2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) | 979-8-3503-0129-8/23/$31.00 ©2023 IEEE | DOI: 10.1109/CVPR52729.2023.00134

High-frequency Stereo Matching Network

Haoliang Zhao[1,4†], Huizhou Zhou[2,4†], Yongjun Zhang[1*], Jie Chen[3], Yitong Yang[1] and Yong Zhao[3,4**]

[1]Text Computing & Cognitive Intelligence Engineering Research Center of National Education Ministry, State Key Laboratory of Public Big Data, College of Computer Science and Technology, Institute of Artificial Intelligence, Guizhou University, Guiyang 550025, Guizhou, China
[2]School of Physics and Optoelectronic Engineering, Guangdong University of Technology, Guangzhou 510006, China
[3]The Key Laboratory of Integrated Microsystems, Shenzhen Graduate School, Peking University, China
[4]Ghost-Valley AI Technology, Shenzhen, Guangdong, China

Abstract

In the field of binocular stereo matching, remarkable progress has been made by iterative methods like RAFT-Stereo and CREStereo. However, most of these methods lose information during the iterative process, making it difficult to generate more detailed difference maps that take full advantage of high-frequency information. We propose the Decouple module to alleviate the problem of data coupling and allow features containing subtle details to transfer across the iterations which proves to alleviate the problem significantly in the ablations. To further capture high-frequency details, we propose a Normalization Refinement module that unifies the disparities as a proportion of the disparities over the width of the image, which address the problem of module failure in cross-domain scenarios. Further, with the above improvements, the ResNet-like feature extractor that has not been changed for years becomes a bottleneck. Towards this end, we proposed a multi-scale and multi-stage feature extractor that introduces the channel-wise self-attention mechanism which greatly addresses this bottleneck. Our method (DLNR) ranks 1st on the Middlebury leaderboard, significantly outperforming the next best method by 13.04%. Our method also achieves SOTA performance on the KITTI-2015 benchmark for D1-fg. Code and demos are available at: `https://github.com/David-Zhao-1997/High-frequency-Stereo-Matching-Network`.

1. Introduction

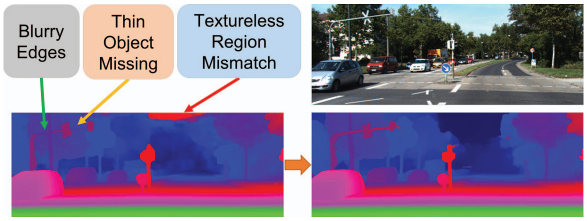

Figure 1. Motivation. We aim to address the problem of blurry edges, thin object missing, and textureless region mismatch.

Stereo depth estimation is becoming the infrastructure for 3D applications. Accurate depth perception is vital for autonomous driving, drones navigation, robotics and other related fields. The main point of the task is to estimate a pixel-wise displacement map also known as disparity that can be used to determine the depth of the pixels in the scene. Traditional stereo matching algorithms [7,11,12] are mainly divided into two types: global methods [6, 16, 17, 26] and local methods [1, 13]. Both methods solve the optimization problem by minimizing the objective function containing the data and smoothing terms, while the former takes into account the global information, the latter simply takes into account the local information, hence both have their own benefits in terms of accuracy and speed when solving the optimization problem. Traditional methods have excellent generalization performance and robustness in different scenarios, but perform poorly on details such as weak

[†] These authors contributed equally.
[*] Corresponding author.
Email: zyj6667@126.com.
[**] Second Corresponding author.
Email: zhaoyong@pkusz.edu.cn

979-8-3503-0129-8/23/$31.00 ©2023 IEEE
DOI 10.1109/CVPR52729.2023.00134

textures and repetitive texture regions. With the development of convolutional neural networks, learning-based approaches [20, 28, 37] have lately demonstrated promising result in tackling the matching problem of challenging regions. Take advantages of the strong regularization performance of the 3D convolution and 4D cost volume, methods [2, 10, 15, 45] using 3D convolution performs well. While their practical applicability is limited by the high computational cost. Subsequent methods [37,43] attempt to use multiple adaptive aggregated and guided aggregated 2D convolutions instead of 3D convolution, reducing computational cost and achieving better performance. The recent appearance of RAFT-Stereo [20] has given rise to a fresh concept for the research of stereo matching. Derived from the optical estimation method RAFT [29], RAFT-Stereo uses the iterative refinement method for a coarse-to-fine pipeline. It first calculates the correlation between all pixel pairs to construct a 3D correlation pyramid. Then an update operator with a convolutional GRU as the core unit is used to retrieves features from the correlation pyramid and updates the disparity map [20].

Despite great progress has been made in learning-based approaches, two major problems remain. (1) Most current approaches fall short when it comes to the finer features of the estimated disparity map. Especially for the edge performance of the objects. In bokeh and rendering applications, the edge performance of the disparity map is critical to the final result. For example, technologies that require pixel-level rendering, such as VR and AR, have high requirements for fitting between the scene model and the image mapping, which means we need a tight fit between the edges in the disparity map and the original RGB image. (2) The mismatch of textureless regions and the missing of thin objects are also important factors that significantly deteriorate the disparity map. For example, the mismatch of weak texture walls and the missing of thin electrical wires are fatal flaws for obstacle avoidance applications.

To alleviate these problems, we propose DLNR (Stereo Matching Network with Decouple LSTM and Normalization Refinement), a new end-to-end data-driven method for stereo matching.

We introduced several improvements based on the iterative model:

Most of the current iterative methods usually apply the original GRU structure as their iterative cell. While the problem is that in the original GRU structure, the information used to generate the update matrix of the disparity map is coupled with the value of the hidden state transfer between iterations, making it hard to keep subtle details in the hidden state. Therefore, we designed the Decouple LSTM module to decouple the hidden state from the update matrix of the disparity map. Experiments and visualizations proved that the module retains more subtle details in the hidden states.

