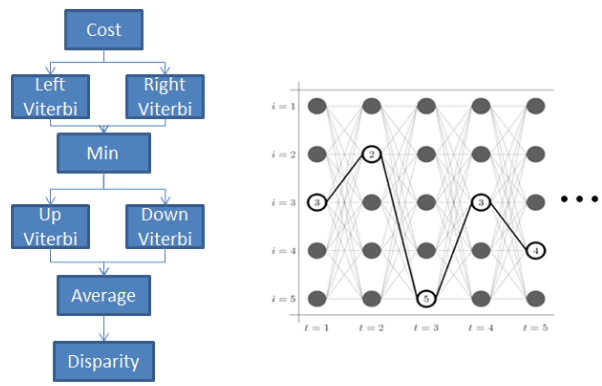


Fig. 4. The details of MPV. Left is the flowchart of proposed MPV algorithm and the Viterbi path is shown in right image

After the cost volume is generated, We add Total Variation (TV) constraint [24] to constrain the disparity variation and smooth non-textured areas. The problem of stereo matching can now be formulated as finding the disparity map D_{opt} that minimizes the energy function E(D).

$$D_{opt} = \arg\min_D E(D) = \arg\min_D C(D) + TV(D) \tag{9}$$

where $C(D)$ is the weighted cost from previous section and $TV(D)$ means the TV constraint.

The Viterbi algorithm is then used to solve the optimization problem. The Viterbi trellis represents a graph of disparity value assigned for all the pixels. The node in this graph denotes a disparity value assigned to a pixel and the edge represents a candidate disparity connection between two neighbouring pixels.

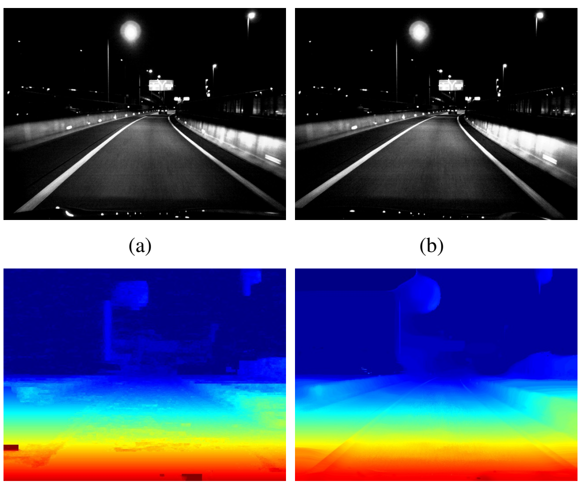
We use 4 directions (left, right, up, down) Viterbi paths to provide disparity results. Fig. 4 shows the details of our MPV algorithm.

To remove the sky noise caused by the rain, snow and ambiguous area, in the MPV process we consider an additional parameter to penalize the large value of disparity in the upper part of the image. In the upper part of the image, the Viterbi algorithm is:

$$e(i,d) = \min_{d' \in L_d} e(i-1,d') + \lambda e^{-|G|}|d-d'| + Cost(i,d) + \alpha d' \tag{10}$$

where $e(i,d)$ is the energy of node with pixel i and disparity d, L_d means the connected nodes from $i-1$ to (i,d) here L_d is set to be all the searching range of disparities, $\lambda e^{-|G|}|d-d'|$ is TV constraint and G is the gradient of the image, $Cost(p,u)$ is the cost value at (i,d), $(\alpha \times d')$ is penalty item for the large disparity value. We set $\lambda = 8$ and $\alpha = 0.5$ in all the experiment.

