MODE: Multi-view Omnidirectional Depth Estimation with 360° Cameras

Ming Li, Xueqian Jin, Xuejiao Hu, Jingzhao Dai, Sidan Du, and Yang Li

Nanjing University, Nanjing, China
{mingli,jcboxq,hxj,dg20230007}@smail.nju.edu.cn,
{coff128,yogo}@nju.edu.cn

Abstract. In this paper, we propose a two-stage omnidirectional depth estimation framework with multi-view 360° cameras. The framework first estimates the depth maps from different camera pairs via omnidirectional stereo matching and then fuses the depth maps to achieve robustness against mud spots, water drops on camera lenses, and glare caused by intense light. We adopt spherical feature learning to address the distortion of panoramas. In addition, a synthetic 360° dataset consisting of 12K road scene panoramas and 3K ground truth depth maps is presented to train and evaluate 360° depth estimation algorithms. Our dataset takes soiled camera lenses and glare into consideration, which is more consistent with the real-world environment. Experimental results show that the proposed framework generates reliable results in both synthetic and real-world environments, and it achieves state-of-the-art performance on different datasets. The code and data are available at https://github.com/nju-ee/MODE-2022

Keywords: Omnidirectional Depth Estimation, Stereo Matching, Spherical Feature Learning, 360°Cameras, Multi-view

1 Introduction

Image-based depth estimation is a long-lasting and fundamental task in computer vision. Recently, omnidirectional depth estimation has attracted attention in many applications such as autonomous driving and robot navigation for its efficient perception of the 360° environment. Many algorithms have been proposed to estimate 360° depth, including monocular [28,14] and binocular [29,18] methods. However, these existing methods either extract spherical features with conventional planar convolution [28,14,29] or do not simplify the spherical epipolar constraint [18]. Apart from this, the monocular and binocular methods cannot obtain reliable depth maps when 360° cameras installed on vehicles are soiled by mud spots, water drops or dazzled by intense light (see Fig. 8).

Ming Li and Xueqian Jin contributed equally to this work.
Corresponding authors: Sidan Du, Yang Li

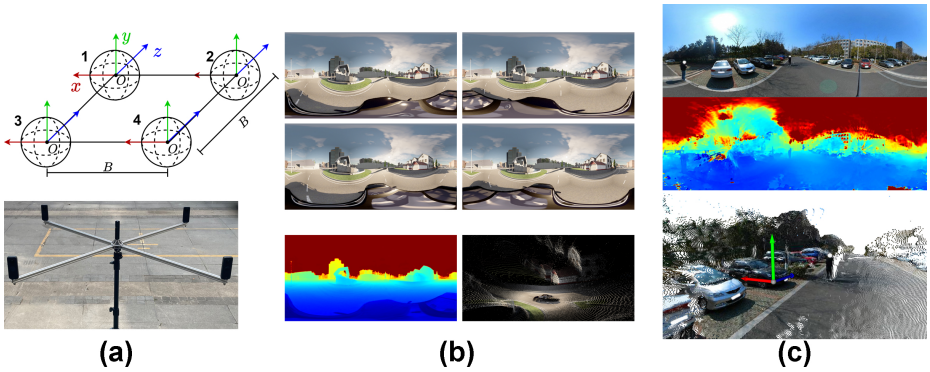

Fig. 1. Overview of the proposed multi-view omnidirectional depth estimation framework. (a) shows the multiple 360° camera rig. (b) shows the result of our method on the proposed synthetic dataset. The first two rows show the panoramas captured by the four cameras and the last row shows the predicted depth map and corresponding point cloud. (c) shows the results on the real-world environment, from top to bottom: reference panorama, predicted depth map and corresponding point cloud

Won et al. proposed multi-view methods SweepNet [30] and OmniMVS [31,32] to estimate 360° depth maps from four fish-eye cameras. However, these methods also use planar convolution to extract spherical features, and the blind areas of fish-eye cameras introduce discontinuity in the spherical cost volume.

In this paper, we decompose the multi-view omnidirectional depth estimation into two stages. In the first stage, we choose several camera pairs from different views for omnidirectional stereo matching and obtain disparity maps. In the second stage, we convert disparity maps to aligned depth maps and fuse them to estimate the final depth map. The information fusion of different stereo pairs improves the accuracy and robustness of the final depth map. In addition, the two parts of the framework can be trained and fine-tuned independently with lower hardware demands. We use Cassini projection [2] to simplify the epipolar constraint of omnidirectional stereo matching and propose a spherical feature extraction module to overcome the distortion of panoramas[1].

Moreover, a large-scale synthetic outdoor omnidirectional dataset, Deep360, is proposed in this work. To evaluate the performance of different 360° depth estimation methods when camera lenses are soiled by mud spots, water drops or dazzled by glare, we also provide a soiled version of the dataset.

Fig. 1 illustrates the overview of the proposed multi-view omnidirectional depth estimation (MODE) framework. It estimates accurate depth maps from four 360° cameras. Experimental results demonstrate that our method generates reliable depth maps in various scenes and achieves state-of-the-art (SOTA) performance on different datasets, especially the one with soiled panoramas. This

[1] We use the terms omnidirectional, 360°, spherical and panorama interchangeably in this document.

validates the robustness of our proposed framework and shows that the framework can be extended to arbitrary 360° multi-camera setups.

In summary, the main contributions of this work are as follows:

- We propose a flexible 360° depth estimation framework called MODE to obtain reliable depth maps against soiled camera lenses or glare. MODE also achieves SOTA performance.
- We introduce the spherical convolution to address panorama distortions in 360° stereo matching. We prove that using an appropriate projection to simplify the epipolar constraint is essential for this problem and introduce the Cassini projection. We adopt the training detail of removing image cropping for 360° stereo matching.
- We present a large-scale synthetic outdoor dataset, Deep360, that contains both high-quality and soiled panorama images.