Decouple LSTM keeps more high-frequency information in the iterative stage through data decoupling, however, in order to balance performance and computational speed, the resolution of the iterative stage is only 1/4 of the original resolution at most. To produce disparity maps with sharp edges and subtle details, a subsequent refinement module is still needed. In our refinement module, we aim to sufficiently exploit the information from the upsampled disparity maps, the original left and right images containing high-frequency information to enhance edges and details. However, due to the large differences in disparity ranges between different images and different datasets, the Refinement module often has poor generalization performance when encountering images with different disparity ranges. In particular, when performing finetune, the module may even fail when encountering disparity ranges that differ greatly. To address this problem, we propose the Disparity Normalization strategy. Experiments and visualizations proved that the module improves performance as well as alleviates the problem of domain difference.

After the above two improvements, we found that the feature extractor became the bottleneck of the performance. In the field of stereo matching, feature extraction has not been improved significantly for years, most learning-based methods still use ResNet-like feature extractors which fall short when providing information for well-designed post-stage structures. To alleviate the problem, we propose the Channel-Attention Transformer feature extractor aims to capture long-range pixel dependencies and preserve high-frequency information.

2. Related Works

2.1. Learning-based Approaches

In recent years, learning-based methods show great improvements in the field of stereo matching. PSMNet [2] solves the regularization problem by using 3D convolutions and proved to be effective, which milestone in the field of stereo matching, and the later CFNet [24] has improved on its basis with a cascade and fused cost volume. AANet [37] uses an adaptive cost aggregation approach instead of the 3D convolution, which achieves high efficiency and alleviates the edge-fattening issue of the previous method. Instead of constructing the cost volume explicitly, HITNet [28] relies on a fast multi-resolution initialization step, differentiable 2D geometric propagation, and warping mechanisms to infer disparity assumptions. The recently emerged RAFT-Stereo [20] exploit the idea of using iterative update for regularization, leading to competitive results.

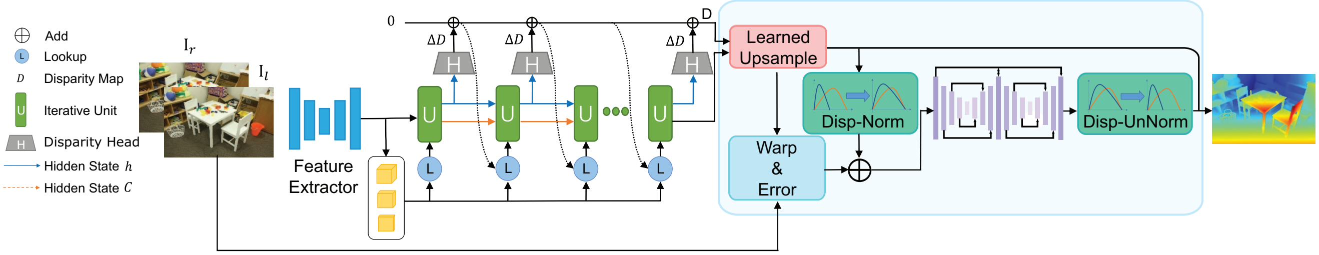

Figure 2. Overall structure of our method. The rectified image pairs are passed in to the Channel-Attention Transformer feature extractor which is capable of long-range pixel modeling and the features are processed by the subsequent Multiscale Decouple LSTM Network which could carry more semantic information across iterations. The refinement module upsamples the disparity map and perform Disparity Normalization before it is processed by the feed-forward structure which could alleviate the problem of domain gap.

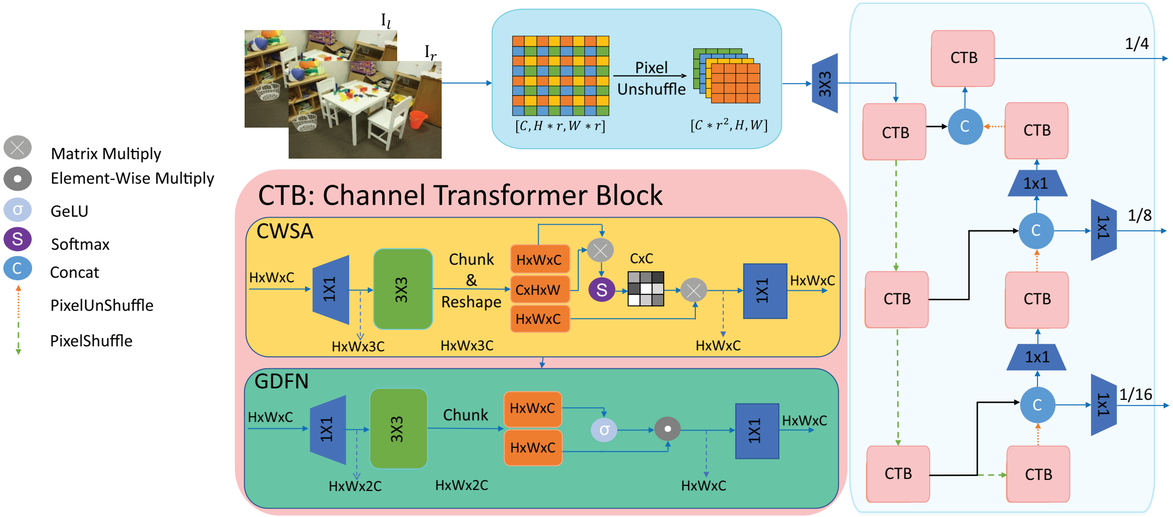

Figure 3. Channel-Attention Transformer extractor. We use a series of CTBs to form a U-shape structure, which output multi-scale and multi-stage features. Pixel Unshuffle is used for downsampling without any losing high-frequency details. CWSA denotes the Channel-wise self-attention and GDFN [42] denotes Gated-Dconv feed-forward network.

