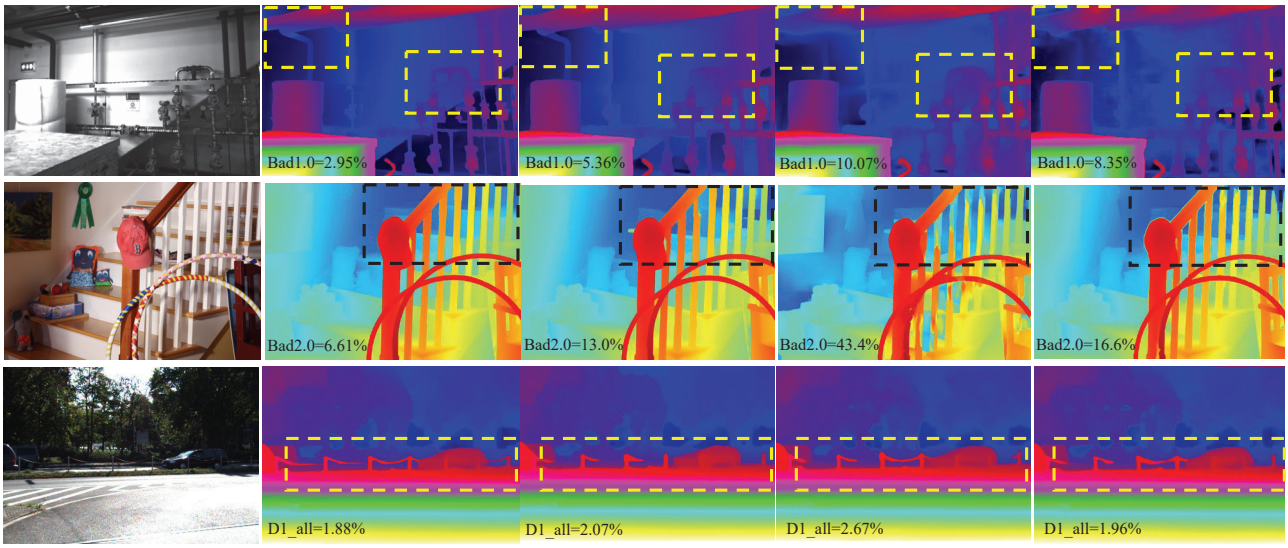
Uncertainty Guided Adaptive Warping for Robust and Efficient Stereo Matching

Junpeng Jing[1,2*] Jiankun Li[2] Pengfei Xiong[3] Jiangyu Liu[2] Shuaicheng Liu[2]
Yichen Guo[1] Xin Deng[1†] Mai Xu[1†] Lai Jiang[4] Leonid Sigal[4]
[1]Beihang University [2]Megvii Research [3]Shopee [4]University of British Columbia
{junpengjing, cindydeng, MaiXu}@buaa.edu.cn



(a) Left image (b) Ours (c) RaftStereo (d) AANet (e) CFNet

Figure 1: Visual and quantitative comparisons between our proposed method and other SOTA methods for robust stereo matching on ETH3D, Middlebury, and KITTI 2015 (from top to bottom). All results from each method are directly predicted by a single trained model with fixed parameters without any fine-tuning or adaption. Our method outperforms others both in cross-domain accuracy and details. Obvious errors and bad cases of other methods are highlighted in the box parts.

Abstract

Correlation based stereo matching has achieved outstanding performance, which pursues cost volume between two feature maps. Unfortunately, current methods with a fixed model do not work uniformly well across various datasets, greatly limiting their real-world applicability.
To tackle this issue, this paper proposes a new perspective to dynamically calculate correlation for robust stereo matching. A novel Uncertainty Guided Adaptive Correlation (UGAC) module is introduced to robustly adapt the same model for different scenarios. Specifically, a variance-based uncertainty estimation is employed to adaptively adjust the sampling area during warping operation. Additionally, we improve the traditional non-parametric warping with learnable parameters, such that the position-specific weights can be learned. We show that by empowering the recurrent network with the UGAC module, stereo matching can be exploited more robustly and effectively. Extensive experiments demonstrate that our method achieves state-of-the-art performance over the ETH3D, KITTI, and Middlebury datasets when employing the same fixed model over these datasets without any retraining procedure. To target real-time applications, we further design a lightweight model based on UGAC, which also outperforms other methods over KITTI benchmarks with only 0.6 M parameters.

1. Introduction

Stereo matching is a fundamental computer vision task [31] that aims to estimate the disparity between two rec-

*Work was done while interning at Megvii. † Corresponding authors.

tified stereo images. In the past decade, stereo matching has become increasingly popular due to the development of deep learning and the support of large synthetic datasets [26, 4]. As a result, it has a breadth of applications spanning autonomous driving [6] to 3D reconstruction [13].

Since there are significant domain differences between stereo matching datasets, existing state-of-the-art methods generally fail to achieve robust stereo matching, when applied to different datasets with a single trained model with fixed parameters. As shown in Fig. 1(a), the Middlebury dataset [32] focuses on indoor scenes with high resolution and large disparity, while ETH3D [33] contains gray-scale images at low resolution, and KITTI [12] concentrates on outdoor driving scenarios. Consequently, the leading methods [45, 24] on one dataset cannot consistently perform well across different datasets without retraining (Fig. 1(c,d)), which fails to meet the generalization requirement of real-world applications.

Large scene differences and unbalanced disparity distribution are the key reasons resulting in noisy and distorted feature maps [50], thus reducing the robustness. In addition, the limited receptive field of convolutions makes it difficult for the network to capture the global features, leading to domain sensitivity to different datasets [25]. To this end, CFNet [35] adopted an adaptive disparity range to enlarge the receptive field and alleviate the poor robustness caused by the fixed disparity range. However, it still incurs the issue of robust matching (shown in Fig. 1 (e)) because blurred textures and unclear edges in features still exist when constructing cost volume, which is generated by non-parametric warping and cannot be solved by adjusting the disparity range. Here, features are warped by the corresponding disparities and fixed sampling points in the neighborhood. Because this process utilizes constant weights, it is inherently position-agnostic and cannot capture different feature details, leading to low robustness.

