OmniVidar: Omnidirectional Depth Estimation from Multi-Fisheye Images

Sheng Xie[1], Daochuan Wang[2], Yunhui Liu[1,2,3,*]
Harbin Institute of Technology, Shenzhen[1]
Hong Kong Center for Logistics Robotics[2], The Chinese University of Hong Kong[3]
xiesheng@stu.hit.edu.cn, wangdaochuan@hotmail.com, yhliu@mae.cuhk.edu.hk

Abstract

Estimating depth from four large field of view (FoV) cameras has been a difficult and understudied problem. In this paper, we proposed a novel and simple system that can convert this difficult problem into easier binocular depth estimation. We name this system OmniVidar, as its results are similar to LiDAR, but rely only on vision. OmniVidar contains three components: (1) a new camera model to address the shortcomings of existing models, (2) a new multi-fisheye camera based epipolar rectification method for solving the image distortion and simplifying the depth estimation problem, (3) an improved binocular depth estimation network, which achieves a better balance between accuracy and efficiency. Unlike other omnidirectional stereo vision methods, OmniVidar does not contain any 3D convolution, so it can achieve higher resolution depth estimation at fast speed. Results demonstrate that OmniVidar outperforms all other methods in terms of accuracy and performance.

1. Introduction

Depth estimation from images is an important research field in computer vision, as it enables the acquisition of depth information with low-cost cameras for a wide range of applications. Traditional stereo cameras with pinhole lenses are limited in their FoV. However, many scenarios require an omnidirectional depth map, such as autonomous driving [42] and robot navigation [13, 43]. Although there are active sensors available that can provide omnidirectional depth information, such as LiDAR [25], their high cost makes them less accessible than stereo vision. Passive sensors, such as RGB cameras, are a common choice for depth estimation due to their low cost, lightweight, and low power consumption. To increase the FoV, fisheye lenses are often introduced into stereo vision setups.

Over the past few decades, various methods have been proposed for depth estimation using fisheye cameras. These include the binocular fisheye system [31], up-down fisheye system [13], and catadioptric cameras [17, 23, 32]. However, all of these approaches have limitations. The binocular fisheye system cannot provide an omnidirectional perception. The up-down fisheye system and catadioptric cameras can offer horizontal 360° depth perception, but their vertical FoV is limited. Furthermore, catadioptric cameras tend to be bulky, challenging to calibrate, and prone to errors. It turns out that the best choice for omnidirectional depth estimation is a system consisting of four cameras with extremely wide FoV (> 180°). This system enables

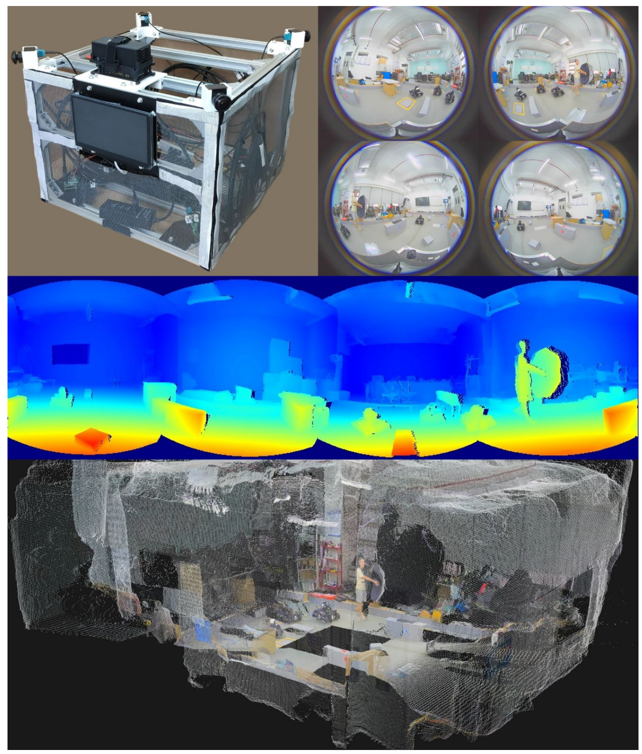

Figure 1. Our prototype that built with four 250° fisheye cameras and its results of dense inverse distance map and cloud points. It can get great depth estimation results in real scenes and achieve real-time performance on modern GPU.

*Corresponding author.

360° depth estimation both horizontally and vertically, and is light-weight, convenient to calibrate and maintain. Several studies [21,29,39,41] have shown excellent results with this approach. However, these methods often use many 3D convolutions, resulting in low efficiency. To adapt to limited memory, the images must be down-sampled, leading to a lower resolution of the output depth map. Furthermore, many of these approaches perform depth estimation using the original fisheye image without explicitly processing image distortion, which leads to the construction of a tedious and complicated cost volume and can result in reduced accuracy.

Moreover, due to the complex imaging process of large FoV fisheye lenses, handling fisheye image distortion using mathematical formulas can be a challenging task. While several excellent large FoV fisheye camera models have been proposed [2, 18, 19, 27, 30, 36], our experiments have shown that none of these models can accurately approximate the imaging process, and we get less satisfactory depth estimation results when using them.

Inspired by the above observations, we propose OmniVidar, a novel and simple multi-fisheye omnidirectional depth estimation system, as shown in Figure 2. OmniVidar contains three components. Firstly, we improve the DSCM [36] and propose a new camera model, named Triple Sphere Camera Model (TSCM), which can better approximate the imaging process and achieve the best accuracy. Then we propose an epipolar rectification algorithm designed for multi-fisheye camera system. We leverage a cubic-like projection approach to transform the four fisheye camera systems into four binocular camera systems and then conduct epipolar rectification on each binocular system. This method solves the distortion issue and reduces the complex multi-fisheye omnidirectional depth estimation problem to a much simpler binocular depth estimation problem. Finally, we design a lightweight binocular depth estimation network based on RAFT [35]. We add Transformer encoder [37] into feature extraction in RAFT to combine the advantages of Transformer and GRU, and it's easy to balance the accuracy and efficiency.

