

# SCV-STEREO: LEARNING STEREO MATCHING FROM A SPARSE COST VOLUME

Hengli Wang<sup>\*</sup> Rui Fan<sup>†</sup> Ming Liu<sup>\*</sup>

<sup>\*</sup> Hong Kong University of Science and Technology, Hong Kong SAR, China

<sup>†</sup> Tongji University, Shanghai 201804, China

hwangdf@connect.ust.hk, rui.fan@ieee.org, eelium@ust.hk

## ABSTRACT

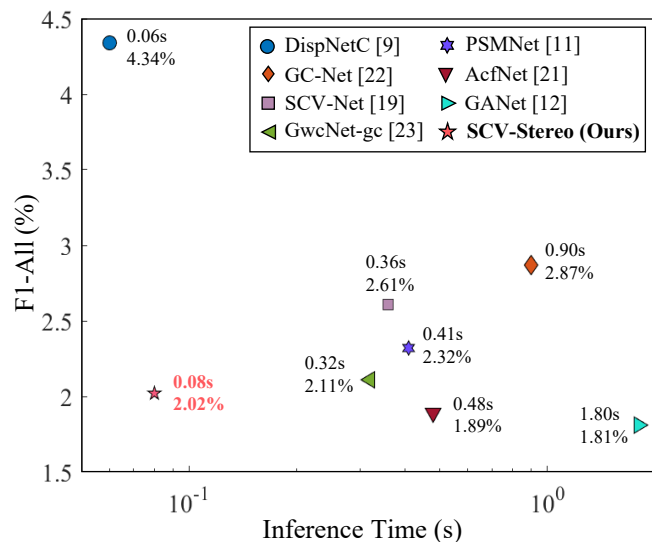
Convolutional neural network (CNN)-based stereo matching approaches generally require a dense cost volume (DCV) for disparity estimation. However, generating such cost volumes is computationally-intensive and memory-consuming, hindering CNN training and inference efficiency. To address this problem, we propose SCV-Stereo, a novel CNN architecture, capable of learning dense stereo matching from sparse cost volume (SCV) representations. Our inspiration is derived from the fact that DCV representations are somewhat redundant and can be replaced with SCV representations. Benefiting from these SCV representations, our SCV-Stereo can update disparity estimations in an iterative fashion for accurate and efficient stereo matching. Extensive experiments carried out on the KITTI Stereo benchmarks demonstrate that our SCV-Stereo can significantly minimize the trade-off between accuracy and efficiency for stereo matching. Our project page is <https://sites.google.com/view/scv-stereo>.

**Index Terms**— stereo matching, disparity estimation, sparse cost volume representation.

## 1. INTRODUCTION

Stereo matching aims at finding correspondences between a pair of well-rectified left and right images [1]. As a fundamental computer vision and robotics task [2, 3, 4, 5, 6], it has been studied extensively for decades [7]. Traditional approaches [8, 1] consist of four main steps: (i) cost computation, (ii) cost aggregation, (iii) disparity optimization, and (iv) disparity refinement [7]. With recent advances in deep learning, many data-driven approaches [9, 10, 11, 12] based on convolutional neural networks (CNNs) have been proposed for stereo matching. These approaches generally adopt a three-stage pipeline: (i) feature extraction, (ii) cost

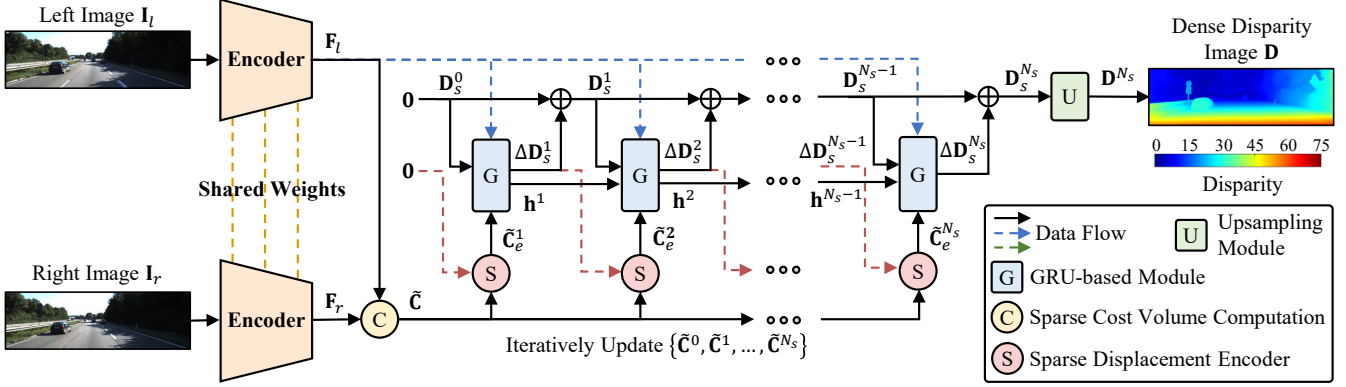
This work was supported in part by the Collaborative Research Fund by Research Grants Council Hong Kong under Project C4063-18G, in part by the Department of Science and Technology of Guangdong Province Fund under Project GDST20EG54, and in part by the Zhongshan Municipal Science and Technology Bureau Fund under project ZSST21EG06, awarded to Prof. Ming Liu.