VII. Weighted Median Filter



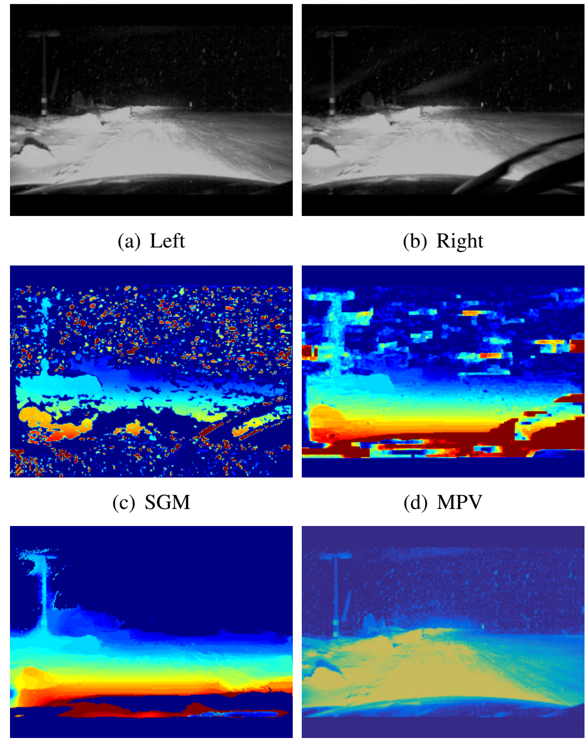
(a) (b) (c) (d)

Fig. 5. We show the weighted median filter result of a night scene. (a) is the left image of stereo camera. (b) is is the right image of stereo camera. (c) is the result before weighted median filtering. (d) is the final result of weighted median filtering

The median filter and weighted median filter are widely used as the post-processing step to refine the disparity results. In this work we utilize the constant time weighted median filter for stereo refinement [25]. The (unweighted) median filter treats each pixel in the neighborhood equally and may lead to morphological artifacts. To address this problem, the weighted median filter [25], [26] has been introduced. The pixels are weighted for the median filter, and the median filter can be done as a histogram filter.

$$h(x,i) = \sum_{x' \in N(x)} \omega(x,x')\delta(D(x')-i) \tag{11}$$

where $h(x,i)$ represents the histogram built on pixel x, i is the bin index of the histogram, $N(x)$ is the neighborhood of the pixel x, $D(x)$ is the value in the disparity map, $\omega(x,x')$ denotes the weight function of the weighted median filter, $\delta(.)$ is the Kronecker delta function. It is straightforward that we can find the median value by accumulating this histogram.



(a) Left (b) Right (c) SGM (d) MPV (e) Proposed (f) Pseudo color left

Fig. 6. The stereo results of a snow scene. (a) is gray scale left image; (b) is gray scale right image. (c) is the disparity result of SGM method [17]; The color encodes red for close and blur for far away. (d) is the disparity result of MPV method [5]; (e) is result of the proposed method; (f) we also show the pseudo color vision of left image to better visualize the snow in the sky. The color represents the object distance (red=close, blur=far)

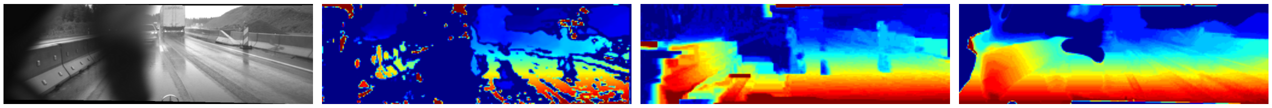
For weight function $\omega(x,x')$, we can use different weight function to generate the weights. The greatest advantage of weighted median filter is that we can compute the weight based on another image other than the disparity result only. By this we can fuse the information of other images to refine the disparities. Here we use the guided filter to compute the weight, to keep the edge of the disparity result similar to the original left image.

In Fig. 5, we show an example about the performance of WMF. We can see without WMF the proposed algorithm already can produce better stereo results and after the WMF the result is further improved by removing the noise in the sky and road surface.

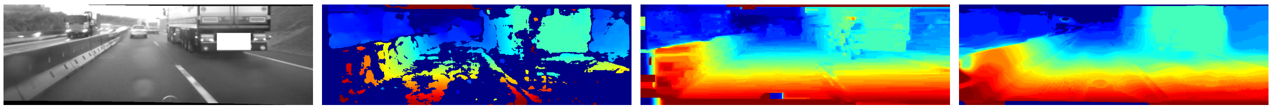
VIII. Experiment

We implement the proposed algorithm by GPU (Geforce TITAN X) and the computation time for each step is shown in Tab. I where the image size is 1280×960 and the disparity range is $[0,96]$. The proposed algorithm cost less than 40 milliseconds per frame which means 25Hz for the stereo vision system.