We compare OmniVidar with existing methods on several datasets. The results demonstrate that our method outperforms all the others in terms of speed and accuracy, achieving State-of-The-Art performance.

2. Related Work

2.1. Multiview Fisheye Stereo

Gao *et al*. [13] mount two large FoV cameras (245°) vertically and reversely to obtain a 360° horizontal and 65° vertical FoV overlap and use this system in UAV for depth perception [43]. Won *et al*. [40] use four cameras with 220° FoV and propose SweepNet algorithm. They propose Sphere-Sweep method modified from Plane-Sweep [16] to construct the cost volume and then use the cost aggregation algorithm in SGM [15] to obtain depth. After this, Won *et al*. propose OmniMVS [39,41], which uses 3D convolution instead of SGM for depth regression and achieves higher accuracy. Komatsu *et al*. [21] propose CrownConv360. They improve OmniMVS by using the icosahedral projection in the feature extraction and depth regression stage for distortion reduction. Instead of using neural network, Meuleman *et al*. [29] choose to investigate cost aggregation and propagation in traditional stereo matching. And they achieve real-time high-resolution depth estimation at 29 fps on the embedded device TX2. However, the algorithm is only suitable for short camera baseline.

All of the above works have more or fewer deficiencies. We propose OmniVidar, an omnidirectional depth estimation method with high accuracy, which can run in real-time and solve the fisheye image distortion problem with good generalization in various scenes.

2.2. Fisheye Image Distortion

To solve the distortion problem, a camera model suitable for fisheye is required, and the distortion should be handled explicitly in depth estimation.

Scaramuzza *et al*. [30] propose a high-order polynomial-based camera projection model, which is highly generalizable and can be widely used for various catadioptric and fisheye cameras. But it has no closed-form solution, so the inverse projection equation needs to be fitted with a high-order polynomial, which introduces errors. Mei *et al*. [27] propose the MUCM, which is an improvement of the UCM [2] to achieve higher calibration accuracy and propose a novel intrinsic parameters initialization method. Usenko *et al*. [36] propose the DSCM, which has only one more intrinsic parameter than the MUCM model, but the accuracy is substantially improved.

Su *et al*. [34] propose a mini-network that learns to modify the shape of filters based on location. Coors *et al*. [8] chose to use a camera projection model to shape the convolution kernel to directly address distortion. An alternative idea was chosen by Cohen *et al*. [7] who define a form of convolution on the Lie group $SO(3)$ that convolves over the rotation space rather than the translation space. Eder *et al*. [9] compared the performance of three approaches: equirectangular projection, cubic projection, and 7-level subdivided icosahedral projection, on depth estimation and semantic segmentation tasks. The results showed that icosahedral projection achieved the best results in all tasks. Komatsu *et al*. [21] used icosahedral projection in multi-fisheye omnidirectional stereo vision to improve the depth estimation results of OmniMVS. Eder *et al*. [10] proposed a tangent plane projection modified from the icosahedron projection.

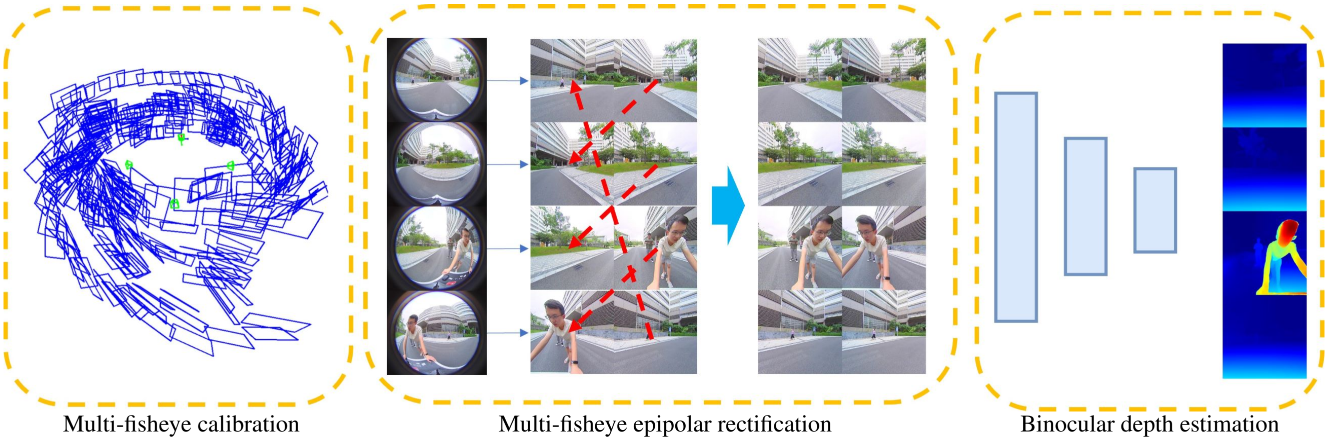

Figure 2. System overview. Our system contains three parts: multi-fisheye calibration, multi-fisheye epipolar rectification, and binocular depth estimation. In epipolar rectification, the images in the left column are original. The images in the middle column are the images after abstracting each fisheye camera into two pinhole cameras, where the two images connected by a red dotted arrow are a pair of binocular cameras. The images are rearranged and combined to obtain the right column.

The above algorithms have a common feature that they are all designed for the monocular image. Thus they perform well on tasks that require only one camera, yet not much good for multi-fisheye tasks. Therefore, to address this problem, we propose a distortion removal method designed for multi-fisheye systems, which greatly simplifies the depth estimation problem at the same time.

2.3. RAFT in Stereo Vision

Zachary *et al*. [35] propose Recurrent All-Pairs Field Transforms (RAFT), a highly efficient and accurate deep network architecture for optical flow. RAFT uses modified GRU blocks to iteratively refine optical flow and shares weights between refinement units. Lahav *et al*. [24] apply the RAFT to binocular rectified stereo vision and introduce multi-level convolutional GRUs, which can more efficiently propagate information across the image. Wang *et al*. [38] modify RAFT and propose PVStereo which greatly outperforms other self-supervised stereo matching approaches. Li *et al*. [22] combine RAFT and Transformer and propose CREStereo. They add Transformer encoder in image feature extraction and repeat the cross-attention between left and right images in the iterative update stage. This method performs well in various binocular datasets with excellent generalization.