2 Related Work

2.1 Deep Learning-based Stereo Matching Methods

Deep learning methods report much improved performance in stereo matching. Zbontar and Lecun [36] propose MCCNN that extracts features by CNNs and computes disparity via conventional matching cost aggregation. GCNet [15] builds cost volume with feature maps and obtains disparity maps through 3D CNN blocks. PSMNet [4] adopts spatial pyramid pooling in feature extraction and uses the stacked hourglass architecture in regression to improve the performance. GA-Net [37] proposes the local-guided and semi-global aggregation layers to capture local and whole-image dependencies respectively. AANet [33] adopts an adaptive aggregation algorithm and replaces the costly 3D-CNN for an efficient architecture. Lipson et al. [19] proposes RAFT-Stereo which adopts multi-level GRU modules to estiamte the accurate disparity maps. CFNet [25] proposes a network based on the cascade and fused cost volume to improve the robustness in stereo matching. DispNet [20] and CRL [22] compute left-right feature correlation and then estimate disparity maps.

Multi-view stereo (MVS) has developed rapidly in recent years as well. Yao et al [35]. proposed the end-to-end MVSNet that builds cost volume by warping feature maps of different views into front-parallel planes of the reference camera to obtain depth maps. Point-MVSNet [5] adopts the feature augmented point cloud to refine the depth map iteratively. Cascade-MVS [10] and CVP-MVS [34] improve the performance with multi-scale coarse-to-fine architectures.

These stereo matching methods are designed for perspective cameras with normal field-of-view (FoV) and do not consider the property of panoramas.

2.2 Omnidirectional Depth Estimation

Recently, some learning-based algorithms have been proposed for omnidirectional depth estimation. Zioulis et al. propose two monocular networks using

supervised learning [39], and adopt the extra coordinate feature in CoordNet [38] for learning context in the equirectangular projection (ERP) domain. Wang et al. [28] propose BiFuse for monocular depth estimation which combines the ERP and CubeMap projection to overcome the distortion of panoramas. Jiang et al. [14] develop BiFuse and propose UniFuse which achieves better performance via a more efficient fusion scheme. Cheng et al. [6] regard omnidirectional depth estimation as an extension of the partial depth map. Wang et al. [29] propose the 360SD-Net which estimates omnidirectional depth in the ERP domain for up-down stereo pairs. CSDNet [18] focuses on the left-right stereo and uses Mesh CNNs [13] to solve the spherical distortions. However, these methods either extract spherical features with planar convolution [39,28,14,29] or do not simplify the spherical epipolar constraint [18].

There are also some methods for obtaining omnidirectional depth maps based on multi-view fish-eye cameras. Won et al. propose SweepNet [30] which builds cost volume via spherical sweeping and estimates spherical depth by cost aggregation. They further improve the algorithm and propose the end-to-end OmniMVS [31,32] architecture to achieve better performance. However, these methods also use planar convolution to extract spherical features and the blind areas of fish-eye cameras introduce discontinuity in the spherical cost volume.

2.3 Omnidirectional Depth Datasets

Large-scale datasets with high variety are essential for training and evaluating learning-based algorithms. Recently released omnidirectional depth datasets can be divided into two categories according to the input images, one with the panoramas, and the other with the fish-eye images. These datasets are mainly collected from public available real-world and synthetic 3D datasets by repurposing them to omnidirectional by rendering.

For datasets with panoramas, Wang et al. [27] collect an indoor monocular 360° video dataset named PanoSUNCG from [26]. De La Garanderie et al. [16] provide an outdoor monocular 360° benchmark with 200 images generated from the CARLA autonomous driving simulator [8]. MP3D and SF3D [29] are indoor binocular 360° datasets collected from [3,1]. 3D60 by Zioulis et al. [38] is an indoor trinocular (central, right, up) 360° dataset collected from [3,1,26,11]. For datasets with fish-eye images, Won et al. [30,31,32] present three datasets: Urban, OmniHouse and OmniThings. All three datasets are virtually collected in Blender with four fish-eye cameras.

The fish-eye images need complementary information to estimate an omnidirectional depth map, which means discontinuity and requirements for camera directions. In contrast, the panoramas record all 360° information continuously without blind areas. However, as summarized above, the datasets with stereo panoramas [29,38] consist of indoor scenes only. A detailed summary of omnidirectional depth datasets can be found in Table 2.

3 Multi-view Omnidirectional Depth Estimation

3.1 Multi-view Omnidirectional Camera System

Camera System Settings In this paper, we use the camera rig shown in Fig. 1(a) to implement the proposed framework. Four 360° cameras are arranged on a horizontal plane to form a square with side length B. The cameras are numbered from 1 to 4. Any two of the cameras can form a stereo pair, so there are 6 (C_4^2) pairs in total. We mark the different stereo pairs with the numbers of the cameras, which are 1-2, 1-3, 1-4, 2-3, 2-4 and 3-4(i.e. 1-2 denotes the image pair of cameras 1 and 2).

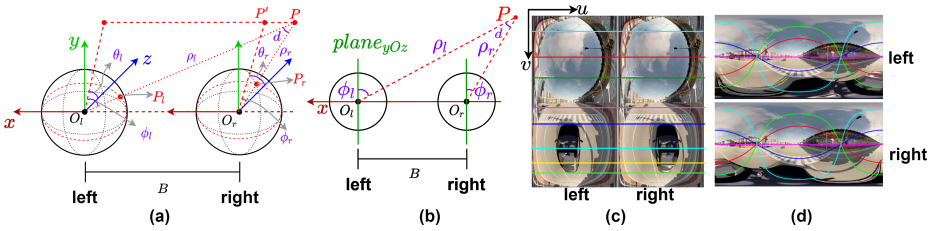

Fig. 2. Cassini projection and spherical epipolar geometry of the omnidirectional images. (a) illustrates the spherical coordinate system of omnidirectional left-right stereo cameras. (b) illustrates the angular disparity of spherical images. (c) shows the linear epipolar constraint in Cassini projection domain. The matching points are located in the same row of the left and right images. (d) shows the nonlinear epipolar constraint in ERP domain. The matching points are located on curves with the same color

Spherical Projection and Epipolar Constraint As shown in Fig. 2(a), we define the spherical coordinate system (ρ, ϕ, θ) as follows: ρ is the distance between the camera optic center O and the point P; ϕ is the angle between line OP and the plane yOz; and θ is the angle between line OP' and positive z, where P' is the projection of P on yOz. Thus, the transformation between Cartesian coordinates and the Cassini spherical coordinates is:

$$\begin{cases} x = \rho \sin(\phi) \\ y = \rho \cos(\phi) sin(\theta), \\ z = \rho \cos(\phi) cos(\theta) \end{cases} \qquad \begin{cases} \rho = \sqrt{(x^2 + y^2 + z^2)} \\ \phi = arcsin(\frac{x}{\rho}) \\ \theta = arctan(\frac{y}{z}) \end{cases} \tag{1}$$

where $\phi \in [-\frac{\pi}{2}, \frac{\pi}{2}]$, $\theta \in [-\pi, \pi]$.