**Fig. 1.** Evaluation results on the KITTI Stereo 2015 benchmark [13], where “F1-All” denotes the percentage of erroneous pixels over all regions. Our SCV-Stereo can greatly minimize the trade-off between accuracy and efficiency for stereo matching.

volume computation, and (iii) disparity estimation. Extensive studies have demonstrated their compelling performance as well as the tremendous potential to be a practical solution to 3D geometry reconstruction [9, 10, 11, 12].

According to the manner of cost volume computation, existing data-driven stereo matching approaches can be grouped into two classes: (i) 2D CNN-based [9, 10] and (ii) 3D CNN-based [11, 12]. The former generally adopt a correlation layer, e.g., using dot-product operation, to produce cost volumes. In contrast, the latter first perform left and right feature concatenation and then employ 3D convolution layers to generate cost volumes. While 3D CNN-based approaches perform more accurately than 2D CNN-based approaches, they generally present poor performance in memory consumption and inference speed, making them inapplicable in practice [12]. Hence, boosting the accuracy of 2D CNN-based approaches while maintaining their computational efficiency has become a promising research direction.



**Fig. 2.** An overview of our SCV-Stereo. It first extracts useful visual features to build a sparse cost volume, and then employs a novel sparse displacement encoder as well as a GRU-based module to iteratively update dense disparity estimations.

Recently, Teed *et al.*[14] proposed RAFT, a novel 2D CNN-based architecture, for optical flow estimation. Unlike other approaches for dense correspondence estimation, RAFT first constructs an all-pairs dense cost volume (DCV) by computing the pair-wise dot product between the left and right feature maps [14]. Then, it iteratively updates the optical flow estimation at a single resolution with a gated recurrent unit (GRU) [15]. Our previous work has employed this effective architecture for stereo matching [10]. However, the all-pairs DCV can consume a lot of memory and computational resources, limiting the feature maps used to only  $1/8$  image resolution [14]. This further frustrates the performance of dense correspondence estimation.

To address this problem, we propose SCV-Stereo, a novel 2D CNN-based stereo matching network, in this paper. It can greatly minimize the trade-off between accuracy and efficiency for stereo matching, as shown in Fig. 1. Our SCV-Stereo adopts a sparse cost volume (SCV) representation learning scheme, as illustrated in Fig. 2. An SCV representation only stores the best  $K$  matching costs for each pixel to avoid the intensive computation and memory consumption of existing DCV representations. We also employ a sparse displacement encoder to extract useful visual information from SCV representations. Using such sparse representations, our SCV-Stereo can iteratively update disparity estimations at  $1/4$  image resolution for accurate and efficient stereo matching.

## 2. METHODOLOGY

### 2.1. Network Architecture

SCV-Stereo generates a dense disparity image  $\mathbf{D}$  by comparing the difference between a pair of well-rectified left image  $\mathbf{I}_l$  and right image  $\mathbf{I}_r$ , as illustrated in Fig. 2. It consists of three stages: (i) feature extraction, (ii) sparse cost volume computation, and (iii) iterative disparity update. We also employ a sparse displacement encoder to provide useful information for disparity estimation. The details are introduced as follows.