We imitate CREStereo and improve it based on RAFT to obtain a binocular stereo vision depth estimation network with a better trade-off between efficiency and accuracy.

3. Methods

The overall flow of our method is shown in Figure 2 and is divided into three parts: camera calibration, epipolar rectification, and depth estimation.

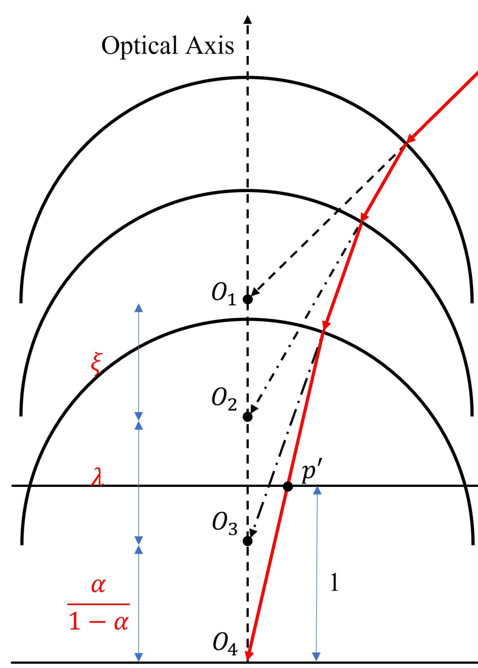

Figure 3. Our Triple Sphere Camera Model. First, point P is projected onto the first sphere and then onto the second sphere, then the third. The second sphere is shifted w.r.t. the first sphere by ξ, and the third is shifted w.r.t. the second sphere by λ. Finally, the point is projected onto the image plane of a pinhole camera which is shifted by $\frac{\alpha}{1-\alpha}$ from the third sphere.

3.1. Camera Calibration

We propose the Triple Sphere Camera Model, which can better fit the imaging process of large FoV fisheye cameras and has a closed-form unprojection solution.

As shown in Figure 3, our projection model considers that the incident light is refracted three times, and the displacements of the three unit spherical centers are ξ and λ. After three refractions, the incident light is finally projected to the image plane according to the pinhole camera model, and the displacement of the pinhole camera model's optical center from the third unit sphere is $\frac{\alpha}{1-\alpha}$. Therefore, our model has totally 7 camera internal parameters: $f_x, f_y, c_x, c_y, \xi, \lambda, \alpha$. The projection equations are defined as follows.

$$\pi(P, i) = \frac{1}{\zeta}\begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \end{bmatrix}\begin{bmatrix} X \\ Y \\ \zeta \end{bmatrix} \tag{1}$$

$$\zeta = Z + \xi d_1 + \lambda d_2 + \frac{\alpha}{1-\alpha} d_3 \tag{2}$$

$$d_1 = \sqrt{X^2 + Y^2 + Z^2} \tag{3}$$

$$d_2 = \sqrt{X^2 + Y^2 + (\xi d_1 + Z)^2} \tag{4}$$

$$d_3 = \sqrt{X^2 + Y^2 + (\lambda d_2 + \xi d_1 + Z)^2} \tag{5}$$

where i is the vector of intrinsic parameters, π is the projection function.

A set of 3D points that results in valid projection is expressed as follows:

$$\Omega = \{x \in \mathbb{R}^3 | z > -w_2 d_1\} \tag{6}$$

$$w_2 = \frac{\xi + \lambda + w_1}{\sqrt{1 + (\xi + \lambda)^2 + 2w_1(\xi + \lambda)}} \tag{7}$$

$$w_1 = \frac{\alpha}{1-\alpha} \tag{8}$$

The unprojection function is computed as follows:

$$\pi^{-1}(p, i) = \mu \begin{bmatrix} \eta\gamma x \\ \eta\gamma y \\ m_z \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ \xi \end{bmatrix} \tag{9}$$

$$\mu = \xi m_z + \sqrt{\xi^2 {m_z}^2 - \xi^2 + 1} \tag{10}$$

$$m_z = \eta(\gamma - \zeta) - \lambda \tag{11}$$

$$\eta = \lambda(\gamma - \zeta) + \sqrt{\lambda^2(\gamma - \zeta)^2 - \lambda^2 + 1} \tag{12}$$

$$\gamma = \frac{\zeta + \sqrt{1 + (1-\zeta^2)(x^2 + y^2)}}{x^2 + y^2 + 1} \tag{13}$$

$$\zeta = \begin{cases} \frac{\alpha}{1-\alpha} & \text{if } \alpha \leq 0.5 \\ \frac{1-\alpha}{\alpha} & \text{if } \alpha > 0.5 \end{cases} \tag{14}$$

where (x, y) is the normalized coordinate.

We calibrate the intrinsic and extrinsic parameters of multi-fisheye camera system using a highly accurate planar checkerboard. The goal of the calibration is to minimize the reprojection error of the corner points in all pictures. For each image n in the calibration sequence, the corner detector can obtain the projection point u_{nk} of the kth corner point x_k. The coordinate of u_{nk} is related to the camera intrinsic and extrinsic parameters. Let $s = [i, T_{cam_0}, T_{cam_1}, ..., T_{cam_n}]$ be the parameter to optimize. We can construct the nonlinear optimization problem as follows:

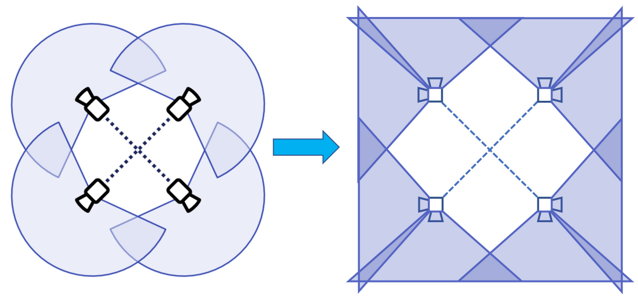

Figure 4. The multi-fisheye epipolar rectification schematic. Each fisheye camera is converted into two pinhole cameras with an angle of 90°. Then the system becomes four binocular systems.

$$s = \arg\min_{s'} \sum_{n=0}^{N} \sum_{k \in K} \rho((\pi(T_{cam_n} x_k, \boldsymbol{i}) - u_{nk})^2) \tag{15}$$

where $T_{cam_n} \in SE(3)$ is th transformation from the coordinate frame of the calibration grid to the camera coordinate frame for image n. K is a set of detected corner points for the image n and ρ is the robust Huber norm.