In this paper, we propose an uncertainty guided adaptive correlation module to tackle the above problem, and further develop an advanced cascaded recurrent framework based on CREStereo [21], namely CREStereo++, to achieve robust stereo matching. Specifically, towards the problem caused by a fixed sampling area and limited receptive field, we employ a variance-based uncertainty estimation module to adaptively adjust the sampling range in the warping process. Moreover, we improve the traditional non-parametric warping operation with content-adaptive weights. In this way, for those areas with high uncertainty, such as textureless and occluded parts, the network adopts a wide sampling range. For parts that have achieved accurate matching, a small range of sampling area is suitable enough. Experimentally, as shown in Fig. 1 (b), our method achieves SOTA performances on all three datasets simultaneously without adaptation. To benefit real-time applications, we further propose a lightweight version, namely Lite-CREStereo++, to enable real-time performance. Our Lite-CREStereo++ outperforms all the published real-time methods with less than 60ms inference time on KITTI2012 benchmarks with only 0.6 M parameters.

The main contributions of this paper are as follows:

- We introduce a new perspective to calculate correlation dynamically for robust stereo matching that can adapt to various datasets.
- We develop an uncertainty guided adaptive warping module that enhances the robustness of the network for different scenarios, which is also valuable in general matching tasks.
- We conduct extensive experiments on commonly used benchmarks and achieve SOTA results in terms of both robustness and efficiency, making the proposed approach universal.
- Our method obtains the championship on the stereo task of Robust Vision Challenge 2022.

2. Related Works

Deep Stereo Matching. Recently, the success of convolution neural networks has driven the community to develop learning based solutions for stereo matching [48, 26, 28, 22, 5, 18, 15, 24, 43, 21]. Specifically, Mayer *et al*. [26] proposed the first end-to-end method DispNetC, which directly calculated the correlation between left and right features by multiplying the pixels at the corresponding position. Chang *et al*. introduced PSMNet [5], using a spatial pyramid pooling module to leverage the capacity of global context information in different scales. Based on this, Guo *et al*. [15] proposed GwcNet via group-wise correlation, achieving better performance and reducing parameters simultaneously. For most methods, the diversity in disparity distribution is the main challenge for model performance, which can be improved through an iterative mechanism. Following the great success of RAFT [39] in optical flow task, RaftStereo [24] was proposed for stereo matching with iterative refinement. Li *et al*. [21] proposed CREStereo, which illustrates the effectiveness of cascaded recurrent network.

Robust Stereo Matching. Robust stereo matching oriented toward robustness and real-world applications is a less explored problem. Jia *et al*. [41] introduced an end-to-end network with scene geometry priors to improve the network's generalization ability to unseen scenes. Song *et al*. [36] introduced a domain adaptation method to handle the gap between synthetic and real-world domains. Zhang *et al*. [50] proposed a domain-invariant approach via a domain normalization layer and learnable graph-based filter. MCV-MFC [23] proposed a two-stage training strategy to

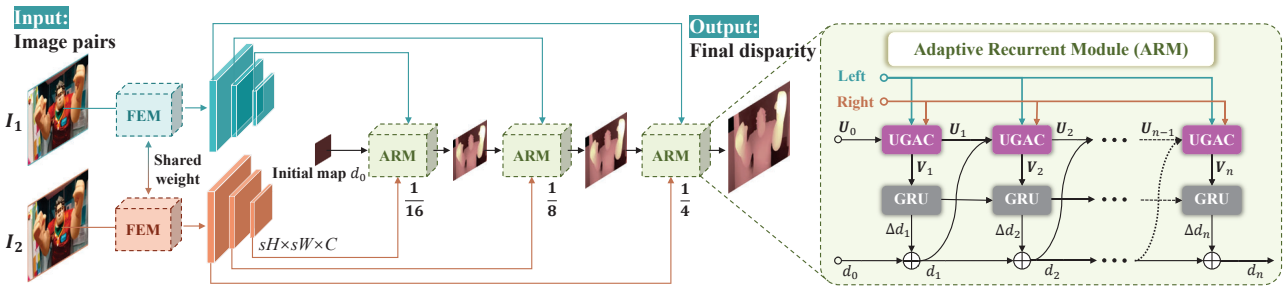


Figure 2: The overall framework of our method. It comprises two shared-weight feature extraction modules (FEM) and a reusable adaptive recurrent module (ARM). Starting from an initial input (a 0 disparity map), the output of disparity prediction in the former stage is fed to the next ARM. For each iteration in ARM, we first apply Uncertainty Guided Adaptive Correlation (UGAC) to compute the correlation between two features. The disparities are then refined with correlation using Gate Recurrent Unit (GRU). Note that ARM has the same set of parameters and is used repeatedly in each stage.

transfer the model to target datasets gently. Shen *et al*. [35] proposed CFNet, a cascaded and fused cost volume based network to deal with the domain difference, illustrating the potential of cascaded architecture for robust vision tasks. However, it still suffers from a lack of flexibility for modeling sampling in complicated structures.

Real-time Stereo Matching. Several recent works [19, 11, 3, 42, 44] focus on real-time performance while maintaining satisfactory accuracy. StereoNet [19] introduced an edge-preserving refinement network to leverage left images to recover high frequency details towards information loss at low resolution. DeepPruner [11] built a sparse cost volume by PatchMatch [3], and pruned the search space based on the predicted disparities, which were further refined under the guidance of image features. Xu *et al*. [45] proposed AANet, designed a sparse points based cost aggregation method and replaced the commonly used 3D convolutions to achieve fast inference speed. Xu *et al*. [44] introduced Fast-ACVNet, which adopted an attention mechanism to suppress redundant information and enhance matching-related information in the concatenation volume, which is quite efficient. In this paper, we also introduce a real-time stereo matching network based on the proposed network architecture while maintaining accuracy.

3. Methods

3.1. Overall Framework

Inspired by [35], a cascaded network is developed in our method to predict disparity from a low resolution to a high resolution. This way, a larger receptive field can be obtained to better learn the global structural representations. Precisely, as shown in Fig. 2, we follow the framework in [21] and design a much simplified cascaded backbone, which is composed of only two basic components without any aggregation or attention mechanism.