As demostrated in [18], the spherical epipolar lines are projected to sinusoidal curves on the widely used ERP images (shown in Fig. 2(d)), which makes stereo

matching difficult in ERP domain. While in Cassini projection [2] domain, the epipolar lines are projected to horizontal lines with the mapping function:

$$\begin{cases} u = (\phi + \frac{\pi}{2}) \cdot \frac{W}{\pi} \\ v = (\theta + \pi) \cdot \frac{H}{2\pi} \end{cases} \tag{2}$$

where (u, v) denotes the image pixel coordinates in Cassini projection and H, W denote the height and width of the image (see Fig. 2(c)). We adopt the Cassini projection in this work to achieve the linear epipolar constraint for omnidirectional stereo matching.

Fig. 2(b) illustrates the angular disparity of the spherical stereo. Since the matching points have the same θ, the angular disparity d is defined as the difference of ϕ : $d = \phi_l - \phi_r$. The depth of P to the left camera is computed as:

$$\rho_l = B \cdot \frac{sin(\phi_r + \frac{\pi}{2})}{sin(d)} = B \cdot \left[\frac{sin(\phi_l + \frac{\pi}{2})}{tan(d)} - cos(\phi_l + \frac{\pi}{2}) \right]. \tag{3}$$

3.2 Network Architecture

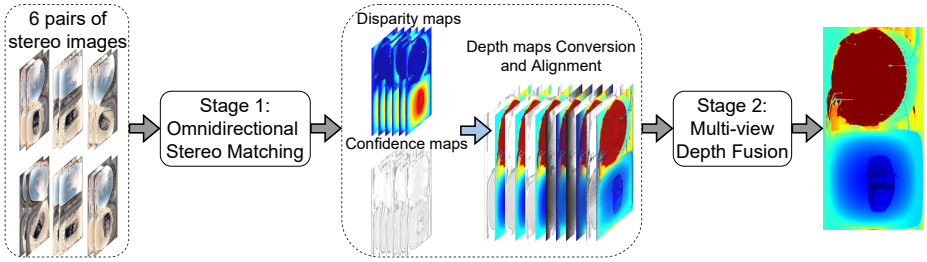

Fig. 3. The architecture of the proposed MODE

As shown in Fig. 3, the proposed MODE consists of two stages. In the first stage, six pairs of left-right panoramas are fed into the omnidirectional stereo matching network to estimate the disparity maps and confidence maps. The disparity maps from different stereo pairs are converted to depth maps and then aligned to the same viewpoint. In the second stage, we estimate the final depth map through a designed multi-view depth fusion network. The network details for the two stages are introduced in Sec. 3.3 and Sec. 3.4.

3.3 Omnidirectional Stereo Matching with Spherical Convolution

The rectified left-right panoramas follow the linear epipolar constraint in Cassini projection, but the distortion of panoramas still affects the results of stereo

matching. The regular convolution kernel suffers distortions of the 360° images near the poles. Therefore, we propose an omnidirectional stereo matching network with spherical convolution.

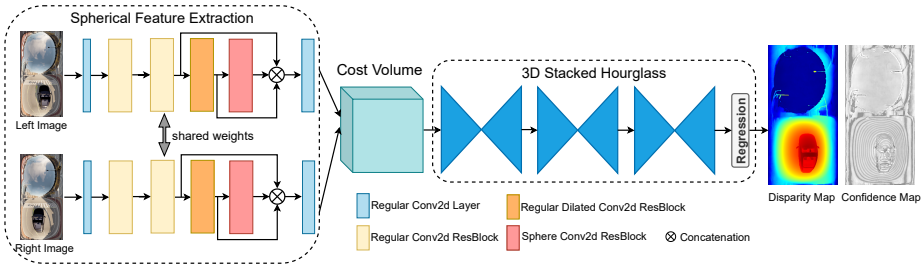

Fig. 4. The architecture of the proposed omnidirectional stereo matching network. We propose a spherical feature extraction module with spherical convolution to overcome the distortion in panoramas

As shown in Fig. 4, we first build a spherical feature extraction module with spherical convolution to overcome the distortions. We follow [7] to implement the spherical convolution operator and accelerate it with CUDA. The kernel sampling pattern of the spherical convolution and the comparison with the regular convolution are shown in Fig. 5.

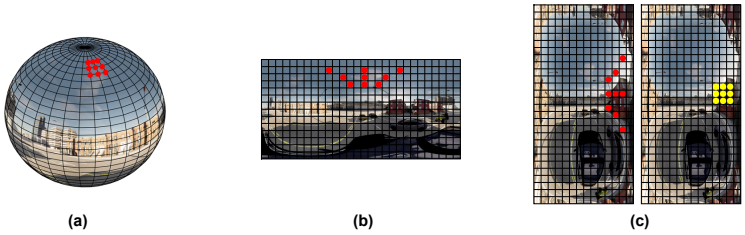

Fig. 5. The kernel sampling pattern of the sphere convolution in proposed omnidirectional stereo matching network. Red points in (a) and (b) show the sampling pattern on the sphere and the ERP image, respectively. (c) illustrates the comparison of the sphere convolution (red) and the regular convolution (yellow) on the Cassini projection image

The proposed spherical feature extraction module contains four sets of residual blocks [12]. The parameters are detailed in Table 1. The dilated convolution is applied in ResBlock3 for larger receptive fields. We apply spherical convolution in the last residual block to learn the high-level semantic and context features on spherical images. We concatenate the output of ResBlock2, ResBlock3 and Res-