Due to the highly nonlinear nature of the image,the above optimization problem requires a good initial value. We initialize the intrinsic parametes using the previously proposed method [27] and find initial poses using the UPnP algorithm [20]. After obtaining the initial values of the parameters, we completed the optimization operation using Ceres Solver [1].

3.2. Multi-Fisheye Camera Epipolar Rectify

We propose a simple and effective undistortion method that cleverly transforms the omnidirectional depth estimation problem into a binocular depth estimation problem. And the method can be used in the system with any number of fisheye cameras. As shown in Figure 4, we abstract each fisheye camera as two pinhole cameras with 90° angle [1]. Then the four fisheye cameras system becomes four independent binocular systems. Each binocular systems can be rectified through epipolar rectification. We modified the classical epipolar rectification algorithm proposed by Fusiello [11] for our system.

[1] Although here we only show the horizontal binoculars, it is possible to construct binoculars facing both above and below to estimate the true omnidirectional depth. However, we consider that the above and below depth values are rarely used in practical applications. So we just use a limited 110° vertical FoV, which satisfies most of the application requirements.

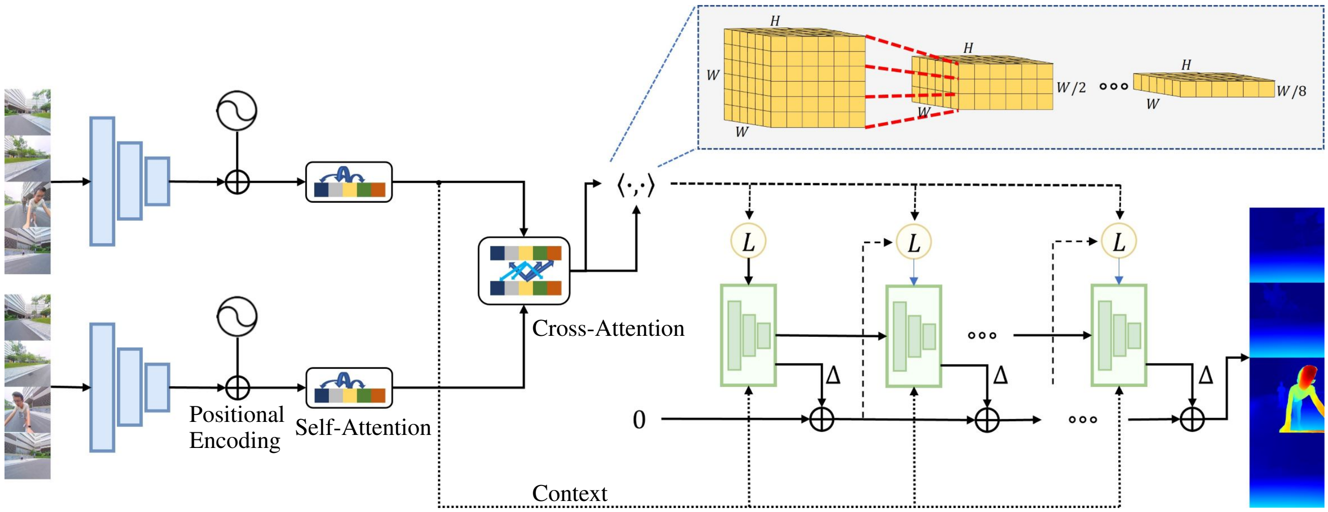

Figure 5. The structure of our proposed binocular depth estimation network. A pair of stereo images are fed into feature extraction networks. Then the output features are calculated by self-attention and cross-attention, and after doing self-attention, the features of left image are used for context information for latter recurrent update stage. Then the left and right features are multipled to construct cost volume. Finally use the disparity update model same as RAFT [35].

Firstly, the rotation matrix of the cameras in binocular system is calculated. Let the optical centers of the left camera and right camera be $C_1 = \begin{bmatrix} x_1 & y_1 & z_1 \end{bmatrix}^T$, $C_2 = \begin{bmatrix} x_2 & y_2 & z_2 \end{bmatrix}^T$, the rotation matrix can be calculated as follows:

$$R = \begin{bmatrix} r_1^T & r_2^T & r_3^T \end{bmatrix} \tag{16}$$

$$r_1 = \begin{bmatrix} x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \end{bmatrix}.normalize() \tag{17}$$

$$r_3 = \begin{bmatrix} -(z_2 - z_1) & 0 & x_2 - x_1 \end{bmatrix}.normalize() \tag{18}$$

$$r_2 = r_3 \times r_1 \tag{19}$$

Then remap the image for epipolar rectification. Taking the left camera in one binocular system as an example. Let the rotation matrix of the corresponding fisheye camera be R, the optical center be C. For a centain spatial point $P = (X, Y, Z, 1)$, its projection point on normalized plane is $p = (u, v, 1)$. The projection equation is shown below.

$$\mu \begin{bmatrix} u \\ v \\ f(u,v) \end{bmatrix} = \begin{bmatrix} R & -RC \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{20}$$

Let the rotation matrix of the rectified pinhole camera be R', and the corresponding normalized plane coordinate be $(u', v', 1)$. Then the projection equation is shown below.

$$\mu' \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} R' & -R'C \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{21}$$

The relationship between fisheye image and virtual binocular image can be obtained, see equation (22).

$$\lambda \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} R' & -R'C \end{bmatrix} \begin{bmatrix} R & -RC \end{bmatrix}^{-1} \begin{bmatrix} u \\ v \\ f(u,v) \end{bmatrix} \tag{22}$$

That is, for each pixel in the original fisheye image, the pixel coordinate in the pinhole image can be calculated. We can use pixel interpolation to obtain the rectified image.