Given an input image pair of $\mathbf{I}_L$ and $\mathbf{I}_R$ where $\mathbf{I}_L, \mathbf{I}_R \in \mathbb{R}^{H \times W \times 3}$, two share-weighted feature extraction modules (FEM) are employed to pyramidally extract multi-scale features $\{\mathbf{F}_L^s\}, \{\mathbf{F}_R^s\} \in \mathbb{R}^{sH \times sW \times C}$. Note that $s \in \{1/4, 1/8, 1/16\}$ represents the set of down-sampled scales and C is the channel number. Then, the multi-scale features pass through 3 cascaded stages of the proposed adaptive recurrent module (ARM), which is composed of an uncertainty guided adaptive correlation (UGAC) module and a gate recurrent unit (GRU) [8]. In the ARM, the cost volume is calculated via UGAC and then input into GRU, for iteratively refining the disparity prediction results. To simultaneously enhance the robustness and preserve the details of the input, the final disparity prediction of ARM in each stage is adopted as the initial disparity of the GRU in the next stage. Note that the ARMs used in 3 cascaded stages share the same parameters, which shows a high potential to implement a lightweight model. Finally, the predicted disparity at the last stage is up-sampled to the original resolution by convex up-sampling [39].

3.2. Uncertainty Guided Adaptive Correlation

As shown in Fig. 3, the UGAC module consists of a content-aware warping layer, a correlation layer, and uncertainty estimation. For the n-th iteration of ARM, the right features $\{\mathbf{F}_R^s\}$ are first warped via the content-aware warping layer, considering the prediction disparity $\boldsymbol{d}_{n-1}$ and the uncertainty map $\boldsymbol{U}_{n-1}$ at the $(n-1)$-th iteration. Then, the cost volume $\boldsymbol{V}_n$ between the left features and the warped right features is calculated by the correlation layer. Given the cost volume $\boldsymbol{V}_n$, the uncertainty map $\boldsymbol{U}_n$ is estimated and fed to the UGAC of the next iteration. Note that $\boldsymbol{V}_n$ is used as the input of GRU.

Correlation Layer. In the correlation layer, the cost volume is calculated on the top of local correlation mechanism. Specifically, the cost volume $\boldsymbol{V}_n$ at position $\boldsymbol{p}$ can be formu-

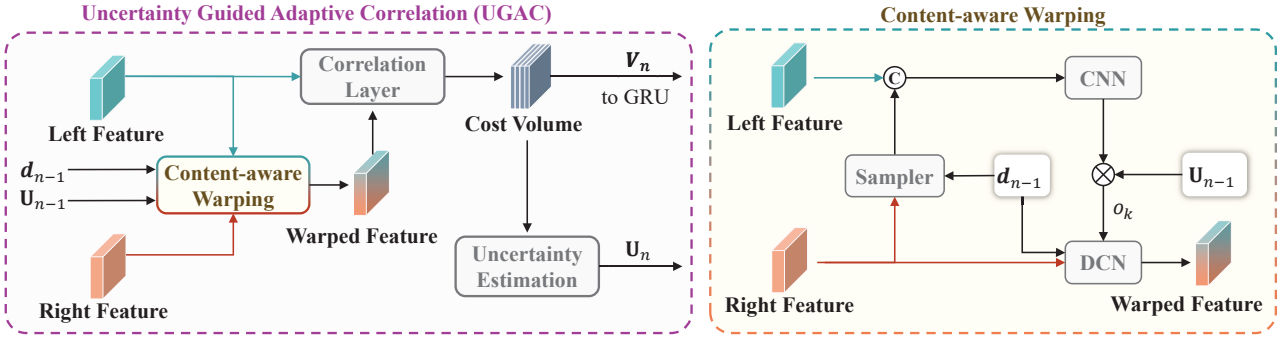


Figure 3: Left: The architecture of our uncertainty guided adaptive correlation (UGAC), is composed of a content-aware warping layer, correlation layer, and uncertainty estimation. Right: The workflow of the content-aware warping layer, where DCN represents deformable convolution network, CNN is three 3×3 convolutions with leakyReLU following each layer. For simplicity, we take the UGAC of the n-th iteration as the example.

lated as follows,

$$\boldsymbol{V}_n(\boldsymbol{p}) = \sum_{\boldsymbol{r}\in\boldsymbol{R}} \langle \mathbf{F}_L(\boldsymbol{p}) \cdot \mathbf{F}_R(\boldsymbol{p}+\boldsymbol{r}) \rangle \,, \tag{1}$$

where $\boldsymbol{R}$ denotes the search range of the current pixel in specific directions, and $\langle\cdot\rangle$ represents the channel-wise product operation.

Content-aware Warping Layer. In existing methods based on PWCNet [37], the warping layer warps right features $\mathbf{F}_R$ towards the left features $\mathbf{F}_L$ via current disparity $\boldsymbol{d}_{n-1}$ to obtain the warped right features $\hat{\mathbf{F}}_R$, formulated as follows:

$$\hat{\mathbf{F}}_R(\boldsymbol{p}) = \sum_{\boldsymbol{k}\in\boldsymbol{K}} \boldsymbol{c}_k \cdot \mathbf{F}_R\left(\boldsymbol{p} + \boldsymbol{d}_{n-1}(\boldsymbol{p}+\boldsymbol{k})\right), \tag{2}$$

where $\boldsymbol{K}$ denotes the sampling point area centered on pixel $\boldsymbol{p}$, and $\boldsymbol{d}_{n-1}(\boldsymbol{p}+\boldsymbol{k})$ represents the corresponding disparity at the position of $\boldsymbol{p}+\boldsymbol{k}$. Besides, $\boldsymbol{c}_k$ is the weight for the k-th point, usually set as a constant.