#### 2.1.1. Feature Extraction

We first use two weight-sharing encoders to separately extract useful visual features  $\mathbf{F}_l$  and  $\mathbf{F}_r$  from  $\mathbf{I}_l$  and  $\mathbf{I}_r$ . Each encoder consists of several residual blocks [16].  $\mathbf{F}_l$  and  $\mathbf{F}_r$  have the size of  $H \times W \times C$  at  $1/4$  image resolution of  $\mathbf{I}_l$  and  $\mathbf{I}_r$ .

#### 2.1.2. Sparse Cost Volume Computation

Existing 2D approaches often compute a DCV, denoted as  $\mathbf{C}$ , to model the similarity for possible matching pairs between  $\mathbf{F}_l$  and  $\mathbf{F}_r$  [14], which can be formulated as follows:

$$\mathbf{C}(\mathbf{p}, \mathbf{d}) = \langle \mathbf{F}_l(\mathbf{p}), \mathbf{F}_r(\mathbf{p} - \mathbf{d}) \rangle, \quad (1)$$

where  $\mathcal{P}$  denotes the domain of  $\mathbf{F}_l$  and  $\mathbf{p} \in \mathcal{P}$  denotes the pixel coordinates;  $\mathcal{D}$  denotes the disparity candidate set and  $\mathbf{d} \in \mathcal{D}$  denotes the candidates; and  $\langle \cdot, \cdot \rangle$  denotes the dot product operation.

Referring to [1], it is unnecessary to have DCV representations for disparity estimation, and an SCV representation filling with inconsecutive and discrete matching costs is feasible to produce accurate disparity images. Therefore, we follow [17] and use an SCV representation  $\tilde{\mathbf{C}}$  that only stores the best  $K$  matching costs for each pixel:

$$\tilde{\mathbf{C}} = \{ \mathbf{C}(\mathbf{p}, \mathbf{d}) \mid \mathbf{p} \in \mathcal{P}, \mathbf{d} \in \mathcal{D}_{\mathbf{p}}^K \},$$

$$\mathcal{D}_{\mathbf{p}}^K = \arg \max_{\tilde{\mathcal{D}} \subset \mathcal{D}, |\tilde{\mathcal{D}}|=K} \sum_{\mathbf{d} \in \tilde{\mathcal{D}}} \mathbf{C}(\mathbf{p}, \mathbf{d}). \quad (2)$$

We achieve this by using a  $k$ -nearest neighbors ( $k$ NN) module [18]. Please note that only the gradients correlated to the selected  $K$  matching costs are back-propagated for CNN parameter optimization. Considering the trade-off between accuracy and efficiency, we set  $K = 8$ , and it is much smaller than the number of disparity candidates in  $\mathcal{D}$ . Moreover, after constructing  $\tilde{\mathbf{C}}$ , we only need to update the cost volume coordinates instead of re-computing dot products as usual. Such a

**Table 1.** Evaluation results of our SCV-Stereo with different setups on the Scene Flow dataset [9]. The adopted setup (the best result) is shown in bold type.

No.	Cost Volume	Resolution	$K$	AEPE (px)
(a)	Dense	1/8	–	1.05
(b)	Sparse	1/8	8	1.28
(c)	Sparse	1/4	1	2.36
(d)	Sparse	1/4	2	1.52
(e)	Sparse	1/4	4	1.01
(f)	Sparse	1/4	8	<b>0.93</b>

unique SCV representation enables our SCV-Stereo to iteratively update disparity estimations at 1/4 image resolution for accurate and efficient stereo matching.

### 2.1.3. Sparse Displacement Encoder

Inspired by RAFT [14], we adopt an iterative disparity update scheme for disparity refinement, as shown in Fig. 2. Specifically, at step  $i$ , we first estimate the residual disparity  $\Delta \mathbf{D}_s^i$ . The disparity estimation  $\mathbf{D}_s^i$  is then updated by  $\mathbf{D}_s^i = \mathbf{D}_s^{i-1} + \Delta \mathbf{D}_s^i$ . To perform this iterative disparity update scheme based on the constructed SCV representation  $\tilde{\mathbf{C}}$ , we design a sparse displacement encoder, which contains a displacement update phase and a multi-scale encoding phase.