As shown in Figure 2, it can be seen that after epipolar rectification, the four fisheye images are converted into a pair of binocular images, where each image is stitched together from the four virtual pinhole camera images. A simple binocular depth estimation algorithm is then used.

3.3. Stereo Depth Estimation

Since our system needs to estimate the depth values of four binocular systems simultaneously, which is very computationally intensive, our primary goal is not to design a binocular algorithm with the highest accuracy, but to minimize the accuracy loss while ensuring real-time performance.

The structure of our binocular depth estimation network is shown in Figure 5. We added Transformer [37] with self-attention and cross-attention in the image feature extraction stage. In binocular depth estimation, limited by computational performance and graphics card memory, many algorithms have to limit the matching search range of pixels. In this paper, we design an ingenious cost-volume calculation that can extend the estimable disparity range to the image width, while also reducing the computational complexity.

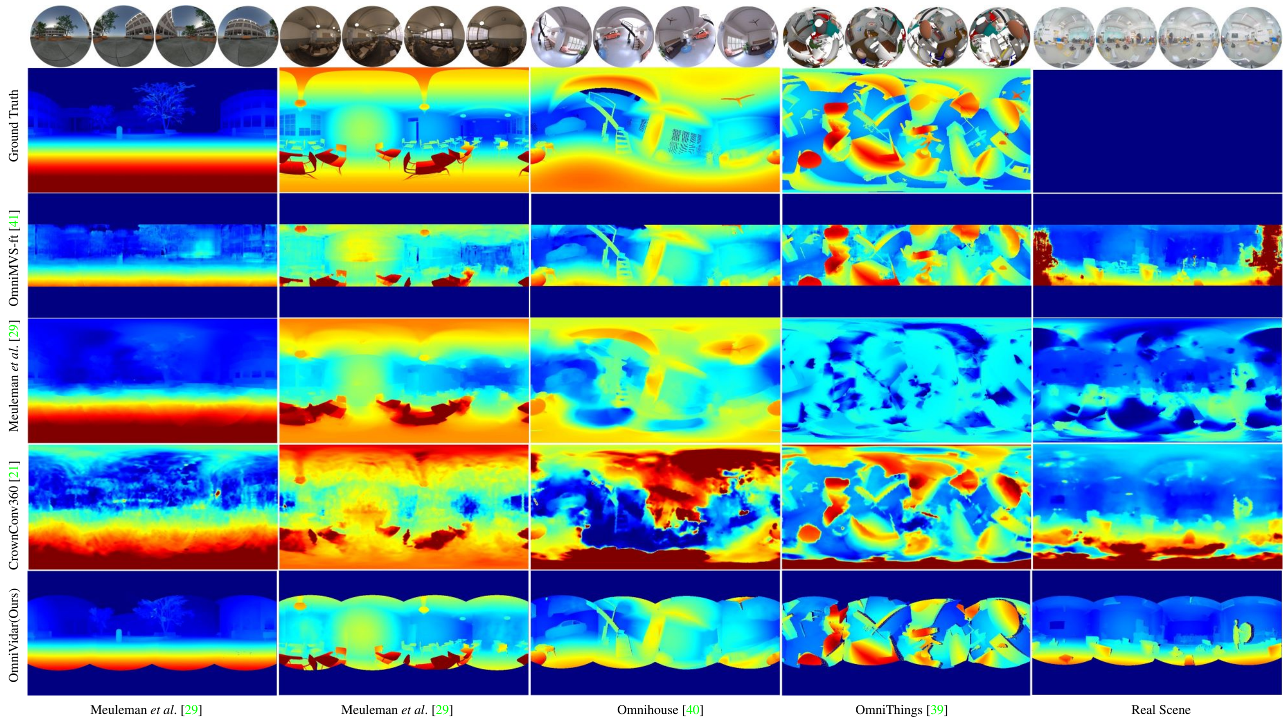

Figure 6. The depth estimation comparison between ours and other methods. The final row is the results from our method, and the column annotations indicate the dataset source.

Given two images $I_1 \in \mathbb{R}^{H \times W \times C}$, $I_2 \in \mathbb{R}^{H \times W \times C}$, we need to compute the cost volume $C(I_1, I_2) \in \mathbb{R}^{H \times W \times W}$. We found that the cost volume can be obtained by single matrix operation as follows.

$$C(I_1, I_2) = I_1 \in \mathbb{R}^{H \times W \times C} \cdot I_2^T \in \mathbb{R}^{H \times C \times W} \tag{23}$$

To enable the network to learn information at multiple scales, we used the cost volume pooling opertaion in RAFT-Stereo [24].

4. Experiments

4.1. Implementation

Our depth estimation network structure is shown in Figure 5. The number of channels in our network is identical to RAFT [35]. The feature encoder consists of 6 residual blocks, 2 at 1/2 resolution, 2 at 1/4 resolution, and 2 at 1/8 resolution. The number of channels is 64, 128, and 256, respectively. The 1/8 resolution feature map is then fed into the Transformer encoder [37]. We set the number of heads in attention to 8. The GRU iterative update part is identical to RAFT, and more details can be found in [35]. In addition, we replace all the normalization operators in RAFT with domain normalize [44].

Since we convert the multi-fisheye cameras to four binocular cameras, we can use a large number of public binocular datasets for training, instead of multi-fisheye datasets, which are very limited. Similar to the training strategy in CREStereo [22], we collected most of public binocular datasets, including SceneFlow [26], Sintel [4], CREStereo [22], KITTI [14, 28], InStereo2K [3], and Virtual KITTI [12]. The images are uniformly randomly cropped with a fixed resolution during the training process to solve the problem of different image resolutions in different datasets.

We use a single NVIDIA RTX2080Ti for training, the batch size is set to 4, the learning rate is 0.0004, and we use Adam to train 200 epochs to get the final network weights.