However, the above equation neglects the diversity in the warping process and adopts a content-agnostic treatment for all cases. It is thereby tough to implement "perfect" warping, leading to distorted and noisy features due to the mismatching caused by occlusions, non-texture and repetitive-texture areas. We thus calculate $\boldsymbol{c}_k$ in a content-specific manner, denoted as $\boldsymbol{w}_k(\boldsymbol{p})$. Moreover, considering the different disparity ranges and distributions in different cases, it is reasonable to adopt different sampling ranges to alleviate the domain-sensitive problem. To this end, we introduce an extra offset $\boldsymbol{o}(\boldsymbol{p},\boldsymbol{k})$ to expand the sampling range and achieve learnable warping formulated as follows:

$$\hat{\mathbf{F}}_R(\boldsymbol{p}) = \sum_{\boldsymbol{k}\in\boldsymbol{K}} \boldsymbol{w}_k(\boldsymbol{p}) \cdot \mathbf{F}_R\left(\boldsymbol{p} + \boldsymbol{d}_{n-1}(\boldsymbol{p}+\boldsymbol{k}) + \boldsymbol{o}(\boldsymbol{p},\boldsymbol{k})\right), \tag{3}$$

which is achieved with group-wise deformable convolutions [9] in practice.

Uncertainty Estimation. The ambiguity caused by traditional non-parametric warping usually accounts for a small proportion of each sample. Therefore, we expand the sampling range of ill-posed pixels and conduct adaptive pixel-level adjustments. Previous works [35, 18] have observed that ill-posed areas, texture-less regions, and occlusions tend to be multi-modal distributions with a high estimation error rate. Motivated by this, we introduce a variance-based uncertainty estimation to guide the offset $\boldsymbol{o}(\boldsymbol{p},\boldsymbol{k})$, and further balance the disparity distributions of different datasets, which is formulated as follows,

$$\boldsymbol{U}_n = 1 - \sigma\Big(\sum \big(\bar{\boldsymbol{V}}_n - \boldsymbol{V}_n\big)^2\Big), \tag{4}$$

$$\boldsymbol{o} = \boldsymbol{U}_n \cdot \mathrm{CNN}[\mathbf{F}_L, \mathcal{S}(\mathbf{F}_R, \boldsymbol{d}_{n-1})], \tag{5}$$

where $\boldsymbol{V}_n$ is cost volume and $\boldsymbol{S}$ represents bilinear sampler, $\bar{\boldsymbol{V}}_n$ represents the average value of $\boldsymbol{V}_n$, $\sigma(\cdot)$ is the sigmoid function. Through this, the network can leverage the prior knowledge of disparity prediction at the current iteration to adaptively capture more possible sampling objects.

Comparison with Existing Adaptive Mechanisms. It is worth noting that other works [45, 21] also leverage the idea of adaptive mechanisms. Here, we emphasize the critical difference in our method. In previous typical methods, some works [17, 16] calculate adaptive weights for correlation or adaptively control the window size in correlation. In AANet [45], the adaptive aggregation is conducted after warping, where a set of deformable convolutions are developed to replace the original convolutions. However, the cost volume is still built via traditional warping operation in Eq. 2, which still embeds the error during the alignment of two features. Therefore, it is necessary to refine the features in the warping process. In CREStereo [21], the cost volume

Table 1: Ablation study of the proposed method (Lite-CREStereo++) on Middlebury, KITTI2015, and ETH3D dataset. The network component is evaluated individually in each section of the table and the approach used in our final model is underlined. UE: uncertainty estimation. Inference time is measured on KITTI by V100 GPU.

Experiment	Method	Middlebury		ETH3D		KITTI15	Params.(M)	Runtime(ms)
		Bad 2.0	AvgErr	Bad 1.0	AvgErr	D1-all		
GRU Kernel Size	3×3	14.21	2.78	2.30	0.27	2.43	**0.514**	**47.9**
	1×5	12.95	2.61	2.34	0.27	2.42	0.523	53.1
	1×15	12.02	**2.46**	2.10	**0.25**	2.38	0.584	54.9
	$\underline{1 \times 5 + 1 \times 15}$	**11.86**	**2.46**	**1.98**	0.26	**2.36**	0.595	56.2
	$1 \times 5 + 1 \times 15 + 1 \times 31$	11.98	2.55	2.00	0.26	2.39	0.745	63.3
Warping	Bilinear	14.79	2.80	2.31	0.27	2.55	**0.527**	**41.0**
	Content-aware	12.96	**2.46**	2.02	**0.26**	2.47	0.595	55.8
	UE + Content-aware	**11.86**	**2.46**	**1.98**	**0.26**	**2.36**	0.595	56.2
Uncertainty Estimation	Variance + Tanh	11.98	2.46	2.01	**0.25**	2.42	0.595	**56.2**
	Variance + Sigmoid	**11.86**	**2.46**	**1.98**	0.26	**2.36**	**0.595**	**56.2**
	Error-aware + Sigmoid	12.40	2.49	2.05	**0.25**	2.40	0.595	57.1

is calculated with an adaptive position based on the local correlation. The adaptive mechanism is applied to change the matching window shape. It is ineffective to adapt the position only on the warped features. Compared with these approaches, our method conducts effective adaptation during warping before building the cost volume, alleviating the blur and inaccurate problems caused by occlusions and texture-less areas from the source. We visualize the difference between the traditional warping and ours, as shown in Fig. 4. In addition, we make full use of the prior information to produce an uncertainty map, that the adaptive mechanism is guided by a variance-based uncertainty estimation instead of directly learned by convolutions, which makes the adaptive process more reasonable and stable.

3.3. Lite-CREStereo++

We also design a lightweight version of the proposed model, namely Lite-CREStereo++, which adopts the same backbone but with fewer channels and iteration numbers. Specifically, the channel number C is reduced to 64 from 256 in the feature extraction module, which is also reduced correspondingly in the following ARM. To achieve real-time disparity prediction without sacrificing too much accuracy, we introduce an extra convolution layer with a super kernel size 1×15 in GRU, which improves the accuracy with little extra cost. The effectiveness of the lightweight model is verified in Sec. 4.3. Besides, different from the slow-fast setting in RaftStereo [24], we increase the iteration numbers of ARM from small resolution to large resolution instead. In detail, the iteration numbers are set as $2, 4$, and 6, respectively. In this way, we achieve a competent balance between accuracy and speed.

3.4. Loss Function

We supervise the optimization with l_1 distance between the ground truth disparities and the predictions to train the model in an end-to-end manner. All disparity predictions in all GRU cells are supervised with ground truth in training, while only the last disparity prediction is obtained as the final output. The total loss is formulated as follows:

$$\mathcal{L} = \sum_{s} \sum_{i=1}^{n} \gamma^{n-i} ||\mathbf{d}_{\text{gt}} - \mathcal{S}(\mathbf{d}_i^s)||_1, \tag{6}$$

where the exponentially weight γ is set to 0.8, and $\mathcal{S}(\mathbf{d}_i^s)$ represents the predictions after sampler $\mathcal{S}$.