During the displacement update phase, we update  $\tilde{\mathbf{C}}$  based on the current disparity residual for further residual estimation, *i.e.*,  $\tilde{\mathbf{C}}^i(\mathbf{p}, \mathbf{d}^{i-1} - \Delta \mathbf{D}_s^{i-1}) = \tilde{\mathbf{C}}^{i-1}(\mathbf{p}, \mathbf{d}^{i-1})$ . During the multi-scale encoding phase, we aim to encode  $\tilde{\mathbf{C}}^i$  to a dense tensor for further iterative disparity update. We first construct a five-level SCV pyramid  $\{\tilde{\mathbf{C}}_l^i\}_{l=0,1,2,3,4}$ , where  $\mathbf{d}_l^i = \mathbf{d}^i/2^l$  for  $\tilde{\mathbf{C}}_l^i(\mathbf{p}, \mathbf{d}_l^i)$ . Since the coordinates  $\mathbf{d}_l^i$  can be float numbers, we then follow [17] and linearly propagate the matching costs to the two nearest coordinates. Afterwards, we can build a 1D dense tensor (with the size of  $(2d_{\max} + 1)$ ) for each pixel  $\mathbf{p} \in \mathcal{P}$  in  $\tilde{\mathbf{C}}_l^i$  as follows:

$$\left\{ \left( \mathbf{d}_l^i, \tilde{\mathbf{C}}_l^i(\mathbf{p}, \mathbf{d}_l^i) \right) \mid \|\mathbf{d}_l^i\|_\infty \leq d_{\max}, \mathbf{d}^i \in \mathcal{D}_{\mathbf{p}}^K \right\}, \quad (3)$$

where  $\|\cdot\|_\infty$  denotes the infinity norm; and the missing matching costs are filled by 0. Based on (3),  $\{\tilde{\mathbf{C}}_l^i\}_{l=0,1,2,3,4}$  can be transformed to dense tensors, which are then reshaped and concatenated to form a 3D dense tensor  $\tilde{\mathbf{C}}_e^i$ .  $\tilde{\mathbf{C}}_e^i$  can provide useful position information and matching costs for further iterative disparity update.

### 2.1.4. Iterative Disparity Update

As mentioned above, we follow the paradigm of RAFT [14] and utilize a GRU-based module [15] to iteratively update disparity estimations  $\{\mathbf{D}_s^1, \dots, \mathbf{D}_s^{N_s}\}$  at 1/4 image resolution

(with an initialization  $\mathbf{D}_s^0 = 0$ ), as shown in Fig. 2. Specifically, in step  $i$ , we denote the concatenation of  $\mathbf{D}_s^{i-1}$ ,  $\tilde{\mathbf{C}}_e^i$  and  $\mathbf{F}_l$  as  $\mathbf{x}^i$ , and then send it to the GRU-based module with the hidden state  $\mathbf{h}^i$  to estimate the residual disparity  $\Delta \mathbf{D}_s^i$ , which then updates the disparity estimation  $\mathbf{D}_s^i$  by  $\mathbf{D}_s^i = \mathbf{D}_s^{i-1} + \Delta \mathbf{D}_s^i$ . We also use an upsampling module that consists of an upsampling layer followed by two convolution layers to provide disparity estimations at full image resolution  $\{\mathbf{D}^1, \dots, \mathbf{D}^{N_s}\}$ .

## 2.2. Loss Function

Our adopted loss function  $\mathcal{L}$  is defined as follows:

$$\mathcal{L} = \frac{1}{\sum_{i=1}^{N_s} \alpha^{N_s-i}} \sum_{i=1}^{N_s} \frac{\alpha^{N_s-i}}{N_{\tilde{\mathbf{D}}}} \sum_{\mathbf{p} \in \tilde{\mathbf{D}}} l(|\tilde{\mathbf{D}}(\mathbf{p}) - \mathbf{D}^i(\mathbf{p})|),$$

$$l(x) = \begin{cases} x - 0.5, & x \geq 1 \\ x^2/2, & x < 1 \end{cases}, \quad (4)$$

where  $l(\cdot)$  denotes the smooth L1 loss;  $\tilde{\mathbf{D}}$  denotes ground-truth labels;  $N_{\tilde{\mathbf{D}}}$  is the number of valid pixels in  $\tilde{\mathbf{D}}$ ; and we set  $\alpha = 0.8$  in our experiments, as suggested in [14].