Table 1. Parameters of the Spherical Feature Extraction Module

Name	Layer Settings	Output Dimension
Input	-	$H \times W \times 3$
Conv0	7×7, 32 3×3, 32 3×3, 32	$\frac{1}{2}H \times \frac{1}{2}W \times 3$
ResBlock1	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$	$\frac{1}{2}H \times \frac{1}{2}W \times 64$
ResBlock2	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 8$	$\frac{1}{4}H \times \frac{1}{4}W \times 64$
ResBlock3	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4, \textbf{dila} = 2$	$\frac{1}{4}H \times \frac{1}{4}W \times 64$
ResBlock4	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 8, \textbf{spherical}$	$\frac{1}{4}H \times \frac{1}{4}W \times 128$
Concatenation	[ResBlock2,ResBlock3,ResBlock4]	$\frac{1}{4}H \times \frac{1}{4}W \times 256$
featureFusion	1×1, 256 3×3, 128 1×1, 32	$\frac{1}{4}H \times \frac{1}{4}W \times 32$

Block4 and fuse these feature maps extracted by different kinds of convolutions through the feature fusion layers.

After spherical feature extraction, the feature maps of the stereo panoramas are shifted and concatenated to build the cost volume. Then, the omnidirectional disparity map is regressed through the 3D stacked hourglass as in [4]. The smooth L1 loss function is applied to train the network.

Moreover, many stereo matching algorithms [4,33] take a random crop of images as the network input. However, different crop areas on spherical projection images have different distributions in the high-level feature space due to the image distortions. Thus, we use the full omnidirectional images without cropping as the input of the proposed network to achieve better performance.

3.4 Multi-view Depth Map Fusion

The second stage of the proposed framework is depth map fusion with the incorporation of confidence maps and reference panoramas. The confidence map is used to estimate the reliability of disparity maps in many recent works. [23] reviews developments in the field of confidence estimation for stereo matching and evaluates existing confidence measures.

We first convert the omnidirectional disparity maps to depth maps according to Eq. 2 and Eq. 3. Then, all the depth maps are aligned to the same viewpoint based on the extrinsic matrix and visibility.

Considering that the stereo matching network computes each disparity value through a probability weighted sum over all disparity hypotheses, the probability distribution along the hypotheses thus reflects the quality of disparity estimation [35]. We compute the confidence for each inferred disparity value by taking a probability sum over the three nearest disparity hypotheses, which corresponds to the probability that the inferred disparity meets the 1-pixel error requirement. Then, we add the confidence map into the second stage of MODE to provide extra information for the depth map fusion since higher confidence implies higher fusion weight.

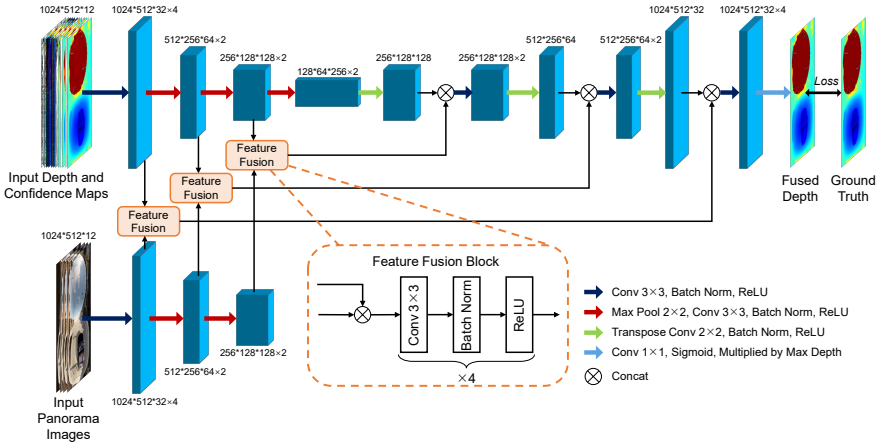

Fig. 6. The network architecture of multi-view depth map fusion. Input multi-view depth maps with confidence maps and reference panoramas are fed into two independent 2D encoder blocks. The final fused depth map output is regressed through one decoder block with skip connections between encoder and decoder blocks at each scale. '×n' denotes n times the block repeats

The architecture of the proposed depth map fusion network is illustrated in Fig. 6. In general, the network design follows the architecture of [24], which consists of an encoder-decoder path for global context extraction and skip connections between the two blocks for the transmission and localization of precise depth values. In addition to depth maps and corresponding confidence maps, we add reference panoramas to provide accurate boundary information for the fused depth map. To extract boundary features from panoramas independently, we separate the encoder block for panoramas from that for depth and confidence maps (shown at the bottom left of Fig. 6). Then, these two kinds of feature maps are fused through a designed feature fusion block at multi-scale to form a more informative feature map. The final fused depth is computed as

$$\hat{y}(\theta, \phi) = d_{max} \cdot \frac{1}{1 + e^{-\boldsymbol{D}(\theta,\phi)}}, \tag{4}$$

where d_{max} is the given maximum depth and $\boldsymbol{D}$ is the normalized depth map regressed by the decoder block.

For the loss function, we adopt the training loss developed from Scale-Invariant Error (SILog) [9] as

$$Loss(\hat{y}, y^{\star}) = \frac{1}{n}\sum_{i} d_i^2 - \frac{\lambda}{n^2}\left(\sum_{i} d_i\right)^2 \tag{5}$$

$$d_i = \log \hat{y}_i - \log y_i^{\star}, \tag{6}$$

where $\hat{y}$ is the predicted depth map, $y^\star$ is the ground truth and $\lambda \in [0, 1]$. We follow [9] to set $\lambda = 0.5$ in the experiments, which averages the scale-invariant depth error and absolute-scale error [9].