4. Experiments

More details about datasets, implementation, and evaluation can be seen in the supplementary materials.

4.1. Datasets

For training, several public datasets are used, including Middlebury [30], ETH3D [33], KITTI [27], SceneFlow [26], Sintel [4], Falling Things [40], InStereo2K [2], Carla [10], and the dataset proposed in [21]. For evaluation, following the previous methods, we adopt the commonly used benchmarks, including Middlebury 2014 [30] (full resolution), ETH3D [33], and KITTI 2012/2015 [27].

4.2. Implementation Details

We conduct a two-stage training strategy to train the proposed method. First, during the pre-training process, all the datasets above are used except KITTI. Since the ground truth of KITTI is sparse, with more than $1/4$ pixels masked, adding KITTI at an early stage will reduce the

Table 2: Robustness comparison among ETH3D, Middlebury, and KITTI2015 testsets with existing SOTA methods in RVC. All methods are tested on three datasets with a single fixed model. The overall rank is obtained by Schulze Proportional Ranking [34] to combine multiple rankings into one. Our approach achieves the best overall performance.

Method	Middlebury				KITTI2015				ETH3D				Overall
	bad 1.0	bad 2.0	AvgErr	Rank	D1-bg	D1-fg	D1-all	Rank	bad 1.0	bad 2.0	AvgErr	Rank	Rank
AANet_RVC [45]	42.9	31.8	12.8	10	2.23	4.89	2.67	10	5.41	1.95	0.33	10	12
CVANet_RVC	58.5	38.5	8.64	11	1.74	4.98	2.28	9	4.68	1.37	0.34	9	11
GANet_RVC [49]	43.1	24.9	15.8	11	1.88	4.58	2.33	7	6.97	1.25	0.45	10	10
HSMNet_RVC [46]	31.2	16.5	3.44	6	2.74	8.73	3.74	12	4.40	1.51	0.28	8	9
MaskLacGwcNet_RVC [15]	31.3	15.8	13.5	8	1.65	3.68	1.99	5	6.42	1.88	0.38	12	8
GEStereo_RVC	22.8	14.1	3.78	3	2.29	4.79	2.71	11	3.95	1.25	0.29	6	7
CroCo_RVC	32.9	19.7	5.14	9	2.04	3.75	2.33	7	**1.54**	0.50	0.21	2	6
NLCANet_V2_RVC [29]	29.4	16.4	5.60	7	**1.51**	3.97	1.92	3	4.11	1.20	0.29	6	5
CFNet_RVC [35]	26.2	16.1	5.07	5	1.65	3.53	1.96	3	3.70	0.97	0.26	5	4
iRaftStereo _RVC [24]	24.0	13.3	2.90	2	1.88	3.03	2.07	6	1.88	0.55	0.17	3	3
raft+_RVC [39]	22.6	14.4	3.86	4	1.60	**2.98**	**1.83**	**1**	2.18	0.71	0.21	4	2
CREStereo++_RVC (ours)	**16.5**	**9.46**	**2.20**	**1**	1.55	3.53	1.88	2	1.70	**0.37**	**0.16**	**1**	**1**

(a) left image (b) disparity image

(c) traditional warping (d) our warping

Figure 4: Visual comparison between traditional warping and ours. The feature warped by traditional method has obviously blurry edges and distortions, while warped by our method has sharper details, as indicated by the red arrows.

overall strength of supervised training, thereby weakening the network's ability of precise matching. Thus, we remove KITTI from the trainset at the pre-training stage. Due to the different quantities of each dataset, we have balanced their proportions in advance. The number of iterations is set to 150k with a learning rate of 4×10^{-4} using Adam [20] optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. To balance input data from various aspect ratios, the group of stereo images and disparity are first resized to a similar size and then cropped to 384×512. Second, in the fine-tuning stage, KITTI 2012/2015 is reintroduced for another 50k iterations with a much lower learning rate 1×10^{-4} after model convergence, and its proportion is adjusted to half of the whole training set. Considering the aspect ratio in KITTI is much larger than other datasets (> 3), during the fine-tuning stage, the input size is set to 256×512 for CREStereo++_RVC and 384×1248 for Lite-CREStereo++.

4.3. Ablation Study

As shown in Table 1, we study a specific component of our approach in isolation and underline the settings used in final model. Experiments are conducted on the lite model.

GRU Kernel Size. We explore the effect of different kernel sizes in GRU. Specifically, the kernel size is increased from 1×5 to 1×31, growing $2\times$ each time with different combination ways. The commonly used 3×3 kernel is also tested. From the table "GRU Kernel Size", we can see the combination of 1×5 and 1×15 achieves the best overall performance. Although it takes 8ms longer time consumption than 3×3 kernel, it achieves 2.35 and 0.32 improvement on Middlebury and ETH3D, respectively.

Warping Types. In order to compare the performance of different types of warping, we replace our warping layers with other forms. Specifically, "UE" represents uncertainty estimation and "Content-aware" denotes the warping operation in Eq. 3. As shown in the table, the proposed uncertainty guided adaptive warping achieves the best performance with an acceptable computation complexity. Besides, compared with the traditional bilinear warping operation, learnable warping without uncertainty still has significant advantages, which illustrates the effectiveness of the proposed uncertainty estimation and deformable warping. Visualization of the difference between traditional warping and ours can be seen in Fig. 4. Compared to traditional warping, our method has obviously sharper feature details. The original method causes the warping of hair and leg areas to be misled by the background.

Uncertainty Estimation. We also explore the effect of different uncertainty estimation approaches, as shown in the table "Uncertainty Estimation". Error map means the guided map for deformable warping is calculated directly

Table 3: Cross-domain robustness evaluation on ETH3D, Middlebury, and KITTI2012/2015 trainsets. All methods are only trained on the Scene Flow dataset and evaluated on each dataset with fixed parameters.