To test the performance of OmniVidar in real scenarios, we built a test setup. We used four leopard LI-USB30-OV10640-490-GMSL cameras with an image resolution of 1280×1024, paired with a fisheye lens with 250° FoV. The four cameras look in four directions, front and back, left and right, as shown in Figure 1. We use the device to capture indoor, corridor, and outdoor images, which can fully test the performance of OmniVidar in real scenes.

4.2. Evaluation

We perform the comparison on the Omnihouse, OmniThings provided by Won *et al*. [40, 41], and the datasets provided by Meuleman *et al*. [29]. Our OmniVidar has only been trained on the binocular dataset mentioned in Section 4.1, not on the test dataset, and the methods involved in the

Method	Meuleman *et al.* [29]			Omnihouse [40]			Omnithings [39]		
	bad 1.0	RMSE	MAE	bad 1.0	RMSE	MAE	bad 1.0	RMSE	MAE
OmniVidar(Ours)	**18.499**	**0.921**	**0.803**	**4.886**	1.002	1.042	**11.199**	**0.958**	**0.461**
OmniMVS-ft [41]	60.372	2.306	1.732	26.357	0.929	0.705	59.753	2.270	1.723
CrownConv360 [21]	69.119	4.020	3.005	85.559	11.077	6.729	93.946	9.545	6.518
Meuleman *et al.* [29]	44.111	1.664	1.273	11.304	**0.671**	**0.410**	65.348	2.550	1.941

Table 1. Quantitative comparison of algorithm accuracy on several datasets. The less the number is, the more accurate the method is.

comparison all use the training method corresponding to the original paper.

As can be seen in the Figure 6, although OmniVidar is not trained and fine-tuned on the above datasets, its depth estimation accuracy on these datasets far exceeds that of other methods, with accurate depth estimation in weakly textured regions.

Method	Runtime	Memory	Parameters
OmniVidar (Ours)	66 ms	3.7 GB	4.8 MB
OmniMVS-ft [41]	110 ms	6.8 GB	43.7 MB
CrownConv360 [21]	623 ms	6.3 GB	16.7 MB

Table 2. Efficiency comparison with other deep learning methods.

The analysis of the quantitative results are shown in the Table 1 and Table 2. It can be seen that OmniVidar has the least memory consumption, the shortest time consumption and the highest accuracy compared to other deep learning based solutions. Compared with non-deep learning methods, the accuracy is much higher than it, despite the relatively low efficiency.

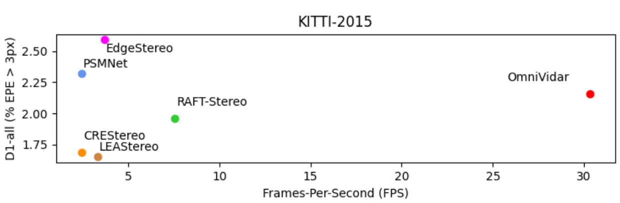

Figure 7. Plot comparing the accuracy and efficiency evaluated in KITTI-2015 [28]. It can be seen that OmniVidar achieves real-time performance while maintain the accuracy.

We also compared the binocular depth estimation network in OmniVidar with other binocular depth estimation networks. We chose the KITTI-2015 dataset [28] for the evaluation and the results are shown in Figure 7. We compare our method with CREStereo [22], LEAStereo [6], PSMNet [5], EdgeStereo [33], RAFT-Stereo [24]. It can see that our scheme achieves a better balance between accuracy and efficiency.

4.3. Ablation

Camera Model: We compare the effect of omnidirectional depth estimation under several camera models.

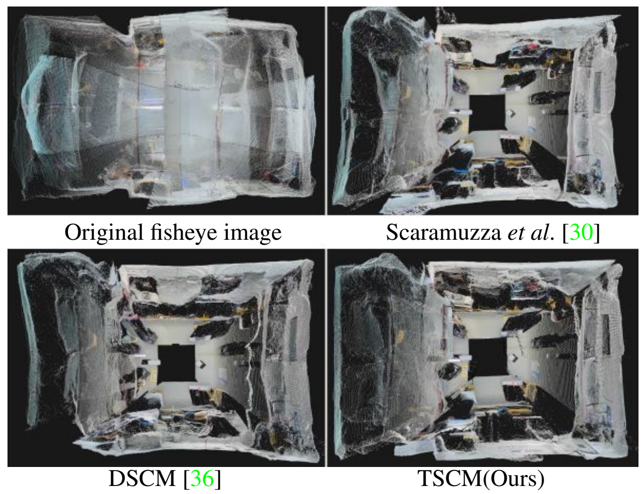

Figure 8. Depth estimation results using different camera projection models. Our TSCM can best remove the fisheye distortion and obtain the most accurate result.

As shown in Figure 8, the distortion of the fisheye lens is not sufficiently handled in the DSCM [36] and the model proposed by Scaramuzza *et al.* [30], which affects the accuracy of depth estimation and leads to serious distortion of objects in the point cloud. In contrast, the quality of the point cloud obtained by the TSCM is much better, with flat wall surfaces and more accurate distance measurements.

FoV	2 sphere	3 sphere (Ours)	4 sphere
195°	0.1530 px	0.1528 px	0.1528 px
220°	0.2031 px	0.1970 px	0.1952 px
250°	0.1365 px	0.1255 px	0.1222 px

Table 3. Mean reprojection error for camera models with different number sphere.