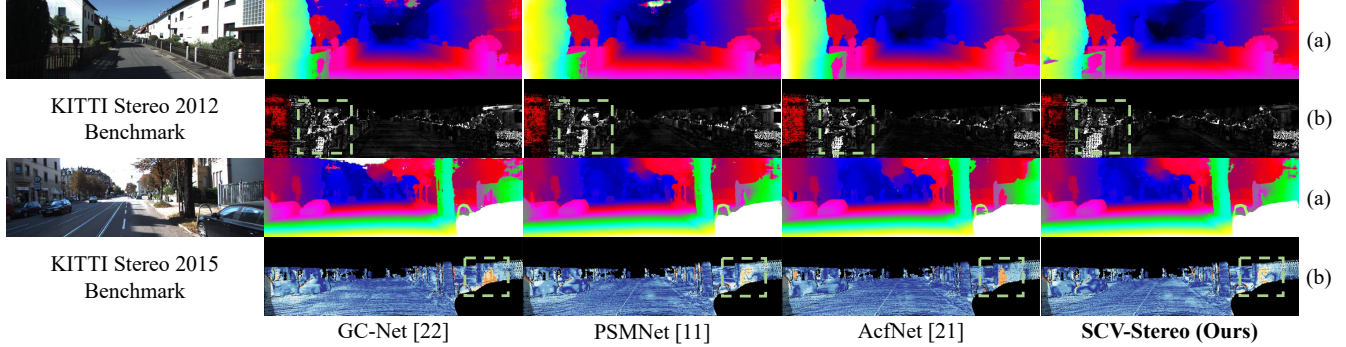
Although SCV-Net [19] was also developed based on SCV representations, there exist major differences between it and our approach. Firstly, we present a novel SCV representation and adopt an iterative disparity update scheme, which is different from SCV-Net [19]. Moreover, the novel SCV representation and network architecture enable our SCV-Stereo to greatly outperform SCV-Net [19] in terms of both accuracy and speed, as shown in Fig. 1 and Table 2.

## 3. EXPERIMENTAL RESULTS

### 3.1. Datasets and Implementation Details

In our experiments, we use three datasets, the Scene Flow dataset [9], the KITTI Stereo 2012 [20] and Stereo 2015 [13] datasets, to demonstrate the effectiveness of our approach. The Scene Flow dataset [9] was rendered from synthetic scenarios. Differently, the KITTI stereo datasets [20, 13] were created based on real-world driving scenarios. Moreover, the KITTI stereo datasets also provide public benchmarks for fair performance comparison.

For the implementation, we set  $d_{\max} = 4$  and  $N_s = 8$ . We use PyTorch to implement our model, and train it using the Adam optimizer. The training scheme follows [12, 21] for fair performance comparison. Afterwards, we use  $\mathbf{D}^{N_s}$  as the dense disparity prediction  $\mathbf{D}$  to evaluate the model’s performance. Moreover, we utilize two standard evaluation metrics, (i) the average end-point error (AEPE) that measures the difference between the disparity estimations and ground-truth labels and (ii) the percentage of the bad pixels whose error is larger than 3 pixels (F1) [20, 13].



**Fig. 3.** Examples on the KITTI Stereo benchmarks [20, 13], where rows (a) and (b) show the disparity estimations and the corresponding error maps, respectively. Significantly improved regions are highlighted with green dashed boxes.

**Table 2.** Evaluation results on the KITTI Stereo 2012<sup>1</sup> [20] and Stereo 2015<sup>2</sup> [13] benchmarks. “Noc” and “All” represent the F1 (%) for non-occluded pixels and all pixels, respectively [20, 13]. Best results are shown in bold type, and the red number indicates the ranking among these eight approaches.