4 Datasets

As summarized in Sec. 2.3, although many datasets have been proposed for omnidirectional depth estimation, no 360° stereo dataset for outdoor road scenes is available due to the difficulty of acquiring 360° outdoor 3D datasets in the real world. Therefore, we create a public available 360° multi-view dataset Deep360 based on the CARLA autonomous driving simulator [8]. Fig. 7 shows some examples of the dataset. Each frame consists of six pairs of rectified panoramas, which cover all the pairwise combinations of four 360° cameras, six corresponding disparity maps and one ground truth depth map. All these images and maps have a resolution of 1024×512.

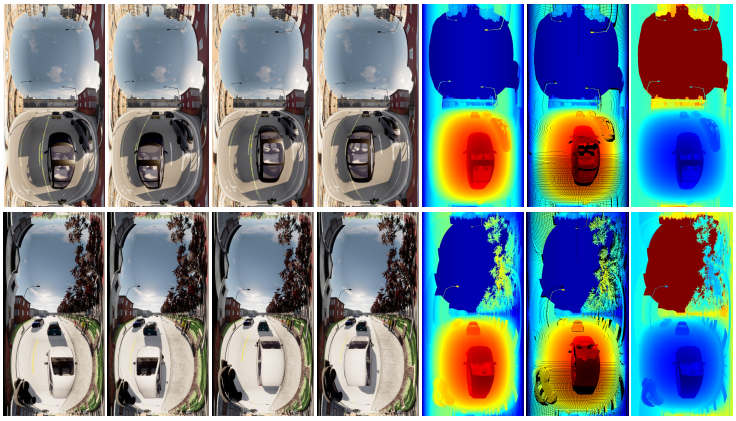

Fig. 7. Examples of the proposed Deep360 dataset. Each row shows an outdoor road scene. From left: four panoramas captured by 360° cameras, two disparity maps of 1-2 and 3-4 camera pairs, and one ground truth omnidirectional depth map

To acquire realistic 360° outdoor road scenes with high variety, we make the car with 360° cameras in CARLA drive automatically in six different towns and spawn many other random actors (pedestrian and vehicles).

We also provide a soiled version of the Deep360 dataset, which can be used to train and evaluate 360° depth estimation algorithms under the harsh circumstances in autonomous driving. The Deep360 (Soiled) dataset contains panoramas soiled or affected by three common outdoor factors: mud spots, water drops and glare. Fig. 8 shows the three kinds of soiled panoramas in our dataset.

An overview of the proposed dataset and other published 360° datasets is listed in Table 2.

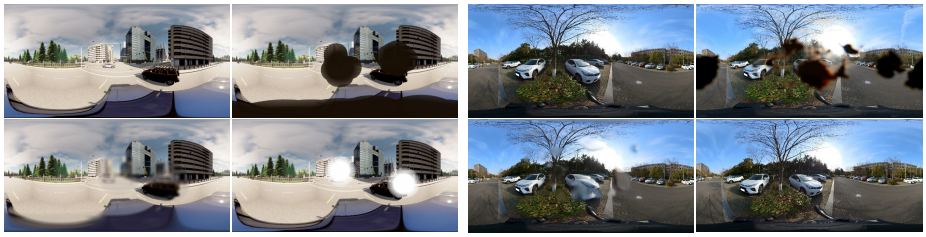

Fig. 8. Soiled panoramas in Deep360 (Soiled) and corresponding real-world examples. Left Four: synthetic panoramas; Right Four: real-world panoramas. Top Left: clear panoramas; Top Right: panoramas soiled by mud spots; Bottom Left: panoramas soiled by water drops; Bottom Right: panoramas dazzled by glare

Table 2. Overview of the proposed datasets and other published datasets

Dataset		Scene Category	Input Image	# Viewpoints	# Training Frames	# Test Frames	# Validation Frames
Won et al. [30,31,32]	Urban	outdoor	fish-eye	4	700	300	-
	OmniHouse	indoor	fish-eye	4	2048	512	-
	OmniThings	random objects	fish-eye	4	9216	1024	-
De La Garanderie et al. [16]	-	outdoor	panorama	1	-	200	-
Wang et al. [29]	SF3D	indoor	panorama	2	800	203	200
	MP3D	indoor	panorama	2	1602	341	431
Zioulis et al. [38]	3D60	indoor	panorama	3	7858	2190	1103
Ours	**Deep360**	outdoor	panorama	4	2100	600	300
	Deep360 (Soiled)	outdoor	panorama	4	2100	600	300

5 Experiments

5.1 Experimental Settings

Datasets We train and evaluate the proposed framework on Deep360 and the widely-used 3D60 [38] because these two datasets cover the outdoor and indoor scenes. For Deep360, panoramas from all four views are used to validate the performance of MODE on a multi-view setup. For 3D60, panoramas from two of three views are used to validate the performance of MODE on a binocular setup. More training details can be found in the supplementary material.

Evaluation Metrics For quantitative evaluation of the proposed framework, we use **MAE** (mean absolute error), **RMSE** (root mean square error), **Px1,3,5** (percentage of outliers with pixel error $> 1, 3, 5$), **D1** [21] (percentage of outliers with pixel error > 3 and $> 5\%$) to evaluate the disparity results, and use **MAE**, **RMSE**, **AbsRel** (absolute relative error), **SqRel** (square relative error), **SILog** [9] (scale-invariant logarithmic error), $\boldsymbol{\delta 1, 2, 3}$ [17] (accuracy with threshold that $\max(\frac{\hat{y}}{y^\star}, \frac{y^\star}{\hat{y}}) < 1.25, 1.25^2, 1.25^3$) to evaluate the depth results.

5.2 Experimental Results

We first evaluate the omnidirectional stereo matching network of MODE on the dataset Deep360. We compare it with the excellent stereo matching algorithms PSMNet [4] and AANet [33], and the omnidirectional algorithm 360SD-Net [29]. Because 360SD-Net is designed for up-down 360° stereo, we modified part of the model for left-right stereo matching. For PSMNet and AANet, we use the pre-trained models from the authors and follow their hyper-parameters to fine-tune on Deep360. The quantitative results in Table 3 illustrate that our stereo matching network with spherical feature learning achieves SOTA performance on 360° stereo matching.