We also compared the accuray with different number of the refract sphere. Table 3 shows the results. It can be seen

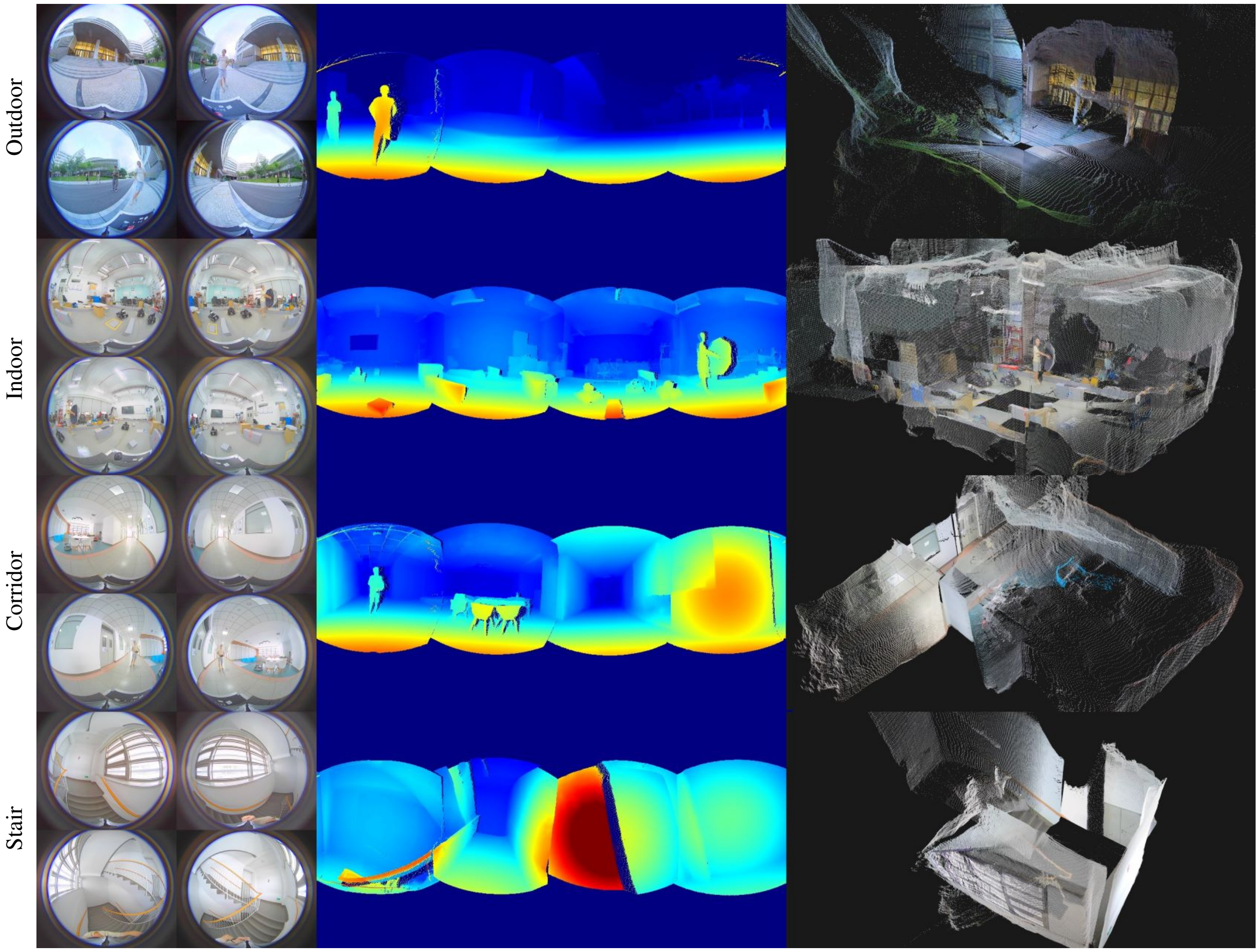

Figure 9. The images captured with our prototype and the inverse depth map and cloud points obtained by OmniVidar. It can be seen that OmniVidar has great generalization in various scenes, and performs well in low-textured areas.

that the camera model with 3 sphere can significantly improve the accuracy in large FoV scenes, while the improvement brought by more sphere is very limited.

Distortion Handle: To verify whether the improvement of depth estimation accuracy is helped by removing the fisheye image distortion using epipolar rectification, we estimate the depth directly using the original fisheye image and compare it with the undistorted method. Unlike the undistorted binocular stereo, we use the Sphere-Sweep algorithm [40] to construct the cost volume. This step does not contain trainable parameters, so it does not affect the network itself and ensures the fairness of the comparison.

The point cloud visualization of the results of the depth estimation of both is shown in Figure 8. It can be seen that when using original image, the depth information obtained from different cameras is not consistent. In contrast, the point cloud obtained by depth estimation after epipolar rectification shows the structural framework of the scene realistically and accurately.

4.4. Test in Real

We test the performance of the OmniVidar algorithm on a real dataset. Figure 9 shows the inverse depth maps and point clouds obtained by OmniVidar on the real dataset. It can be seen that our network has a strong generalization to obtain high-quality point clouds on real datasets.

5. Conclusion

In this paper, we propose a novel, simple and effective system OmniVidar, to address the inefficiency of current omnidirectional depth estimation methods due to the extensive use of 3D convolution and the poor accuracy due to the lack of explicit handling of distortion in fisheye images. Our method outperforms all other methods in terms of time consumption and accuracy.

References

[1] Sameer Agarwal, Keir Mierle, and The Ceres Solver Team. Ceres solver. http://ceres-solver.org. 4

[2] Simon Baker and Shree K. Nayar. A theory of single-viewpoint catadioptric image formation. *Int. J. Comput. Vision*, 35(2):175–196, nov 1999. 2

[3] 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):212101, 2019. 6

[4] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In A. Fitzgibbon et al. (Eds.), editor, *European Conf. on Computer Vision (ECCV)*, Part IV, LNCS 7577, pages 611–625. Springer-Verlag, Oct. 2012. 6

[5] Jia-Ren Chang and Yong-Sheng Chen. Pyramid stereo matching network. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5410–5418, 2018. 7

[6] 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. 7

[7] Taco S. Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Spherical CNNs. In *International Conference on Learning Representations*, 2018. 2

[8] Benjamin Coors, Alexandru Paul Condurache, and Andreas Geiger. Spherenet: Learning spherical representations for detection and classification in omnidirectional images. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, pages 525–541. Springer International Publishing, 2018. 2

[9] Marc Eder and Jan-Michael Frahm. Convolutions on spherical images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019. 2