2.2. Iterative Approaches

Recently, iterative method gain its popularity in 3D tasks such as optic flow, stereo, structure from motion and MVS, etc. Most of them show notable advantages compared to other approaches. Zachary Teed first proposed the iterative model RAFT [29], which first constructs 4D cost volumes by calculating the relationship between all pixel pairs, and then retrieves the iterated features from the correlation volume using the GRU-based update operator for updating the optical flow field. The core of both stereo matching and optical flow estimation tasks is to find the offsets between pixel pairs. Therefore Lahav Lipson migrates the framework of RAFT to stereo matching [20] and proposes a streamlined and reasonable 3D correlation volume based on epipolar constraint. In addition, RAFT-Stereo [20] incorporates a multiscale information processing mechanism for the update operator. Wang introduces the iterative structure into the Multi-View Stereo task [31], and uses the GRU update operator as a probability estimator for each pixel at each depth hypothesis plane. Although iterative structures have made breakthroughs in related fields, the current iterative units are too simple, which limits the accuracy of iterative updates.

3. Approach

To tackle the problem, we designed an end-to-end network consisting of three sequentially arranged modules as shown in Figure 2. The Channel-Attention Transformer extractor takes a pair of rectified images I_l and I_r as input. Through multi-scale and multi-stage process, features F_l, F_m and F_h are pass into the correlation volume and the proposed Decouple LSTM. By the combining use of sampling from the correlation volume and the current state, the Decouple LSTM overlays the disparity map iterative and finally output a 1/4 resolution disparity map and an upsampling mask. The Normalization Refinement module take the above output and finally generate the final disparity map $D_{refined}$.

3.1. Channel-Attention Transformer extractor

In the field of binocular stereo matching, feature extraction has not improved significantly for years, and most learning-based methods still use ResNet-like feature extractors. These types of feature extractors have become bottlenecks in the network when providing information for well-designed post-stage structures.

In recent years, Transformer and self-attention have proved to be effective in many vision tasks for its long-range pixel dependencies modeling ability. While its computational cost grows quadratically with the image resolution. Inspired by Restormer [42], we designed a multi-stage and multi-scale Channel-Attention Transformer as the feature extractor. Detailed structure can be seen in Figure 3.

We aim to design a feature extractor that not only capture long-range pixel dependencies by also preserve as much high-frequency information as possible.

3.1.1 Preserving high-frequency information

To achieve the goal of sharp edges and better deal with weak texture regions, maintaining high-frequency during the pipeline is of vital importance. The most intuitive way is to maintain high-resolution throughout the structure while it lead to extremely high computational cost. While downsampling by convolution with stride or using pooling mechanism will inevitably result in information loss and performance degradation. To alleviate the problem, Pixel Unshuffle is applied to downsample the image to 1/4 the original size and expand the channels without losing any high-frequency information. Specifically, the shape of the original image is $[C, H * r, W * r]$, which is reshaped to $[C * r^2, H, W]$ after Pixel Unshuffle.

3.1.2 Channel Attention Mechanism

Conventional self-attention manages an attention map of $HW \times HW$, which lead to quadratic complexity, making it impractical for vision tasks that requires high resolution. Therefore, we adopt the CWSA module that derived from MDTA [42] module first proposed by Restromer [42] which computes self-attention on the channel dimension with linear complexity.

3.2. Multiscale Decouple LSTM Regularization

Our methods perform regularization using iterative unit. Through each iteration, the iterative unit predict a new update matrix of the disparity map ΔD_i combining the multiscale and multi-stage information F_l, F_m and F_h from the feature extractors, the hidden state generate by the last iteration h^{i-1}, C^{i-1} and the previous disparity map D_{i-1}. The unit is designed with the intention of using feature information as efficiently as possible and transferring valid information efficiently between iterations.

3.2.1 Multiscale Design

In the stereo matching task, it is difficult to find corresponding pixels in texture-less regions due to their weak pattern. Therefore, capturing the pattern of spatially adjacent pixels becomes a critical part of the problem. We handle this problem by multiscale design of our iterative module. Specifically, the iterative module is composed of three submodules of different scales, which is 1/4, 1/8 and 1/16 the size of the image resolution. Each of which interact with its neighboring resolutions. The low-resolution branch has a larger equivalent perceptual field which better deals with the texture-less regions while the high-resolution branch captures more high-frequency details which add more details to the edges and corners of the disparity map.

3.2.2 Decouple Mechanism

In the original GRU [4] structure used by most iterative vision networks, the hidden state h is used to generate the update matrix of disparities (output of the GRU Cell), while the h is also the hidden state of the GRU network (which transfer information to the next iteration). This coupling issue is proved to have a significant impact on the network performance in our ablation experiments.

We address this problem by introducing an extra hidden state C. As shown in Figure 4, the above-mentioned hidden state h is used for generating update matrix through the disparity head while the newly introduced hidden state C is used only for transferring information across iterations. The design decouples the update matrix and the hidden state, which can retain more effective semantic information across iterations. Ablation studies show the effectiveness of the method. More details can be seen in Table 3 and Figure 8.

3.3. Disparity Normalization Refinement

Since our model perform regularization on downsampled resolutions, high-frequency information can not be fully preserved in the process. Towards this end, we design a refinement module intended to capture more subtle details at full resolution.