In Fig. 6, we show the stereo results of a snow scene which is a difficult scenario for stereo vision and in Fig. 6 (a) (b) and (f) we can see that the flying snow in the sky are quite noisy and the wipers in image pairs are different. We compare our result with the semi global matching (SGM)



(a) (b) (c) (d)



(e) (f) (g) (h)

Fig. 7. The stereo results of rainy highway scene from Ground Truth Stixel Dataset [27]. The first column is the left image of stereo camera. The second column is the disparity results of SGM. The third column is the disparity results of MPV-NLC. The last column is the disparity result of proposed method.

TABLE I
AVERAGE COMPUTATION TIME

First scale	8ms
Second scale	7 ms
Last scale	11 ms
WMF	12ms
All	38ms

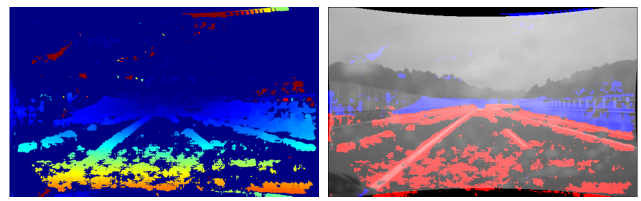
TABLE II
RESULTS

Scene	Method	PPV	TPR	F1
Rain	SGM	89.19%	62.54%	73.53%
	MPV	88.45%	90.92%	89.67%
	Proposed	**94.09%**	**95.07%**	**94.58%**
Snow	SGM	91.98%	52.69%	67.00%
	MPV	89.81%	94.45%	92.07%
	Proposed	**94.39%**	**96.92%**	**95.64%**
Night	SGM	88.77%	35.61%	50.83%
	MPV	92.65%	95.33%	93.97%
	Proposed	**93.01%**	**98.57%**	**95.71%**

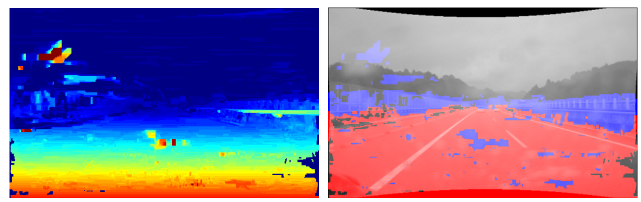
[17] and MPV [5]. In Fig. 6 (c) and (d), we can see that the disparity results of SGM and MPV are considerably affected by the snow and wiper. In Fig. 6 (e), we see the result from the proposed method to suppress the noise better than the other two algorithms.

We show the comparison results for the Stixel dataset in the Fig. 7. We can see that in the rain scene with the wiper passing, the proposed algorithm can improve the stereo results significantly compared to SGM and MPV-NLC results.

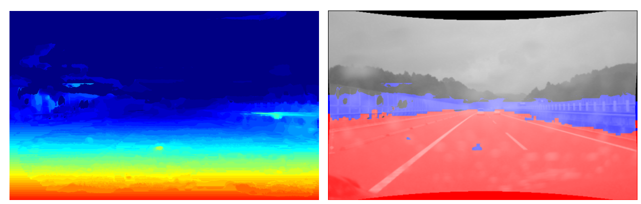
To evaluate the quantitative performance of the proposed algorithm, we construct a ground truth dataset for challenging scenarios by manually labeling the road surface of the image. A v-disparity based method is utilized to threshold the road surface from the image only using the disparity results of different algorithms. We compute the metric by comparing the segmentation results of disparity map with the labeled ground-truth. We adopt three metrics to evaluate the results, which are precision or positive predictive value(PPV), recall or true positive rate (TPR) and F1 score (harmonic mean of PPV and TPR). Although there were many stereo vision methods published, among them only a few number meet the real-time condition which is very important for the autonomous driving. As a result, two real-time algorithms,