Table 3. Quantitative results of stereo matching on the proposed Deep360 dataset. The metrics refer to disparity errors

Methods	Metrics					
	MAE(↓)	**RMSE(↓)**	**Px1(% ↓)**	**Px3(% ↓)**	**Px5(% ↓)**	**D1(% ↓)**
AANet [33]	0.5057	2.2232	7.7282	2.0914	1.1887	1.7929
360SD-Net [29]	0.4235	1.8320	6.6124	1.9080	1.0885	1.7753
PSMNet [4]	0.3501	1.8244	4.3798	1.3559	0.8398	1.2973
Ours	**0.2073**	**1.2347**	**2.6010**	**0.8767**	**0.5260**	**0.8652**

Then we evaluate the whole framework by comparing it with SOTA omnidirectional depth estimation methods. To present the performance of SOTA works on Deep360, we test different types of methods, including monocular UniFuse [14], binocular CSDNet [18], 360SD-Net [29], and multi-view OmniMVS [31]. All these models are fine-tuned with the pre-trained models from the authors. As shown in Table 4 and Fig. 9, our MODE framework performs favorably against SOTA omnidirectional depth estimation methods on different datasets, especially the one with soiled panoramas. Moreover, the consistent performance on datasets with different 360° multi-camera setups validates the extensibility of the framework. There is no result of OmniMVS on 3D60 since it can only take fish-eye images as input. We make a fish-eye version of our Deep360 dataset to implement the training and evaluation of OmniMVS. We evaluate and present the results of 360° depth estimation in ERP domain.

5.3 Ablation Studies

Table 5 shows the ablation studies of the omnidirectional stereo matching network. The results show that using panoramas without cropping and applying spherical convolution improve the performance. Table 6 illustrates the ablation studies of the depth map fusion network. The results show that the fusion stage improves the quality of depth maps. The rows of the table gradually show the improvement of adding each component into the network.

Table 4. Quantitative comparisons of omnidirectional depth estimation methods on different datasets. The metrics refer to depth errors

Datasets	Methods	Metrics							
		MAE(↓)	RMSE(↓)	AbsRel(↓)	SqRel(↓)	SILog(↓)	δ1(% ↑)	δ2(% ↑)	δ3(% ↑)
Deep360	UniFuse [14]	3.9193	28.8475	0.0546	0.3125	0.1508	96.0269	98.2679	98.9909
	CSDNet [18]	6.6548	36.5526	0.1553	1.7898	0.2475	86.0836	95.1589	97.7562
	360SD-Net [29]	11.2643	66.5789	0.0609	0.5973	0.2438	94.8594	97.2050	98.1038
	OmniMVS [31]	8.8865	59.3043	0.1073	2.9071	0.2434	94.9611	97.5495	98.2851
	MODE	**3.2483**	**24.9391**	**0.0365**	**0.0789**	**0.1104**	**97.9636**	**99.0987**	**99.4683**
Deep360 (Soiled)	UniFuse [14]	5.4636	37.4313	0.1119	4.8948	0.1810	95.2379	97.8686	98.7208
	CSDNet [18]	7.5950	38.4693	0.1631	3.7148	0.2521	86.7329	95.3295	97.7513
	360SD-Net [29]	22.5495	97.3958	0.1060	1.1857	0.4465	90.5868	94.1468	95.6262
	OmniMVS [31]	9.2680	62.1838	0.1935	22.6994	0.2597	94.7009	97.3821	98.1652
	MODE	**4.4652**	**31.7124**	**0.0495**	**0.1778**	**0.1458**	**96.3504**	**98.5718**	**99.2109**
3D60 [38]	UniFuse [14]	0.1868	0.3947	0.0799	0.0246	0.1126	93.2860	98.4839	99.4828
	CSDNet [18]	0.2067	0.4225	0.0908	0.0241	0.1273	91.9537	98.3936	99.5109
	360SD-Net [29]	0.0762	0.2639	0.0300	0.0117	1.4578	97.6751	98.6603	99.0417
	MODE	**0.0713**	**0.2631**	**0.0224**	**0.0031**	**0.0512**	**99.1283**	**99.7847**	**99.9250**

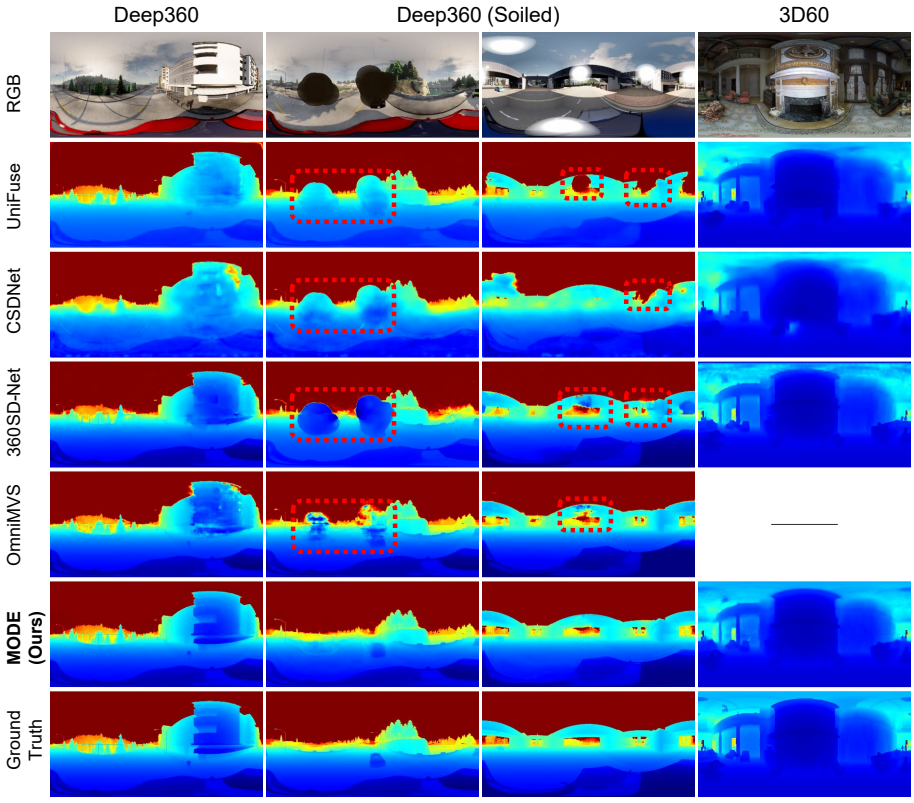

Fig. 9. Qualitative comparisons of omnidirectional depth estimation methods on different datasets. Red dotted boxes indicate the failed depth estimation caused by mud spots or glare