We have observed that in the relatively independent modules in finetuning, the output of the feature maps may be all negative due to domain differences, and after the ReLU activation function, the feature maps all get 0 values, thus causing the network to be unable to finetune this part of the parameters, and can only pass the feature information to the subsequent modules through the skip connection. This leads to the problem that after the network is pre-trained, some modules cannot be finetune well and even encounter

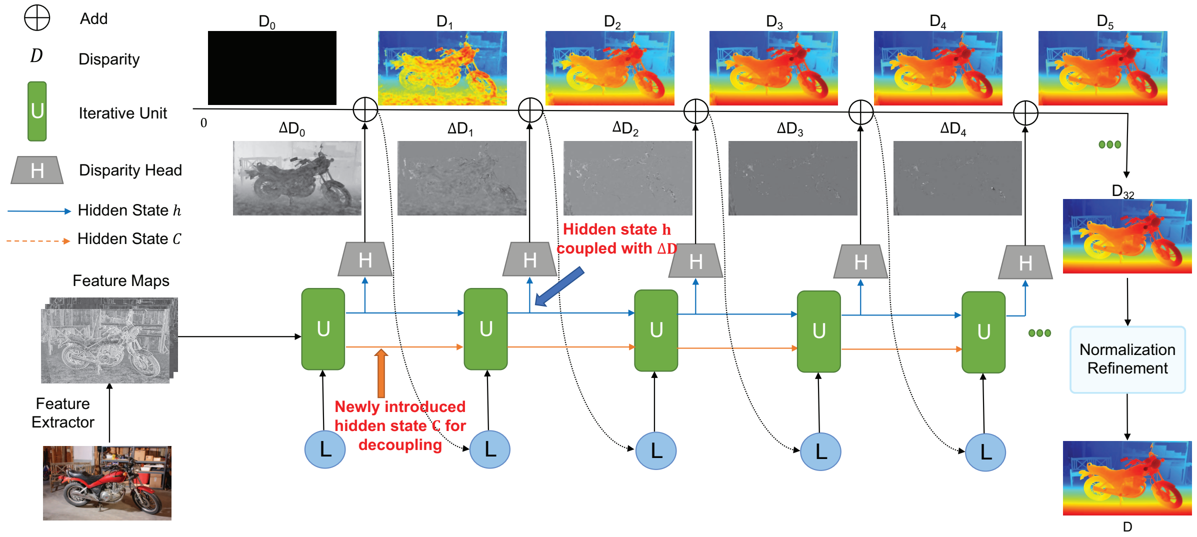

Figure 4. Decouple LSTM. The Iterative unit take the hidden states and the information from the cost volume as input and output the update matrix (ΔD_i in the figure) of the disparity map which adds to the disparity map (D_i in the figure). The update matrix ΔD_i gradually approaches to 0 since the disparity map is more and more refined. L denotes the Lookup [29] operator.

module failure when finetune is performed in other data sets.

As shown in Figure 5, the 1/4 resolution disparity map is first upsampled by learned upsampling. Then warping function is used to convert the right image to the left and calculate an error map.

$$D^{fr} = learnedUpsample(D^{lr}, upMask) \tag{1}$$

$$I_l^{'} = warp(I_r, disp) \tag{2}$$

$$E_l = I_l^{'} - I_l \tag{3}$$

where D^{fr} denotes the disparity map of the full resolution, D^{lr} denotes the disparity map before upsampling.

The upsampled disparities are scaled between 0 and 1. Note that the $min(D^{fr})$ typically equals 0. We use the width of the left image as denominator which is the max possible disparity value.

$$D^{fr}_{Norm} = \frac{D^{fr} - min(D^{fr})}{width(I_l)} \tag{4}$$

Then the information in the normalized disparity map D^{fr}_{Norm}, the error map E_l and the left image I_l are combined and process by the hourglass network and produce a normalized refined disparity map $D^{fr'}$.

$$I_{err} = Conv_{3\times3}([E_l, I_l]) \tag{5}$$

$$D^{fr'} = hourglass([I_{err}, Conv_{3\times3}(D^{fr}_{Norm})]) \tag{6}$$

Finally, disparity unnormalization is performed to generate the final disparity map.

$$D_{refined} = D^{fr'} \times width(I_l) + min(D^{fr'}) \tag{7}$$

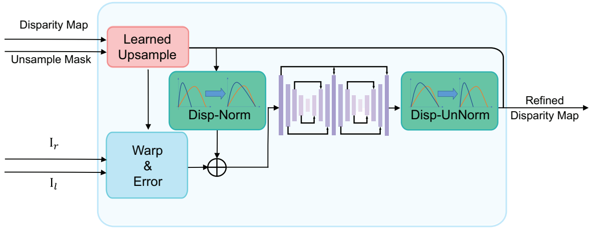

Figure 5. Disparity Normalization Refinement

3.4. Loss Function

We supervised our network by the following equations:

$$L = \sum_{i=1}^{n-1} \gamma^{n-i} L_1 + L_{refine}, where\, \gamma = 0.9. \tag{8}$$

$$L_1 = ||d_{gt} - d_i||_1 \tag{9}$$

$$L_{refine} = ||d_{gt} - d_{refined}||_1 \tag{10}$$

4. Experiments

We implement our DLNR in PyTorch and using AdamW as optimizer. For pretrain and ablations, we trained our model on the augmented Scene Flow training set (both cleanpass and finalpass) for 200k iterations with a batch size of 8. The learning rate uses a OneCycle scheduler with warm up strategy. The learning rate grows to $2e^{-4}$ in the first 2k iterations and gradually decreases to 0 thereafter. Data augmentation is used including saturation change, image perturbance, and random scales. The pretraining process takes roughly 2 days on our server equipped with 2 NVIDIA Tesla A100 GPUs.

We evaluate our model on the Scene Flow [21] dataset and two public benchmarks: Middlebury V3 [23] and KITTI-2015 [22].

4.1. Middlebury

DLNR ranks 1^{st} on the Middlebury V3 leaderboard, outperforming the next best method by 13.04%.

We pretrained our model on the Scene Flow dataset and then finetune on the Middlebury V3 training set, 10 evaluation training sets and 13 additional datasets are also included for training. We finetune our model for 4k iterations with a batch size of 2 and a linear decay learning rate decreasing from 2e-5 to 0. The image resolution is set as 384 × 1024 using random crop. Data augmentation methods are set as the same as pretraining without any additional settings.