Approach	KITTI 2012		KITTI 2015		Time (s)
	Noc	All	Noc	All	
DispNetC [9]	4.11 <sup>7</sup>	4.65 <sup>7</sup>	4.05 <sup>8</sup>	4.34 <sup>8</sup>	<b>0.06</b> <sup>1</sup>
GC-Net [22]	1.77 <sup>6</sup>	2.30 <sup>6</sup>	2.61 <sup>7</sup>	2.87 <sup>7</sup>	0.90 <sup>7</sup>
SCV-Net [19]	–	–	2.41 <sup>6</sup>	2.61 <sup>6</sup>	0.36 <sup>4</sup>
PSMNet [11]	1.49 <sup>5</sup>	1.89 <sup>5</sup>	2.14 <sup>5</sup>	2.32 <sup>5</sup>	0.41 <sup>5</sup>
GwcNet-gc [23]	1.32 <sup>4</sup>	1.70 <sup>4</sup>	1.92 <sup>4</sup>	2.11 <sup>4</sup>	0.32 <sup>3</sup>
AcfNet [21]	<b>1.17</b> <sup>1</sup>	<b>1.54</b> <sup>1</sup>	1.72 <sup>2</sup>	1.89 <sup>2</sup>	0.48 <sup>6</sup>
GANet [12]	1.19 <sup>2</sup>	1.60 <sup>2</sup>	<b>1.63</b> <sup>1</sup>	<b>1.81</b> <sup>1</sup>	1.80 <sup>8</sup>
<b>SCV-Stereo (Ours)</b>	1.27 <sup>3</sup>	1.68 <sup>3</sup>	1.84 <sup>3</sup>	2.02 <sup>3</sup>	0.08 <sup>2</sup>

We first conduct ablation studies on the Scene Flow dataset [9] to validate the effectiveness of our proposed architecture, as presented in Section 3.2. Then, we submit the results of our SCV-Stereo to the two KITTI Stereo benchmarks [20, 13], as shown in Section 3.3.

### 3.2. Ablation Study

Table 1 presents the performance of our SCV-Stereo with different setups on the Scene Flow dataset [9]. By comparing (a) and (b) with (f), we can see that using the feature maps with a larger resolution can greatly improve the performance, and our SCV representation is more effective than the DCV representation for stereo matching. Please note that the DCV representation can consume significant memory, and therefore, we can only use feature maps at 1/8 resolution. Moreover, (c)–(f) show that our SCV-Stereo can even work with  $K = 1$ , and

<sup>1</sup>[http://cvlibs.net/datasets/kitti/eval\\_stereo\\_flow.php?benchmark=stereo](http://cvlibs.net/datasets/kitti/eval_stereo_flow.php?benchmark=stereo)

<sup>2</sup>[http://cvlibs.net/datasets/kitti/eval\\_scene\\_flow.php?benchmark=stereo](http://cvlibs.net/datasets/kitti/eval_scene_flow.php?benchmark=stereo)

a larger  $K$  can provide a better performance. As mentioned previously, we consider the trade-off between accuracy and efficiency for stereo matching, and adopt  $K = 8$  in our SCV-Stereo. All the analysis demonstrates the effectiveness of our proposed architecture.

### 3.3. Evaluations on the Public Benchmarks

Table 2 presents the online leaderboards of the KITTI Stereo benchmarks [20, 13], and Fig. 1 visualizes the results on the KITTI Stereo 2015 [13] benchmark. It is observed that our SCV-Stereo can outperform several existing stereo matching approaches while achieving real-time performance. In addition, according to the rankings in Table 2, we can see that our SCV-Stereo greatly minimizes the trade-off between accuracy and efficiency for stereo matching. Although AcfNet [21] is a challenging baseline, it is exciting to see that our SCV-Stereo can achieve an almost 6× higher inference speed with only an accuracy decrease of 7% compared to AcfNet [21]. Moreover, Fig. 3 shows qualitative results on the KITTI Stereo benchmarks, where it can be seen that our SCV-Stereo can provide more robust and accurate disparity estimations.

## 4. CONCLUSION AND FUTURE WORK

This paper proposed SCV-Stereo, a novel stereo matching network that adopts a sparse cost volume representation learning scheme and an iterative disparity update scheme. Our approach can greatly reduce the computation complexity and memory consumption of existing dense cost volume representations while providing highly accurate disparity estimations. Experiments on the KITTI Stereo benchmarks show that our SCV-Stereo can greatly minimize the trade-off between accuracy and efficiency for stereo matching. We believe that our approach has laid a foundation for future stereo matching research where the cost volume memory consumption is no longer a limiting factor. The use of this sparse cost volume representation is also encouraging in other dense correspondence matching tasks, such as scene flow estimation.