Table 5. Ablation studies for omnidirectional stereo matching on Deep360. We compare the performance of the proposed stereo matching network with and without Input Image Cropping (**cr**) and Spherical Convolution (**SC**). The metrics refer to disparity errors

Network setting	MAE(↓)	RMSE(↓)	Px1(% ↓)	Px3(% ↓)	Px5(% ↓)	D1(% ↓)
w/ cr	0.3220	1.7425	3.9787	1.3042	0.8049	1.2588
w/o cr	0.2109	1.2408	2.6509	0.8967	0.5377	0.8846
w/o cr + SC	**0.2073**	**1.2347**	**2.6010**	**0.8767**	**0.5260**	**0.8652**

Table 6. Ablation studies for the multi-view depth map fusion network on Deep360 (Soiled). The first row shows the results without fusion(w.r.t the results of stereo matching stage). The network in the second row is the baseline in this study, which consists of stacked 2D convolution layers. Different components used by our depth map fusion network are denoted as: Encoder-Decoder and Skip Connection architecture (**En-De-SC**); incorporation of reference panoramas (**img**); incorporation of confidence maps (**conf**). The metrics refer to depth errors

Network Setting	MAE(↓)	RMSE(↓)	AbsRel(↓)	SqRel(↓)	SILog(↓)	δ1(% ↑)	δ2(% ↑)	δ3(% ↑)
w/o fusion	15.2145	77.5905	0.1230	6.3135	0.5466	93.2377	96.0349	97.1837
Baseline	6.8699	50.1859	0.0586	0.8880	0.1996	95.7078	97.9644	98.6917
En-De-SC	6.2548	45.8603	0.0516	0.2702	0.1831	95.9953	98.1431	98.8211
En-De-SC+img	**4.2071**	32.0112	0.0710	0.2443	0.1554	95.1875	98.4766	99.1773
En-De-SC+img+conf	4.4652	**31.7123**	**0.0495**	**0.1778**	**0.1458**	**96.3504**	**98.5717**	**99.2109**

6 Conclusions

In this paper, we propose a two-stage framework, MODE, for multi-view omnidirectional depth estimation from 360° cameras. We adopt the Cassini projection to achieve the linear epipolar constraint of left-right 360° cameras, which improves the performance of omnidirectional stereo matching. The use of spherical convolution effectively overcomes the distortion of panoramas. The multi-view depth fusion improves the robustness of the framework through redundant design. The experimental results show that the proposed MODE achieves state-of-the-art performance on both indoor and outdoor datasets, and it is robust against soiled camera lenses and glare. Moreover, the framework is compatible with arbitrary 360° multi-camera setups. Apart from these, we also provide a large-scale synthetic road scene dataset with both high-quality and soiled panoramas. Finally, we test the proposed framework on the real-world environment with the model trained on synthetic data to validate the generalization and robustness of our framework.

Acknowledgements We acknowledge the computational resources provided by the High-Performance Computing Center of the Collaborative Innovation Center of Advanced Microstructures, Nanjing University, and Nanjing Institute of Advanced Artificial Intelligence.

References

1. Armeni, I., Sax, S., Zamir, A., Savarese, S.: Joint 2d-3d-semantic data for indoor scene understanding. ArXiv **abs/1702.01105** (2017)
2. Cassini projection: Cassini projection — Wikipedia, the free encyclopedia (2022), https://en.wikipedia.org/wiki/Cassini_projection
3. Chang, A., Dai, A., Funkhouser, T., Halber, M., Niebner, M., Savva, M., Song, S., Zeng, A., Zhang, Y.: Matterport3d: Learning from rgb-d data in indoor environments. In: 2017 International Conference on 3D Vision (3DV). pp. 667–676 (2017). https://doi.org/10.1109/3DV.2017.00081
4. Chang, J., Chen, Y.: Pyramid stereo matching network. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5410–5418 (2018). https://doi.org/10.1109/CVPR.2018.00567
5. Chen, R., Han, S., Xu, J., Su, H.: Point-based multi-view stereo network. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 1538–1547 (2019). https://doi.org/10.1109/ICCV.2019.00162
6. Cheng, X., Wang, P., Zhou, Y., Guan, C., Yang, R.: Omnidirectional depth extension networks. In: 2020 IEEE International Conference on Robotics and Automation (ICRA). pp. 589–595 (2020). https://doi.org/10.1109/ICRA40945.2020.9197123
7. Coors, B., Condurache, A.P., Geiger, A.: Spherenet: Learning spherical representations for detection and classification in omnidirectional images. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) Computer Vision – ECCV 2018. pp. 525–541. Springer International Publishing, Cham (2018)
8. Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V.: CARLA: An open urban driving simulator. In: Proceedings of the 1st Annual Conference on Robot Learning. pp. 1–16 (2017)
9. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. Advances in neural information processing systems **27** (2014)
10. Gu, X., Fan, Z., Zhu, S., Dai, Z., Tan, F., Tan, P.: Cascade cost volume for high-resolution multi-view stereo and stereo matching. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2492–2501 (2020). https://doi.org/10.1109/CVPR42600.2020.00257
11. Handa, A., Pătrăucean, V., Stent, S., Cipolla, R.: Scenenet: An annotated model generator for indoor scene understanding. In: 2016 IEEE International Conference on Robotics and Automation (ICRA). pp. 5737–5743. IEEE (2016)
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 770–778 (2016). https://doi.org/10.1109/CVPR.2016.90
13. Jiang, C.M., Huang, J., Kashinath, K., Prabhat, Marcus, P., Niessner, M.: Spherical CNNs on unstructured grids. In: International Conference on Learning Representations (2019), https://openreview.net/forum?id=Bkl-43C9FQ
14. Jiang, H., Sheng, Z., Zhu, S., Dong, Z., Huang, R.: Unifuse: Unidirectional fusion for 360° panorama depth estimation. IEEE Robotics and Automation Letters **6**(2), 1519–1526 (2021). https://doi.org/10.1109/LRA.2021.3058957
15. Kendall, A., Martirosyan, H., Dasgupta, S., Henry, P., Kennedy, R., Bachrach, A., Bry, A.: End-to-end learning of geometry and context for deep stereo regression. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 66–75 (2017). https://doi.org/10.1109/ICCV.2017.17