(a) SGM



(b) MPV-NLC



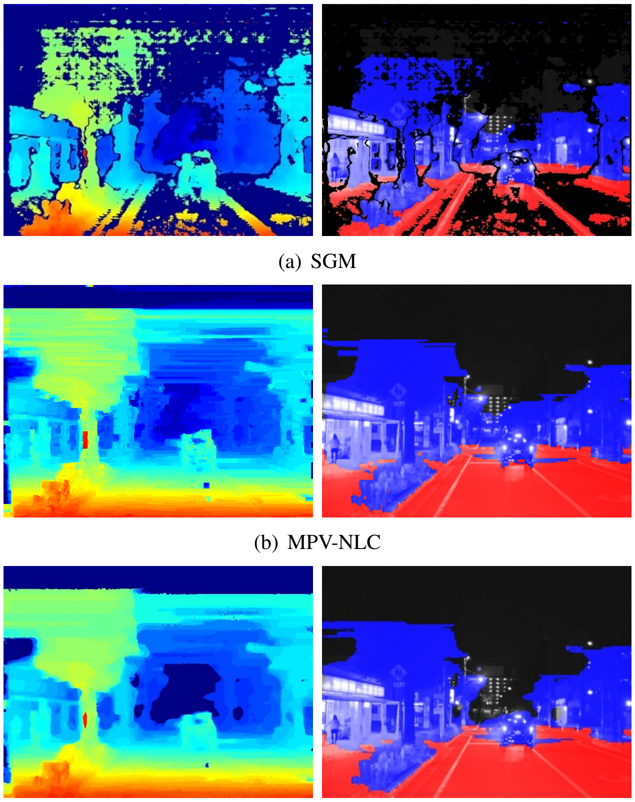
(c) Proposed

Fig. 8. A rainy scene. We compare our results with MPV-NLC [2] and SGM [17]. The first column row shows the disparity results. From top to bottom, the results are generated by SGM, MPV-NLC and the proposed algorithm, respectively. The last column shows the obstacle detection results based on the height information generated from the disparity results, where blue part represents the obstacle and red part denotes the free space.

SGM[17] and MPV-NLC [2], are selected for comparison. We construct three kinds of dataset aim to different scenarios, which are rain, snow and night. The quantitative results of three dataset can be seen in Table II. We also show the qualitative result in the Fig. 8 and 9. In these results, for rain and snow case, we can find the proposed algorithm outperforms the MPV-NLC result in the F1 score by around 2 percent. Since the disparity results generated by SGM method are very sparse, the TPR and F1 score of SGM are very low but we also can find that for the PPV score, the proposed algorithm is still better than SGM.

IX. CONCLUSION

In this paper, we proposed a real-time stereo vision algorithm which can significantly improve the disparity result in



(a) SGM

(b) MPV-NLC

(c) Proposed

Fig. 9. Night-time traffic scene. We compare our results with MPV-NLC [2] and SGM [17]. The first column row shows the disparity results. From top to bottom, the results are generated by SGM, MPV-NLC and the proposed algorithm, respectively.

the challenging environment such as snow, rain and night. We propose the confidence prior to weight the matching cost incorporated with MPV algorithm to solve the problem. The algorithm can run 25Hz in the GPU for 1280×960 stereo images with 96 disparity range. The experiment results show that the proposed method is able to eliminate the great amount of errors in the disparity result.

ACKNOWLEDGMENT

This work was supported by the Research Center for Smart Vehicles of Toyota Technological Institute, SOKEN INC. and DENSO CORPORATION.

REFERENCES

[1] S. Gehrig, M. Reznitskii, N. Schneider, U. Franke, and J. Weickert, “Priors for stereo vision under adverse weather conditions,” in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2013, pp. 238–245.

[2] Y. Xu, Q. Long, S. Mita, H. Tehrani, K. Ishimaru, and N. Shirai, “Real-time stereo vision system at nighttime with noise reduction using simplified non-local matching cost,” in *IV, 2016 IEEE*. IEEE, 2016, pp. 998–1003.

[3] A. Buades, B. Coll, and J.-M. Morel, “A non-local algorithm for image denoising,” in *Computer Vision and Pattern Recognition, 2005. CVPR 2005.*, vol. 2. IEEE, 2005, pp. 60–65.

[4] Q. Long, Q. Xie, S. Mita, K. Ishimaru, and N. Shirai, “A real-time dense stereo matching method for critical environment sensing in autonomous driving,” in *Intelligent Transportation Systems (ITSC), 2014*. IEEE, 2014, pp. 853–860.

[5] Q. Long, Q. Xie, S. Mita, H. Tehrani, K. Ishimaru, and C. Guo, “Real-time dense disparity estimation based on multi-path viterbi for intelligent vehicle applications,” in *Proceedings of the British Machine Vision Conference*. BMVA Press, 2014.