Method	Middlebury bad 2.0	KITTI2012 D1-all	KITTI2015 D1-all	ETH3D bad 1.0
PSMNet [5]	39.5	15.1	16.3	23.8
GWCNet [15]	37.4	12.0	12.2	11.0
CasStereo [14]	40.6	11.8	11.9	7.8
GANet [49]	32.2	10.1	11.7	14.1
DSMNet [50]	21.8	6.2	6.5	6.2
LEAStereo [7]	31.3	9.0	9.4	9.0
CFNet [35]	28.2	**4.7**	5.8	5.8
RAFT-Stereo [24]	21.6	**4.7**	5.5	7.8
CREStereo [21]	15.3	6.7	6.7	5.5
CREStereo++(ours)	**14.8**	**4.7**	**5.2**	**4.4**

by the difference between left image and warped right image, which has limited improvements. Variance-based approach has better performance. It can also be observed that variance with sigmoid is slightly better than with tanh.

4.4. Robustness Evaluation

Robustness measures the generalization ability of a model using a specific set of parameters, which is of great significance in practical applications. Many existing methods are limited to a specific area and only make steady progress on each individual dataset, but cannot obtain comparable results on multiple datasets. To this end, we conduct robustness experiments. More experimental results can be seen in supplementary materials.

Domain Transfer Evaluation. Table. 2 displays the results of our method and existing SOTA methods in stereo matching of robust vision challenge (RVC). We conduct comparison experiments following previous RVC settings in CFNet[35]. In RVC, all methods are evaluated on three real-world public benchmarks with a single fixed model, that has the same model parameters without fine-tuning. As can be seen, raft+_RVC achieves 1st on KITTI2015 among all the methods. However, it fails to obtain comparable results on the other two datasets (4th on ETH3D and Middlebury, respectively), which are far worse than the other top three methods. Similar situations occur in iRaftStereo_RVC, which ranks 2nd on Middlebury and 3rd ETH3D, but ranks 6th on KITTI2015. In contrast, our method shows strong robustness ability and performs well on all three datasets. We get 1st place on ETH3D and Middlebury, outperforming other methods with a large margin, and 2nd on KITTI2015, achieving the best overall performance. Visual comparisons are shown in Fig. 1. In the ETH3D samples, it can be seen that other methods have noticeable disparity distortion at the location of the water pipe (yellow box in the figure). In comparison, our method produces sharper object boundaries and better preserves the overall structures. Similar phenomena exist in Middlebury and KITTI2015.

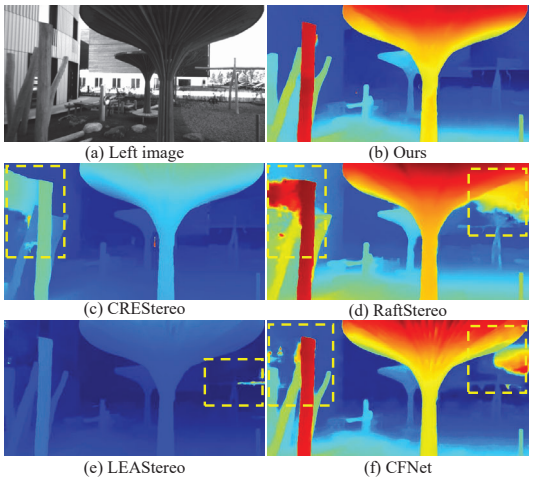


Figure 5: Visual comparisons on ETH3D train sets with existing SOTA methods. All models are trained only on Scene Flow. Zoom in for a best view.

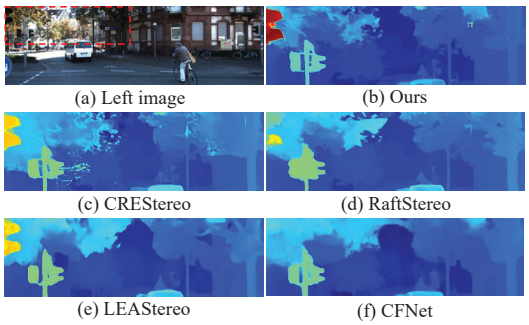


Figure 6: Visual comparisons on KITTI2015 train sets with existing SOTA methods. All models are trained only on Scene Flow. Zoom in for a best view.

Cross Domain Evaluation. Following the experiments in [35], we conduct cross-domain generalization evaluation to further emphasize the effectiveness of our method. As shown in Table. 3, all methods are only trained on synthetic dataset Scene Flow and evaluated on four real datasets, ETH3D, Middlebury, and KITTI2012/2015 trainsets, with fixed parameters. Our method still achieves the best performance on all four datasets, also surpassing the robust methods DSMNet [50] and CFNet [35]. Visual comparisons on ETH3D and KITTI2015 trainsets are shown in Fig. 5 and Fig. 6 respectively.

Table 4: Quantitative evaluation of real-time stereo matching on the online test sets of KITTI 2012 and KITTI 2015. We adopt other SOTA real-time approaches to illustrate the efficiency of the proposed Lite-CREStereo++.

Method	KITTI 2012						KITTI 2015			Params.(M)	Runtime(ms)
	3-noc	3-all	4-noc	4-all	EPE-noc	EPE-all	D1-bg	D1-fg	D1-all		
DispNetC [26]	4.11	4.65	2.77	3.20	0.9	1.0	4.32	4.41	4.34	42.32	60
DeepPrunerFast [11]	–	–	–	–	–	–	2.32	3.91	2.59	7.39	50
AANet [45]	1.91	2.42	1.46	1.87	0.5	0.6	1.99	5.39	2.55	3.93	60
DecNet [47]	–	–	–	–	–	–	2.07	3.87	2.37	–	50
BGNet [42]	1.77	2.15	–	–	0.6	0.6	2.07	4.74	2.51	2.97	44
BGNet+ [42]	1.62	2.03	1.16	1.48	0.5	0.6	1.81	4.09	2.19	5.31	48
CoEx [1]	1.55	1.93	1.15	**1.42**	0.5	0.5	1.79	3.82	2.13	2.70	33
HITNet [38]	**1.41**	1.89	1.14	1.53	0.4	0.5	**1.74**	**3.20**	**1.98**	0.63	**31**
Fast-ACVNet [44]	1.68	2.13	1.23	1.56	0.5	0.6	1.82	3.93	2.17	3.08	45
Lite-CREStereo++ (ours)	1.43	**1.82**	**1.12**	1.44	0.5	0.5	1.79	3.53	2.08	**0.60**	56