16. de La Garanderie, G.P., Abarghouei, A.A., Breckon, T.P.: Eliminating the blind spot: Adapting 3d object detection and monocular depth estimation to 360 panoramic imagery. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 789–807 (2018)
17. Ladický, L., Shi, J., Pollefeys, M.: Pulling things out of perspective. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 89–96 (2014). https://doi.org/10.1109/CVPR.2014.19
18. Li, M., Hu, X., Dai, J., Li, Y., Du, S.: Omnidirectional stereo depth estimation based on spherical deep network. Image and Vision Computing **114**, 104264 (2021). https://doi.org/https://doi.org/10.1016/j.imavis.2021.104264, `https://www.sciencedirect.com/science/article/pii/S0262885621001694`
19. Lipson, L., Teed, Z., Deng, J.: Raft-stereo: Multilevel recurrent field transforms for stereo matching. In: International Conference on 3D Vision (3DV) (2021)
20. Mayer, N., Ilg, E., Häusser, P., Fischer, P., Cremers, D., Dosovitskiy, A., Brox, T.: A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4040–4048 (2016). https://doi.org/10.1109/CVPR.2016.438
21. Menze, M., Heipke, C., Geiger, A.: Joint 3d estimation of vehicles and scene flow. In: ISPRS Workshop on Image Sequence Analysis (ISA) (2015)
22. Pang, J., Sun, W., Ren, J.S., Yang, C., Yan, Q.: Cascade residual learning: A two-stage convolutional neural network for stereo matching. In: 2017 IEEE International Conference on Computer Vision Workshops (ICCVW). pp. 878–886 (2017). https://doi.org/10.1109/ICCVW.2017.108
23. Poggi, M., Kim, S., Tosi, F., Kim, S., Aleotti, F., Min, D., Sohn, K., Mattoccia, S.: On the confidence of stereo matching in a deep-learning era: a quantitative evaluation. IEEE Transactions on Pattern Analysis and Machine Intelligence pp. 1–1 (2021). https://doi.org/10.1109/TPAMI.2021.3069706
24. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241. Springer (2015)
25. Shen, Z., Dai, Y., Rao, Z.: Cfnet: Cascade and fused cost volume for robust stereo matching. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 13906–13915 (June 2021)
26. Song, S., Yu, F., Zeng, A., Chang, A.X., Savva, M., Funkhouser, T.: Semantic scene completion from a single depth image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1746–1754 (2017)
27. Wang, F.E., Hu, H.N., Cheng, H.T., Lin, J.T., Yang, S.T., Shih, M.L., Chu, H.K., Sun, M.: Self-supervised learning of depth and camera motion from 360° videos. In: Asian Conference on Computer Vision. pp. 53–68. Springer (2019)
28. Wang, F.E., Yeh, Y.H., Sun, M., Chiu, W.C., Tsai, Y.H.: Bifuse: Monocular 360 depth estimation via bi-projection fusion. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 459–468 (2020). https://doi.org/10.1109/CVPR42600.2020.00054
29. Wang, N.H., Solarte, B., Tsai, Y.H., Chiu, W.C., Sun, M.: 360sd-net: 360° stereo depth estimation with learnable cost volume. In: 2020 IEEE International Conference on Robotics and Automation (ICRA). pp. 582–588 (2020). https://doi.org/10.1109/ICRA40945.2020.9196975
30. Won, C., Ryu, J., Lim, J.: Sweepnet: Wide-baseline omnidirectional depth estimation. In: 2019 International Conference on Robotics and Automation (ICRA). pp. 6073–6079 (2019). https://doi.org/10.1109/ICRA.2019.8793823

31. Won, C., Ryu, J., Lim, J.: Omnimvs: End-to-end learning for omnidirectional stereo matching. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8987–8996 (2019)
32. Won, C., Ryu, J., Lim, J.: End-to-end learning for omnidirectional stereo matching with uncertainty prior. IEEE transactions on pattern analysis and machine intelligence (2020)
33. Xu, H., Zhang, J.: Aanet: Adaptive aggregation network for efficient stereo matching. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1956–1965 (2020). https://doi.org/10.1109/CVPR42600.2020.00203
34. Yang, J., Mao, W., Alvarez, J., Liu, M.: Cost volume pyramid based depth inference for multi-view stereo. IEEE Transactions on Pattern Analysis and Machine Intelligence pp. 1–1 (2021). https://doi.org/10.1109/TPAMI.2021.3082562
35. Yao, Y., Luo, Z., Li, S., Fang, T., Quan, L.: Mvsnet: Depth inference for unstructured multi-view stereo. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 767–783 (2018)
36. Žbontar, J., LeCun, Y.: Stereo matching by training a convolutional neural network to compare image patches. Journal of Machine Learning Research **17**(65), 1–32 (2016), http://jmlr.org/papers/v17/15-535.html
37. Zhang, F., Prisacariu, V., Yang, R., Torr, P.H.: Ga-net: Guided aggregation net for end-to-end stereo matching. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 185–194 (2019)
38. Zioulis, N., Karakottas, A., Zarpalas, D., Alvarez, F., Daras, P.: Spherical view synthesis for self-supervised 360° depth estimation. In: 2019 International Conference on 3D Vision (3DV). pp. 690–699 (2019). https://doi.org/10.1109/3DV.2019.00081
39. Zioulis, N., Karakottas, A., Zarpalas, D., Daras, P.: Omnidepth: Dense depth estimation for indoors spherical panoramas. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 453–471. Computer Vision – ECCV 2018, Springer International Publishing (2018)