Our method distinguishes subtle details and sharp edges of thin structures such as the leaves and branches of the plants in the Living Room and the overlap regions of the lines on the map and the long thin structures of the metal craft. Our method is also robust on the weak texture regions and occluded regions such as the staircase. Details are shown in Table 1. Visual comparisons are shown in Figure 6.

4.2. KITTI-2015

DLNR achieves SOTA performance on the KITTI-2015 D1-fg metric among all published methods at the time of writing this paper.

We pretrained our model on the Scene Flow datasets and fine-tuned our model on the KITTI training set for 6k iterations with a fixed learning rate of 0.00002 and a batch size of 8. The image resolution is set as 320 × 1024 using random crop. Data augmentation methods are set as the same as pretraining. Evaluation details are shown in Table 2. Visual comparisons are shown in Figure 7.

4.3. Ablations

To verify and better understand the structure of our model, extensive ablation experiments were conducted. For Scene Flow dataset, all hyperparameters settings are identical to the pretraining. And for KITTI dataset, all of finetune strategies are the same as the above mentioned KITTI benchmark. Details are shown in Table 3.

4.3.1 Decouple LSTM

As shown in Table 3, the use of Decouple LSTM significantly decreases the Scene Flow D1-error by 9.73% (from 5.96 to 5.38). To better understand the effect of the Decouple LSTM, we visualize the hidden state h and the newly introduced C of the Decouple LSTM. Details are shown in Figure 8. From the visualization, we can conclude that as motioned in the approach section, the hidden state C retains more features of the edges and more features of the thin objects, resulting in a better detail of the disparity map.

Table 1. Results on the Middlebury stereo dataset V3 [23] leaderboard. The best results for each metric are bolded, second best are underlined. For all metrics, lower is better.

	bad 0.5	bad 1.0	bad 2.0	bad 4.0	avgerr
	nonocc	nonocc	nonocc	nonocc	nonocc
	(%)	(%)	(%)	(%)	(%)
LocalExp [27]	38.7	13.9	5.43	3.69	2.24
NOSS-ROB [14]	38.2	13.2	5.01	3.46	2.08
HITNet [28]	34.2	13.3	6.46	3.81	1.71
RAFT-Stereo [20]	27.2	9.37	4.74	2.75	1.27
CREStereo [19]	28.0	8.25	3.71	2.04	1.15
EAI-Stereo	25.1	7.81	3.68	2.14	1.09
DLNR (Ours)	**23.9**	**6.82**	**3.20**	**1.89**	**1.06**

Table 2. Results on the KITTI-2015 [22] leaderboard. The best results for each metric are bolded, second best are underlined.

Method	D1-all	D1-fg	D1-bg
AcfNet [45]	1.89	3.80	1.51
AMNet [5]	1.82	3.43	1.53
OptStereo [33]	1.82	3.43	1.50
GANet-deep [43]	1.81	3.46	1.48
RAFT-Stereo [20]	1.96	2.89	1.75
HITNet [28]	1.98	3.20	1.74
CFNet [24]	1.88	3.56	1.54
PCWNet [25]	1.67	3.16	**1.37**
ACVNet [36]	**1.65**	3.07	**1.37**
DLNR (Ours)	1.76	**2.59**	1.60

The Decouple LSTM also shows strong cross-domain performance. We pretrain our model on the Scene Flow dataset, and then finetune the model on the KITTI-2015 training set. After finetune, we test our model on the Scene Flow validation set to evaluate the cross-domain performance. Experiments show that the D1-error of model using GRU is 16.38, while the model using Decouple LSTM is only 12.75, decreasing the error by 22.16%. Detailed results are shown in Table 3.

4.3.2 Normalization Refinement

As shown in Table 3, Disparity Refinement further improves the accuracy. This model increases the generalization ability as well. For the Scene Flow epe using KITTI weights, adding Normalization Refinement decreases the error by 11.22% (from 1.96 to 1.74). As shown in Figure 6, the module introduces high-frequency information of the original image, and produces disparity map with subtle details.