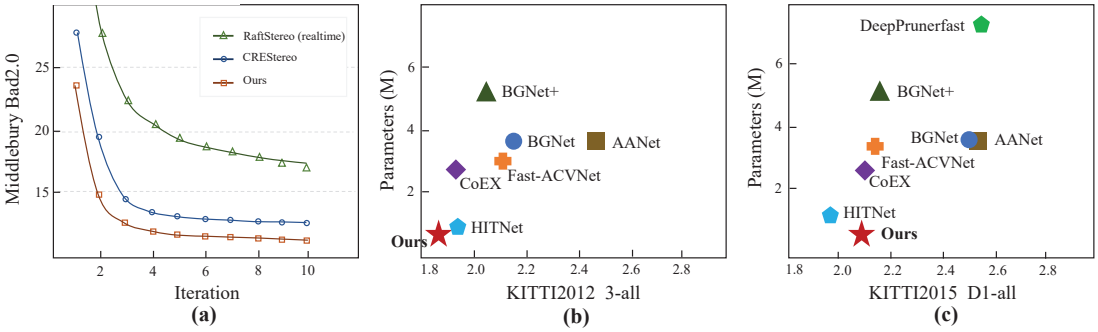


Figure 7: (a) Iterations vs. Bad 2.0 on the Middlebury dataset. (b) 3-all errors vs. Parameters on the KITTI 2012 leaderboard. (c) D1-all error vs. Parameters on the KITTI 2015 leaderboard. Our method outperforms other SOTA methods.

Table 5: Cross-domain generalization evaluation for real-time methods. All methods are only trained on SceneFlow.

Method	Middlebury bad 2.0	KITTI2012 D1-all	KITTI2015 D1-all	ETH3D bad 1.0
AANet [45]	43.8	11.3	12.6	11.4
AANet+ [45]	39.4	8.8	9.0	13.1
BGNet [42]	30.4	6.2	6.6	10.1
HITNet [38]	28.9	**5.9**	**6.5**	10.6
Lite-CREStereo++	**27.5**	6.0	7.0	**9.9**

4.5. Efficiency Evaluation

Since more iteration numbers lead to increased time costs, we analyze the relationship between iteration numbers and performance. Fig. 7 (a) shows the experiment conducted with different recurrent methods [21, 24] under similar time costs at a certain iteration. The performance of our method with 6 iterations can outperform other 15 iterations methods, which do not require a large iteration time.

As shown in Table. 4, we conduct experiments for the proposed lite version method (Lite-CREStereo++) on KITTI2012 and KITTI 2015 online benchmarks. Under similar inference speed to real-time methods, Lite-CREStereo++ achieves SOTA results among all published real-time methods on KITTI2012 benchmark. Meanwhile, it outperforms most published methods on KITTI2015 benchmark at the time of writing. We also note that most existing methods have $4\times$ more parameters than ours, and our method performs much better than these methods, as shown in Fig. 7 (b) and (c). We also conduct cross-domain generalization evaluation for existing real-time methods. From Table. 5 we can see our method still keeps a high robustness ability, outperforming other methods.

5. Conclusion

In this paper, we show that a content-aware warping module based on uncertainty estimation improves the performance of stereo matching, especially on the aspect of robustness. Combined with cascaded architecture and recurrent mechanism, we propose CREStereo++ to recover disparity for robust stereo matching. Moreover, we design a lightweight model with real-time performance. Experimental results show that our approach performs well on various datasets, and has generic applicability. The future direction would be extending our method to other warping-based cost volume tasks, such as multi-view stereo and optical flow.

References

[1] Antyanta Bangunharcana, Jae Won Cho, Seokju Lee, In So Kweon, Kyung-Soo Kim, and Soohyun Kim. Correlate-and-excite: Real-time stereo matching via guided cost volume excitation. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3542–3548. IEEE, 2021.

[2] Wei Bao, Wei Wang, Yuhua Xu, Yulan Guo, Siyu Hong, and Xiaohu Zhang. Instereo2k: a large real dataset for stereo matching in indoor scenes. *Science China Information Sciences*, 63(11):1–11, 2020.

[3] Frederic Besse, Carsten Rother, Andrew Fitzgibbon, and Jan Kautz. Pmbp: Patchmatch belief propagation for correspondence field estimation. *International Journal of Computer Vision*, 110(1):2–13, 2014.

[4] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *ECCV*, pages 611–625, 2012.

[5] Jia-Ren Chang and Yong-Sheng Chen. Pyramid stereo matching network. In *CVPR*, pages 5410–5418, 2018.

[6] Chenyi Chen, Ari Seff, Alain Kornhauser, and Jianxiong Xiao. Deepdriving: Learning affordance for direct perception in autonomous driving. In *Proceedings of the IEEE international conference on computer vision*, pages 2722–2730, 2015.

[7] Xuelian Cheng, Yiran Zhong, Mehrtash Harandi, Yuchao Dai, Xiaojun Chang, Tom Drummond, Hongdong Li, and Zongyuan Ge. Hierarchical neural architecture search for deep stereo matching. *arXiv preprint arXiv:2010.13501*, 2020.

[8] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014.

[9] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017.

[10] Jean-Emmanuel Deschaud. Kitti-carla: a kitti-like dataset generated by carla simulator. *arXiv preprint arXiv:2109.00892*, 2021.

[11] Shivam Duggal, Shenlong Wang, Wei-Chiu Ma, Rui Hu, and Raquel Urtasun. Deeppruner: Learning efficient stereo matching via differentiable patchmatch. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4384–4393, 2019.

[12] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, pages 3354–3361, 2012.

[13] Andreas Geiger, Julius Ziegler, and Christoph Stiller. Stereoscan: Dense 3d reconstruction in real-time. In *2011 IEEE intelligent vehicles symposium (IV)*, pages 963–968. Ieee, 2011.

[14] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2495–2504, 2020.

[15] Xiaoyang Guo, Kai Yang, Wukui Yang, Xiaogang Wang, and Hongsheng Li. Group-wise correlation stereo network. In *CVPR*, pages 3273–3282, 2019.