[6] Y. Xu, S. Mita, H. T. Niknejad, and K. Ishimaru, “Real-time stereo vision system at tunnel.” in *VISIGRAPP (6: VISAPP)*, 2017, pp. 402–409.

[7] J. P. Lewis, “Fast normalized cross-correlation,” in *Vision interface*, vol. 10, no. 1, 1995, pp. 120–123.

[8] R. Zabih and J. Woodfill, “Non-parametric local transforms for computing visual correspondence,” in *Computer VisionłECCV’94*. Springer, 1994, pp. 151–158.

[9] J. Zbontar and Y. LeCun, “Computing the stereo matching cost with a convolutional neural network,” *arXiv preprint arXiv:1409.4326*, 2014.

[10] F. C. Crow, “Summed-area tables for texture mapping,” *ACM SIGGRAPH computer graphics*, vol. 18, no. 3, pp. 207–212, 1984.

[11] X. Mao, Y. Cheng, Y. Xiong, D. Inoue, and M. Kagami, “Pulse position modulation for compact all-fiber vehicle laser rangefinder development,” *Optical Engineering*, vol. 56, no. 10, p. 107102, 2017.

[12] K. He, J. Sun, and X. Tang, “Guided image filtering,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 6, pp. 1397–1409, 2013.

[13] K. Zhang, Y. Fang, D. Min, L. Sun, S. Yang, S. Yan, and Q. Tian, “Cross-scale cost aggregation for stereo matching,” in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE, 2014, pp. 1590–1597.

[14] O. Veksler, “Stereo correspondence by dynamic programming on a tree,” in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2. IEEE, 2005, pp. 384–390.

[15] J. Sun, N.-N. Zheng, and H.-Y. Shum, “Stereo matching using belief propagation,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 25, no. 7, pp. 787–800, 2003.

[16] Y. Boykov, O. Veksler, and R. Zabih, “Fast approximate energy minimization via graph cuts,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 23, no. 11, pp. 1222–1239, 2001.

[17] H. Hirschmüller, “Stereo processing by semiglobal matching and mutual information,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 30, no. 2, pp. 328–341, 2008.

[18] C. Godard, O. Mac Aodha, and G. J. Brostow, “Unsupervised monocular depth estimation with left-right consistency,” in *CVPR*, 2017.

[19] S. K. Gehrig, U. Franke, and N. Schneider, “Exploiting traffic scene disparity statistics for stereo vision,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2014, pp. 674–681.

[20] S. Gehrig, N. Schneider, R. Stalder, and U. Franke, “Stereo vision during adverse weatherłusing priors to increase robustness in real-time stereo vision,” *Image and Vision Computing*, vol. 68, pp. 28–39, 2017.

[21] F. Schule, M. S. Kramer, R. Schweiger, K.-D. Kuhnert, and K. Dietmayer, “Night-vision stereo grid mapping for digital map localization,” in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*. IEEE, 2014, pp. 743–749.

[22] M. P. Philipsen, M. B. Jensen, R. K. Satzoda, M. M. Trivedi, A. Møgelmose, and T. B. Moeslund, “Day and night-time drive analysis using stereo vision for naturalistic driving studies,” in *Intelligent Vehicles Symposium Proceedings, 2015 IEEE*. IEEE, 2015, pp. 1226 – 1231.

[23] X. Hu and P. Mordohai, “A quantitative evaluation of confidence measures for stereo vision,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 11, pp. 2121–2133, 2012.

[24] L. I. Rudin, S. Osher, and E. Fatemi, “Nonlinear total variation based noise removal algorithms,” *Physica D: Nonlinear Phenomena*, vol. 60, no. 1, pp. 259–268, 1992.

[25] Z. Ma, K. He, Y. Wei, J. Sun, and E. Wu, “Constant time weighted median filtering for stereo matching and beyond,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 49–56.

[26] Q. Zhang, L. Xu, and J. Jia, “100+ times faster weighted median filter (wmf),” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2830–2837.

[27] D. Pfeiffer, S. Gehrig, and N. Schneider, “Exploiting the power of stereo confidences,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 297–304.