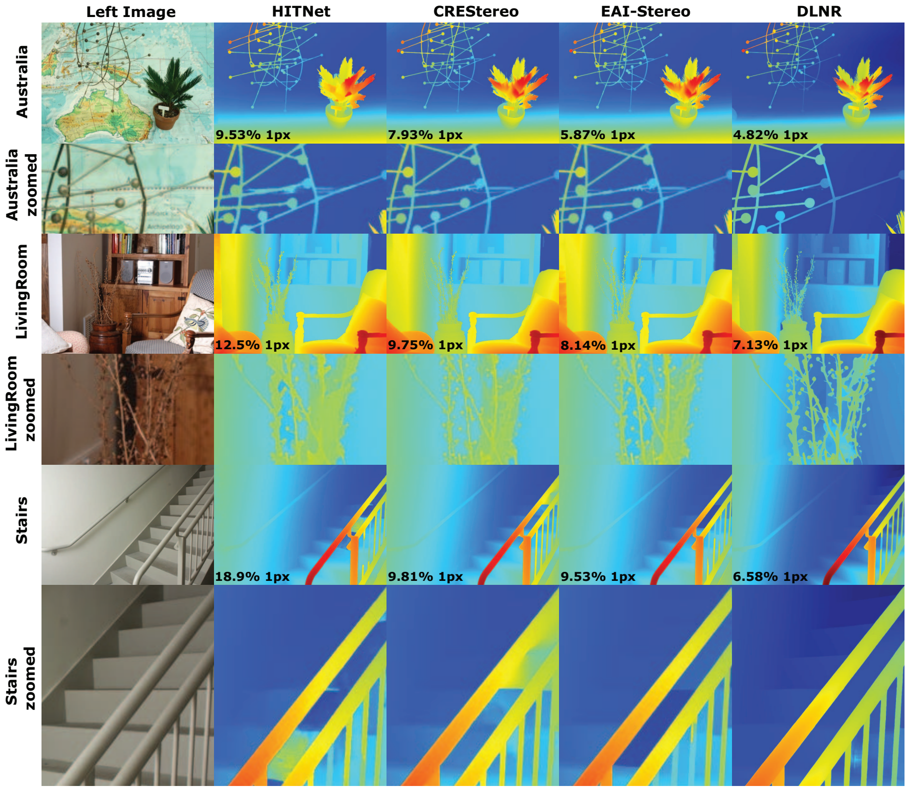

Figure 6. Comparisons on the Middlebury dataset. Our method distinguishes subtle details and sharp edges of thin structures such as the leaves and branches of the plants in the Living Room and the overlap regions of the lines on the map and the long thin structures of the metal craft. Our method is also robust on the weak texture regions and occluded regions such as the staircase. 1px error of each image is marked at the corner. Zoom in for a better view.

Table 3. Ablations on different structures of our proposed model.

Channel-Attention Transformer Extractor	Disparity Normalization Refinement	Decouple LSTM	Sceneflow epe	Sceneflow D1-error	KITTI epe	KITTI D1-error	Scene Flow D1-error (KITTI weights)	Scene Flow epe (KITTI weights)
			-	6.54	0.491	1.290	27.70	2.37
		✓	-	5.87	0.468	1.108	18.60	1.82
	✓	✓	-	5.74	0.401	0.854	17.99	1.87
✓			0.520	5.91	0.354	0.637	14.48	1.97
✓	✓		0.534	5.96	0.344	0.626	16.38	1.95
✓		✓	0.481	5.51	0.356	0.655	14.31	1.96
✓	✓	✓	0.477	5.38	0.335	0.561	12.75	1.74

4.3.3 Channel-Attention Transformer extractor

Channel-Attention Transformer extractor alleviates bottlenecks and shows great improvements in the ablations. By only applying the module on our baseline, the D1-error of the Scene Flow dataset decreased by 9.63% and the D1 error of the KITTI dataset decreased by 50.6%. Compared to the model using ResNet-like extractor, our final model decreased the error by significant 34.3% (from 0.854 to 0.561) in D1-error on the KITTI dataset. Details are shown in Table 3.

4.4. Performance and Inference Speed

In real-world applications, it is important to achieve a balance between performance and inference speed. We have conducted relevant experiments, the results of which are shown in Table 4.

4.5. Evaluation on Multi-View Stereo

We migrate our core modules to Multi-View Stereo task, and demonstrate that the Decouple LSTM and Normalization Refinement can achieve an excellent balance between the efficiency and reconstruct quality. we have achieved

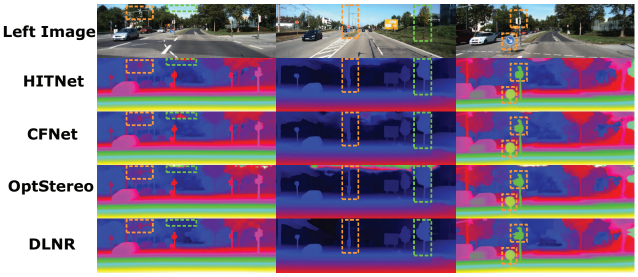

Figure 7. Comparisons on the KITTI dataset. Our method is robust on the weak texture regions and produces disparity maps with sharp edges.

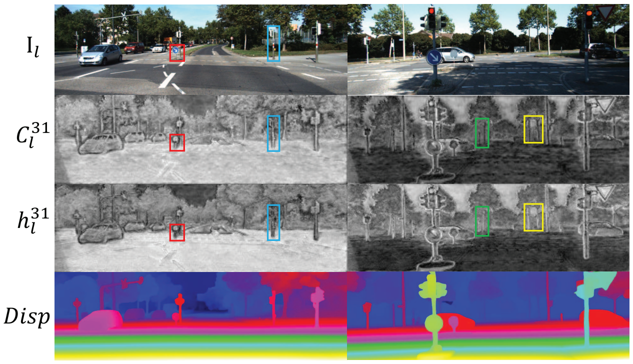

Figure 8. Visualization of the hidden state h and the newly introduced hidden state C. Specifically, we use PCA to reduce the number of channels from 128 to 1. The hidden state C retains more features of the edges (see the red box) and more features of the thin objects (see the blue and yellow boxes). Zoom in for better view.

Table 4. Performance and inference speed with different iterations.

Iterations	KITTI D1 error	KITTI Time (ms)	Scene Flow epe	Scene Flow D1 error	Scene Flow Time (ms)
5	0.637	91	0.593	6.91	99
7	0.582	110	0.538	6.24	112
10	0.561	131	0.502	5.77	135
16	0.554	171	0.483	5.49	180
22	0.557	214	0.478	5.44	224
32	0.561	285	0.477	5.38	297

promising results in both the challenging DTU and Tanks & Temples benchmarks. The results are shown in Table 5 and Table 6 respectively. The depth map is compared in Figure 9.

We use IterMVS [31] as our baseline, which is also an iterative method. We integrate the Multi-scale design and Decouple LSTM into our method, which is our light version. In addition, we add Normalization Refinement to the lightweight version as a full version.

Table 5. Quantitative results of reconstruction quality on the DTU evaluation dataset (↓). A and B are the conventional methods and high-accuracy learning-based methods, respectively. C and D are high-efficiency learning-based methods.