[16] Yong Seok Heo, Kyoung Mu Lee, , and Sang Uk Lee. Robust stereo matching using adaptive normalized cross-correlation. 2011.

[17] Takeo Kanade and M. Okutomi. Stereo matching algorithm with an adaptive window: Theory and experiment. 1994.

[18] 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 *CVPR*, pages 66–75, 2017.

[19] Sameh Khamis, Sean Fanello, Christoph Rhemann, Adarsh Kowdle, Julien Valentin, and Shahram Izadi. Stereonet: Guided hierarchical refinement for real-time edge-aware depth prediction. In *ECCV*, pages 573–590, 2018.

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[21] Jiankun Li, Peisen Wang, Pengfei Xiong, Tao Cai, Ziwei Yan, Lei Yang, Jiangyu Liu, Haoqiang Fan, and Shuaicheng Liu. Practical stereo matching via cascaded recurrent network with adaptive correlation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16263–16272, 2022.

[22] Zhengfa Liang, Yiliu Feng, Yulan Guo, Hengzhu Liu, Wei Chen, Linbo Qiao, Li Zhou, and Jianfeng Zhang. Learning for disparity estimation through feature constancy. In *CVPR*, pages 2811–2820, 2018.

[23] Zhengfa Liang, Yulan Guo, Yiliu Feng, Wei Chen, Linbo Qiao, Li Zhou, Jianfeng Zhang, and Hengzhu Liu. Stereo matching using multi-level cost volume and multi-scale feature constancy. *IEEE transactions on pattern analysis and machine intelligence*, 43(1):300–315, 2019.

[24] Lahav Lipson, Zachary Teed, and Jia Deng. Raft-stereo: Multilevel recurrent field transforms for stereo matching. *arXiv preprint arXiv:2109.07547*, 2021.

[25] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the effective receptive field in deep convolutional neural networks. *Advances in neural information processing systems*, 29, 2016.

[26] 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 *CVPR*, pages 4040–4048, 2016.

[27] Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *CVPR*, pages 3061–3070, 2015.

[28] Jiahao Pang, Wenxiu Sun, Jimmy SJ Ren, Chengxi Yang, and Qiong Yan. Cascade residual learning: A two-stage convolutional neural network for stereo matching. In *CVPRW*, pages 887–895, 2017.

[29] Zhibo Rao, Mingyi He, Yuchao Dai, Zhidong Zhu, Bo Li, and Renjie He. Nlca-net: a non-local context attention network for stereo matching. *APSIPA Transactions on Signal and Information Processing*, 9, 2020.

[30] Daniel Scharstein, Heiko Hirschmüller, York Kitajima, Greg Krathwohl, Nera Nešić, Xi Wang, and Porter Westling. High-resolution stereo datasets with subpixel-accurate ground truth. In *German Conference on Pattern Recognition*, pages 31–42, 2014.

[31] Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47(1):7–42, 2002.

[32] Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47(1):7–42, 2002.

[33] Thomas Schops, Johannes L Schonberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *CVPR*, pages 3260–3269, 2017.

[34] Markus Schulze. A new monotonic, clone-independent, reversal symmetric, and condorcet-consistent single-winner election method. *Social choice and Welfare*, 36(2):267–303, 2011.

[35] Zhelun Shen, Yuchao Dai, and Zhibo Rao. Cfnet: Cascade and fused cost volume for robust stereo matching. In *CVPR*, pages 13906–13915, 2021.

[36] Xiao Song, Guorun Yang, Xinge Zhu, Hui Zhou, Zhe Wang, and Jianping Shi. Adastereo: a simple and efficient approach for adaptive stereo matching. In *CVPR*, pages 10328–10337, 2021.

[37] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8934–8943, 2018.

[38] Vladimir Tankovich, Christian Hane, Yinda Zhang, Adarsh Kowdle, Sean Fanello, and Sofien Bouaziz. Hitnet: Hierarchical iterative tile refinement network for real-time stereo matching. In *CVPR*, pages 14362–14372, 2021.

[39] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *ECCV*, pages 402–419, 2020.

[40] Jonathan Tremblay, Thang To, and Stan Birchfield. Falling things: A synthetic dataset for 3d object detection and pose estimation. In *CVPRW*, pages 2038–2041, 2018.

[41] Jialiang Wang, Varun Jampani, Deqing Sun, Charles Loop, Stan Birchfield, and Jan Kautz. Improving deep stereo network generalization with geometric priors. *arXiv preprint arXiv:2008.11098*, 2020.

[42] Bin Xu, Yuhua Xu, Xiaoli Yang, Wei Jia, and Yulan Guo. Bilateral grid learning for stereo matching networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12497–12506, 2021.

[43] Gangwei Xu, Junda Cheng, Peng Guo, and Xin Yang. Attention concatenation volume for accurate and efficient stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12981–12990, 2022.

[44] Gangwei Xu, Yun Wang, Junda Cheng, Jinhui Tang, and Xin Yang. Accurate and efficient stereo matching via attention concatenation volume. *arXiv preprint arXiv:2209.12699*, 2022.

[45] Haofei Xu and Juyong Zhang. Aanet: Adaptive aggregation network for efficient stereo matching. In *CVPR*, pages 1959–1968, 2020.

[46] Gengshan Yang, Joshua Manela, Michael Happold, and Deva Ramanan. Hierarchical deep stereo matching on high-resolution images. In *CVPR*, pages 5515–5524, 2019.

[47] Chengtang Yao, Yunde Jia, Huijun Di, Pengxiang Li, and Yuwei Wu. A decomposition model for stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6091–6100, 2021.

[48] Jure Zbontar and Yann LeCun. Computing the stereo matching cost with a convolutional neural network. In *CVPR*, pages 1592–1599, 2015.

[49] Feihu Zhang, Victor Prisacariu, Ruigang Yang, and Philip HS Torr. Ga-net: Guided aggregation net for end-to-end stereo matching. In *CVPR*, pages 185–194, 2019.

[50] Feihu Zhang, Xiaojuan Qi, Ruigang Yang, Victor Prisacariu, Benjamin Wah, and Philip Torr. Domain-invariant stereo matching networks. In *European Conference on Computer Vision*, pages 420–439. Springer, 2020.