## 5. REFERENCES

- [1] Rui Fan, Xiao Ai, and Naim Dahnoun, “Road surface 3d reconstruction based on dense subpixel disparity map estimation,” *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 3025–3035, 2018.
- [2] Hengli Wang, Yuxiang Sun, and Ming Liu, “Self-supervised drivable area and road anomaly segmentation using rgb-d data for robotic wheelchairs,” *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 4386–4393, 2019.
- [3] Hengli Wang, Rui Fan, Yuxiang Sun, and Ming Liu, “Applying surface normal information in drivable area and road anomaly detection for ground mobile robots,” in *IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2020.
- [4] Rui Fan, Hengli Wang, Peide Cai, and Ming Liu, “SNE-RoadSeg: Incorporating surface normal information into semantic segmentation for accurate freespace detection,” in *Proc. Eur. Conf. Comput. Vision (ECCV)*. Springer, 2020, pp. 340–356.
- [5] Hengli Wang, Rui Fan, Yuxiang Sun, and Ming Liu, “Dynamic fusion module evolves drivable area and road anomaly detection: A benchmark and algorithms,” *IEEE Trans. Cybern.*, 2021.
- [6] Rui Fan, Hengli Wang, Peide Cai, Jin Wu, Junaid Bocus, Lei Qiao, and Ming Liu, “Learning collision-free space detection from stereo images: Homography matrix brings better data augmentation,” *IEEE/ASME Trans. Mechatronics*, 2021.
- [7] Rui Fan, Li Wang, Mohammad Junaid Bocus, and Ioannis Pitas, “Computer stereo vision for autonomous driving,” *CoRR*, 2020.
- [8] Akihito Seki and Marc Pollefeys, “SGM-nets: Semi-global matching with neural networks,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR)*, 2017, pp. 231–240.
- [9] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox, “A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR)*, 2016, pp. 4040–4048.
- [10] Hengli Wang, Rui Fan, Peide Cai, and Ming Liu, “PVStereo: Pyramid voting module for end-to-end self-supervised stereo matching,” *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 4353–4360, 2021.
- [11] Jia-Ren Chang and Yong-Sheng Chen, “Pyramid stereo matching network,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR)*, 2018, pp. 5410–5418.
- [12] Feihu Zhang, Victor Prisacariu, Ruigang Yang, and Philip HS Torr, “GA-net: Guided aggregation net for end-to-end stereo matching,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR)*, 2019, pp. 185–194.
- [13] Moritz Menze and Andreas Geiger, “Object scene flow for autonomous vehicles,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR)*, 2015, pp. 3061–3070.
- [14] Zachary Teed and Jia Deng, “RAFT: Recurrent all-pairs field transforms for optical flow,” in *Proc. Eur. Conf. Comput. Vision (ECCV)*, 2020.
- [15] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, “Learning phrase representations using RNN encoder-decoder for statistical machine translation,” *CoRR*, 2014.
- [16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, “Deep residual learning for image recognition,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR)*, 2016, pp. 770–778.
- [17] Shihao Jiang, Yao Lu, Hongdong Li, and Richard Hartley, “Learning optical flow from a few matches,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR)*, 2021.
- [18] Jeff Johnson, Matthijs Douze, and Hervé Jégou, “Billion-scale similarity search with GPUs,” *IEEE Trans. Big Data*, 2019.
- [19] Chuanhua Lu, Hideaki Uchiyama, Diego Thomas, Atsushi Shimada, and Rin-ichiro Taniguchi, “Sparse cost volume for efficient stereo matching,” *Remote Sensing*, vol. 10, no. 11, pp. 1844, 2018.
- [20] Andreas Geiger, Philip Lenz, and Raquel Urtasun, “Are we ready for autonomous driving? the KITTI vision benchmark suite,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR)*. IEEE, 2012, pp. 3354–3361.
- [21] Youmin Zhang, Yimin Chen, Xiao Bai, Suihanjin Yu, Kun Yu, Zhiwei Li, and Kuiyuan Yang, “Adaptive unimodal cost volume filtering for deep stereo matching,” in *AAAI*, 2020, pp. 12926–12934.
- [22] Alex Kendall, Hayk Martirosyan, Saumitro Dasgupta, Peter Henry, Ryan Kennedy, Abraham Bachrach, and Adam Bry, “End-to-end learning of geometry and context for deep stereo regression,” in *Proc. IEEE Inter. Conf. Comput. Vision (ICCV)*, 2017, pp. 66–75.
- [23] Xiaoyang Guo, Kai Yang, Wukui Yang, Xiaogang Wang, and Hongsheng Li, “Group-wise correlation stereo network,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR)*, 2019, pp. 3273–3282.