	Method	Acc.	Comp.	overall
A	Tola [30]	0.342	1.190	0.766
	Gipuma [8]	**0.283**	0.873	0.578
B	CasMVSNet [9]	0.325	0.385	0.355
	D^2HC-RMVSNet [39]	0.395	0.378	0.386
	CVP-MVSNet	0.296	0.406	0.351
	AA-RMVSNet [34]	0.376	0.339	0.357
	Vis-MVSNet [44]	0.369	0.361	0.365
C	Fast-MVSNet [41]	0.336	0.403	0.370
	PatchMatchNet [32]	0.427	**0.277**	0.352
	IterMVS [31]	0.373	0.354	0.363
D	IterMVS+MS+DL (ours)	0.372	0.345	0.358
	IterMVS+MS+DL+NR (ours)	0.360	0.328	**0.344**

Table 6. Quantitative results of different methods on the Tanks & Temples benchmark . "Mean" refers to the mean F-score of all scenes (↑).

F-score	Intermediate Dataset								
	Fam.	Franc.	Horse	Light	M60	Pan.	Play.	Train	Mean
OpenMVS	71.69	51.12	42.76	58.98	54.72	56.17	59.77	45.69	55.11
CIDER [38]	56.79	32.39	29.89	54.67	53.46	53.51	50.48	42.85	46.76
CasMVSNet [9]	76.37	58.45	46.26	55.81	56.11	54.06	58.18	49.51	56.84
UCS-Net [3]	76.09	53.16	43.03	54.00	55.60	51.49	57.38	47.89	54.83
CVP-MVSNet [40]	76.50	47.74	36.34	55.12	57.28	54.28	57.43	47.54	54.03
D2HC-RMVSNet [39]	74.69	56.04	49.42	60.08	59.81	59.61	60.04	53.92	59.20
Fast-MVSNet [41]	65.18	39.59	34.98	47.81	49.16	46.20	53.27	42.91	47.39
PatchMatchNet [32]	66.99	52.64	43.24	54.87	52.87	49.54	54.21	50.81	53.15
MVSTR [46]	76.92	59.82	50.16	56.73	56.53	51.22	56.58	47.48	56.93
PatchMatch-RL [18]	60.37	43.26	36.43	56.27	57.30	53.43	59.85	47.61	51.81
RayMVSNet [35]	78.56	61.96	45.48	57.58	61.01	59.76	59.20	52.32	59.49
IterMVS [31]	76.12	55.80	50.53	56.05	57.68	52.62	55.70	50.99	56.94
IterMVS+MS+DL (ours)	76.07	55.09	51.81	56.10	60.23	56.27	54.33	53.35	57.91
IterMVS+MS+DL+NR (ours)	77.85	59.69	54.73	57.69	58.62	56.40	56.19	54.88	**59.51**

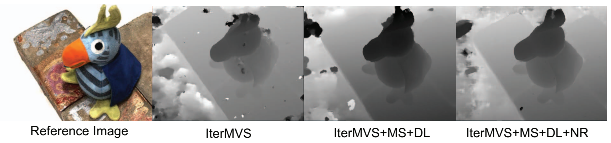

Figure 9. Depth estimation on the DTU dataset. Our method has a clear advantage, Multiscale (MS) and Decouple LSTM (DL) enhance the depth level, Normalization Refinement (NR) refines the edges. Zoom in for better view.

5. Conclusion

We have proposed DLNR, a new learning based method for Stereo Matching task. Decouple LSTM and Normalization Refinement are proposed to capture subtle details and produce disparity maps with sharp edges. Our method ranks first on the Middlebury leaderboard and achieves SOTA performance in foreground prediction on KITTI-2015.

Acknowledgements. This work was supported by Shenzhen Science and Technology Research and Development Fund (JCYJ20180503182133411, JSGG20201102153800002, KQTD20200820113105004).

References

[1] Michael Bleyer and Margrit Gelautz. Simple but effective tree structures for dynamic programming-based stereo matching. In *International Conference on Computer Vision Theory and Applications*, volume 2, pages 415–422. SCITEPRESS, 2008. 1

[2] Jia-Ren Chang and Yong-Sheng Chen. Pyramid stereo matching network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5410–5418, 2018. 2

[3] Shuo Cheng, Zexiang Xu, Shilin Zhu, Zhuwen Li, Li Erran Li, Ravi Ramamoorthi, and Hao Su. Deep stereo using adaptive thin volume representation with uncertainty awareness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2524–2534, 2020. 8

[4] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, Doha, Qatar, Oct. 2014. Association for Computational Linguistics. 4

[5] Xianzhi Du, Mostafa El-Khamy, and Jungwon Lee. Amnet: Deep atrous multiscale stereo disparity estimation networks. *arXiv preprint arXiv:1904.09099*, 2019. 6

[6] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient belief propagation for early vision. *International journal of computer vision*, 70(1):41–54, 2006. 1

[7] Wade S Fife and James K Archibald. Improved census transforms for resource-optimized stereo vision. *IEEE Transactions on Circuits and Systems for Video Technology*, 23(1):60–73, 2012. 1

[8] Silvano Galliani, Katrin Lasinger, and Konrad Schindler. Massively parallel multiview stereopsis by surface normal diffusion. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 873–881, 2015. 8

[9] 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. 8

[10] Xiaoyang Guo, Kai Yang, Wukui Yang, Xiaogang Wang, and Hongsheng Li. Group-wise correlation stereo network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3273–3282, 2019. 2

[11] Yong Seok Heo, Kyong Mu Lee, and Sang Uk Lee. Robust stereo matching using adaptive normalized cross-correlation. *IEEE Transactions on pattern analysis and machine intelligence*, 33(4):807–822, 2010. 1

[12] Heiko Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on pattern analysis and machine intelligence*, 30(2):328–341, 2007. 1

[13] Asmaa Hosni, Christoph Rhemann, Michael Bleyer, Carsten Rother, and Margrit Gelautz. Fast cost-volume filtering for visual correspondence and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(2):504–511, 2012. 1