[10] Marc Eder, Mykhailo Shvets, John Lim, and Jan-Michael Frahm. Tangent images for mitigating spherical distortion. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12423–12431, 2020. 2

[11] Andrea Fusiello, Emanuele Trucco, and Alessandro Verri. A compact algorithm for rectification of stereo pairs. *Machine Vision and Applications*, 12(1):16–22, 2000. 4

[12] Adrien Gaidon, Qiao Wang, Yohann Cabon, and Eleonora Vig. Virtualworlds as proxy for multi-object tracking analysis. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4340–4349, 2016. 6

[13] Wenliang Gao and Shaojie Shen. Dual-fisheye omnidirectional stereo. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6715–6722, 2017. 1, 2

[14] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, 2012. 6

[15] Heiko Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):328–341, 2008. 2

[16] Christian Häne, Lionel Heng, Gim Hee Lee, Alexey Sizov, and Marc Pollefeys. Real-time direct dense matching on fisheye images using plane-sweeping stereo. In *2014 2nd International Conference on 3D Vision*, volume 1, pages 57–64, 2014. 2

[17] Carlos Jaramillo, Roberto G. Valenti, Ling Guo, and Jizhong Xiao. Design and analysis of a single—camera omnistereo sensor for quadrotor micro aerial vehicles (mavs). *Sensors*, 16(2), 2016. 1

[18] J. Kannala and S.S. Brandt. A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(8):1335–1340, 2006. 2

[19] Bogdan Khomutenko, Gaëtan Garcia, and Philippe Martinet. An enhanced unified camera model. *IEEE Robotics and Automation Letters*, 1(1):137–144, 2016. 2

[20] Laurent Kneip, Hongdong Li, and Yongduek Seo. Upnp: An optimal o(n) solution to the absolute pose problem with universal applicability. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 127–142. Springer International Publishing, 2014. 4

[21] Ren Komatsu, Hiromitsu Fujii, Yusuke Tamura, Atsushi Yamashita, and Hajime Asama. 360° depth estimation from multiple fisheye images with origami crown representation of icosahedron. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10092–10099, 2020. 2, 6, 7

[22] 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, 2022. 3, 6, 7

[23] Weiming Li and Y. F. Li. Single-camera panoramic stereo imaging system with a fisheye lens and a convex mirror. *Opt. Express*, 19(7):5855–5867, Mar 2011. 1

[24] 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, 2021. 3, 6, 7

[25] Bin Lv, Hao Xu, Jianqing Wu, Yuan Tian, Sheng Tian, and Suoyao Feng. Revolution and rotation-based method for roadside lidar data integration. *Optics & Laser Technology*, 119:105571, 2019. 1

[26] Nikolaus Mayer, Eddy Ilg, Philip Häusser, 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 *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4040–4048, 2016. 6

[27] Christopher Mei and Patrick Rives. Single view point omnidirectional camera calibration from planar grids. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 3945–3950, 2007. 2, 4

[28] M. Menze, C. Heipke, and A. Geiger. Joint 3d estimation of vehicles and scene flow. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-3/W5:427–434, 2015. 6, 7

[29] Andreas Meuleman, Hyeonjoong Jang, Daniel S. Jeon, and Min H. Kim. Real-time sphere sweeping stereo from multiview fisheye images. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11418–11427, 2021. 2, 6, 7

[30] Davide Scaramuzza, Agostino Martinelli, and Roland Siegwart. A toolbox for easily calibrating omnidirectional cameras. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5695–5701, 2006. 2, 7

[31] Johannes Schneider, Cyrill Stachniss, and Wolfgang Förstner. On the accuracy of dense fisheye stereo. *IEEE Robotics and Automation Letters*, 1(1):227–234, 2016. 1

[32] Miriam Schönbein and Andreas Geiger. Omnidirectional 3d reconstruction in augmented manhattan worlds. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 716–723, 2014. 1

[33] Xiao Song, Xu Zhao, Hanwen Hu, and Liangji Fang. Edgestereo: A context integrated residual pyramid network for stereo matching. In C.V. Jawahar, Hongdong Li, Greg Mori, and Konrad Schindler, editors, *Computer Vision – ACCV 2018*, pages 20–35. Springer International Publishing, 2019. 7

[34] Yu-Chuan Su and Kristen Grauman. Kernel transformer networks for compact spherical convolution. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9434–9443, 2019. 2

[35] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. *ECCV*, 12347(1):402–419, 2020. 2, 3, 5, 6

[36] Vladyslav Usenko, Nikolaus Demmel, and Daniel Cremers. The double sphere camera model. In *2018 International Conference on 3D Vision (3DV)*, pages 552–560, 2018. 2, 7

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pages 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc. 2, 5, 6

[38] 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. 3

[39] Changhee Won, Jongbin Ryu, and Jongwoo Lim. Omnimvs: End-to-end learning for omnidirectional stereo matching. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8986–8995, 2019. 2, 6, 7

[40] Changhee Won, Jongbin Ryu, and Jongwoo Lim. Sweepnet: Wide-baseline omnidirectional depth estimation. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 6073–6079, 2019. 2, 6, 7, 8

[41] Changhee Won, Jongbin Ryu, and Jongwoo Lim. End-to-end learning for omnidirectional stereo matching with uncertainty prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):3850–3862, 2021. 2, 6, 7

[42] Changhee Won, Hochang Seok, Zhaopeng Cui, Marc Pollefeys, and Jongwoo Lim. Omnislam: Omnidirectional localization and dense mapping for wide-baseline multi-camera systems. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 559–566, 2020. 1

[43] Hao Xu, Yichen Zhang, Boyu Zhou, Luqi Wang, Xinjie Yao, Guotao Meng, and Shaojie Shen. Omni-swarm: A decentralized omnidirectional visual-inertial-uwb state estimation system for aerial swarm. *arXiv preprint arXiv:2103.04131*, 2021. 1, 2

[44] Feihu Zhang, Xiaojuan Qi, Ruigang Yang, Victor Prisacariu, Benjamin Wah, and Philip Torr. Domain-invariant stereo matching networks. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 420–439. Springer International Publishing, 2020. 6