[14] Penglei Ji, Jie Li, Hanchao Li, and Xinguo Liu. Superpixel alpha-expansion and normal adjustment for stereo matching. *Journal of Visual Communication and Image Representation*, 79:103238, 2021. 6

[15] 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 *Proceedings of the IEEE international conference on computer vision*, pages 66–75, 2017. 2

[16] Andreas Klaus, Mario Sormann, and Konrad Karner. Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. In *18th International Conference on Pattern Recognition (ICPR'06)*, volume 3, pages 15–18. IEEE, 2006. 1

[17] Vladimir Kolmogorov and Ramin Zabih. Computing visual correspondence with occlusions using graph cuts. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 2, pages 508–515. IEEE, 2001. 1

[18] Jae Yong Lee, Joseph DeGol, Chuhang Zou, and Derek Hoiem. Patchmatch-rl: Deep mvs with pixelwise depth, normal, and visibility. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6158–6167, 2021. 8

[19] 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 Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16263–16272, 2022. 6

[20] Lahav Lipson, Zachary Teed, and Jia Deng. Raft-stereo: Multilevel recurrent field transforms for stereo matching. In *2021 International Conference on 3D Vision (3DV)*, pages 218–227. IEEE, 2021. 2, 3, 6

[21] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4040–4048, 2016. 6

[22] Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3061–3070, 2015. 6

[23] 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 Xiaoyi Jiang, Joachim Hornegger, and Reinhard Koch, editors, *Pattern Recognition*, pages 31–42, Cham, 2014. Springer International Publishing. 6

[24] Zhelun Shen, Yuchao Dai, and Zhibo Rao. Cfnet: Cascade and fused cost volume for robust stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13906–13915, 2021. 2, 6

[25] Zhelun Shen, Yuchao Dai, Xibin Song, Zhibo Rao, Dingfu Zhou, and Liangjun Zhang. Pcw-net: Pyramid combination

and warping cost volume for stereo matching. In *European Conference on Computer Vision*, pages 280–297. Springer, 2022. 6

[26] Jian Sun, Nan-Ning Zheng, and Heung-Yeung Shum. Stereo matching using belief propagation. *IEEE Transactions on pattern analysis and machine intelligence*, 25(7):787–800, 2003. 1

[27] Tatsunori Taniai, Yasuyuki Matsushita, Yoichi Sato, and Takeshi Naemura. Continuous 3d label stereo matching using local expansion moves. *IEEE transactions on pattern analysis and machine intelligence*, 40(11):2725–2739, 2017. 6

[28] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14362–14372, 2021. 2, 6

[29] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pages 402–419. Springer, 2020. 2, 3, 5

[30] Engin Tola, Christoph Strecha, and Pascal Fua. Efficient large-scale multi-view stereo for ultra high-resolution image sets. *Machine Vision and Applications*, 23(5):903–920, 2012. 8

[31] Fangjinhua Wang, Silvano Galliani, Christoph Vogel, and Marc Pollefeys. Itermvs: Iterative probability estimation for efficient multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8606–8615, 2022. 3, 8

[32] Fangjinhua Wang, Silvano Galliani, Christoph Vogel, Pablo Speciale, and Marc Pollefeys. Patchmatchnet: Learned multi-view patchmatch stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14194–14203, 2021. 8

[33] Hengli Wang, Rui Fan, Peide Cai, and Ming Liu. Pvstereo: Pyramid voting module for end-to-end self-supervised stereo matching. *IEEE Robotics and Automation Letters*, 6(3):4353–4360, 2021. 6

[34] Zizhuang Wei, Qingtian Zhu, Chen Min, Yisong Chen, and Guoping Wang. Aa-rmvsnet: Adaptive aggregation recurrent multi-view stereo network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6187–6196, 2021. 8

[35] Junhua Xi, Yifei Shi, Yijie Wang, Yulan Guo, and Kai Xu. Raymvsnet: Learning ray-based 1d implicit fields for accurate multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8595–8605, 2022. 8

[36] 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. 6

[37] Haofei Xu and Juyong Zhang. Aanet: Adaptive aggregation network for efficient stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1959–1968, 2020. 2

[38] Qingshan Xu and Wenbing Tao. Learning inverse depth regression for multi-view stereo with correlation cost volume. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12508–12515, 2020. 8

[39] Jianfeng Yan, Zizhuang Wei, Hongwei Yi, Mingyu Ding, Runze Zhang, Yisong Chen, Guoping Wang, and Yu-Wing Tai. Dense hybrid recurrent multi-view stereo net with dynamic consistency checking. In *European conference on computer vision*, pages 674–689. Springer, 2020. 8

[40] Jiayu Yang, Wei Mao, Jose M Alvarez, and Miaomiao Liu. Cost volume pyramid based depth inference for multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4877–4886, 2020. 8

[41] Zehao Yu and Shenghua Gao. Fast-mvsnet: Sparse-to-dense multi-view stereo with learned propagation and gauss-newton refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1949–1958, 2020. 8

[42] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *CVPR*, 2022. 3, 4

[43] F. Zhang, V. Prisacariu, R. Yang, and P. S. Torr. Ga-net: Guided aggregation net for end-to-end stereo matching. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 185–194, Los Alamitos, CA, USA, jun 2019. IEEE Computer Society. 2, 6

[44] Jingyang Zhang, Shiwei Li, Zixin Luo, Tian Fang, and Yao Yao. Vis-mvsnet: Visibility-aware multi-view stereo network. *International Journal of Computer Vision*, pages 1–16, 2022. 8

[45] 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 *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12926–12934, 2020. 2, 6

[46] Jie Zhu, Bo Peng, Wanqing Li, Haifeng Shen, Zhe Zhang, and Jianjun Lei. Multi-view stereo with transformer. *arXiv preprint arXiv:2112.00336*, 2021. 8