Estimating Depth of Monocular Panoramic Image with Teacher-Student Model Fusing Equirectangular and Spherical Representations

Jingguo Liu[1], Yijun Xu[1], Shigang Li[2], Jianfeng Li[1]*
[1]Southwest University, Chongqing, China
[2]Hiroshima City University, Hiroshima, Japan
popqlee@swu.edu.cn

May 28, 2024

ABSTRACT

Disconnectivity and distortion are the two problems which must be coped with when processing 360 degrees equirectangular images. In this paper, we propose a method of estimating the depth of monocular panoramic image with a teacher-student model fusing equirectangular and spherical representations. In contrast with the existing methods fusing an equirectangular representation with a cube map representation or tangent representation, a spherical representation is a better choice because a sampling on a sphere is more uniform and can also cope with distortion more effectively. In this processing, a novel spherical convolution kernel computing with sampling points on a sphere is developed to extract features from the spherical representation, and then, a Segmentation Feature Fusion(SFF) methodology is utilized to combine the features with ones extracted from the equirectangular representation. In contrast with the existing methods using a teacher-student model to obtain a lighter model of depth estimation, we use a teacher-student model to learn the latent features of depth images. This results in a trained model which estimates the depth map of an equirectangular image using not only the feature maps extracted from an input equirectangular image but also the distilled knowledge learnt from the ground truth of depth map of a training set. In experiments, the proposed method is tested on several well-known 360 monocular depth estimation benchmark datasets, and outperforms the existing methods for the most evaluation indexes.

1 Introduction

Wider field of view means richer visual information. Estimating the depth from a single 360°panoramic image is an interesting topic, and until now a lot of researches have reported on it [1, 2, 3, 4, 5, 6, 7]. Since a 360°panoramic image is usually represented as an Equi-Rectangular Projection(ERP)[8, 9], this problem is formulated as the estimation of depth from a single ERP image concretely.

However, when a 360°panoramic image is represented as an ERP image, the problems of disconnectivity and distortion arise. While the disconnectivity can be solved easily by padding the left side using the right side image, how to coping with the distortion is tricky. In the existing methods, combining a cubemap representation [10] with an ERP image is used cope with this problems [11, 12]. In comparison with the distortion increasing greatly as approaching to poles of an ERP image, a cubemap representation is made up of six square perspective images.

Although a cubemap representation of a 360°panoramic image can improve the distortion of an ERP image effectively, it has its own limitations. First, since a cubemap representation is made up of six square perspective images, padding operations is necessary when carrying out convolution on the boundaries of each perspective image. Next, theoretically,

---

*Corresponding author

a cubemap representation is not a ideal one for a 360°panoramic image to cope with image distortion because a perspective image has its own distortion.

Similarly, tangent representation[13] is proposed to use to cope with the distortion. Tangent representation represents the panoramic image with any number of perspective images. However, due to the large number of views, there is a significant amount of redundancy in many regions, and the fusion processing of these repetitive regions will introduce new issues[14].

It is known that an ideal representation for a 360°panoramic image is a spherical image because the distortion of a scene object does not change with its position on a sphere. This isotropic property of a spherical image makes it superior to other representations. Additionally, on a sphere the problem of disconnectivity is eliminated completely. In this paper, we estimate the depth of a monocular panoramic image by fusing a spherical representation with an ERP image. A spherical convolution method is also developed, which enables a spherical convolution is carried out on sphere directly. Moreover, the feature maps extracted by the spherical convolution is fused with those extracted from the ERP image to achieve better performance than the existing methods.

Additionally, the existing methods of estimating depth of monocular panoramic image use the known ground truth of depth map in loss function to update the parameters of neural network during the back-propagation process. On the other hand, three dimensional structure of environments has its own inherent characteristics, especially for indoor environment having ceilings, floors and walls.

Based on this idea, we design a teacher-student model to learn the inherent cues of depth images of training set.

In this paper, we train an encoder-decoder structure with depth image input and depth image output to extract the inherent characteristics of panoramic depth images first, and then using this pretrained model as the teacher model to supervise the student network learning. The experimental results show that the accuracy of depth estimation is improved.

To evaluate the proposed approach, we conducted experiments on the 3D60[15], Matterport3D[16], and Stanford2D3D[17] datasets. The results demonstrate that our method surpasses existing approaches on the Matterport3D[16] and Stanford2D3D[17] datasets and achieves competitive performance on the 3D60[15] dataset. In summary, the contributions of this paper are as follows:

- In contrast with the existing methods fusing an ERP representation with a cubemap representation or a tangent representation, a Segmentation Feature Fusion(SFF) methodology is designed to combine spherical representation with the equirectangular representation to improve the performance of depth estimation.

- To realize the spherical representation, we design a new spherical kernel to carry out spherical convolution on a sphere, which solves the problems of disconnectivity and distortion of an ERP image effectively.

- We propose an encoder-decoder network to exploit the inherent cues of depth images of training set and supervise the backbone network learning in a distilled knowledge way. Our proposed teacher-student model is different from the existing methods which only use depth map as ground truth in the loss function of the network output at training phase.

2 Related Work

2.1 Monocular 360 depth estimation

Monocular 360 depth estimation is an extension of monocular depth estimation that focuses on predicting depth information in a 360-degree panoramic view by utilizing a single image as input. For example, [6] explored the spherical view synthesis to learn monocular 360-degree depth via a self-supervised method. [7] builds a two-stage pipeline for omnidirectional monocular depth estimation. [5] predicts the depth directly on the spherical mesh without projection preprocessing and achieved a good results. To address the spherical distortion in ERP images, [1] employed deformable convolution to adapt the sampling grids in response to geometric distortions within panoramic images. Moreover, [3] adaptively combines convolution kernels with varying dilations to expand the receptive field.

[2] devised a distortion-aware deformable convolution filter for testing purposes, a filter that can be trained using conventional perspective images. Differently, [4] represents the scene as compact vertical slices of a sphere and predict depth with convolution layers. These methods have demonstrated the feasibility of applying convolution directly on ERP images to eliminate distortions.

Recently, there has been a growing interest in utilizing fusion-based approaches to cope with the distortion. [11] proposed to effectively combines the cubemap and ERP features from both the encoder and decoder stages. Furthermore,
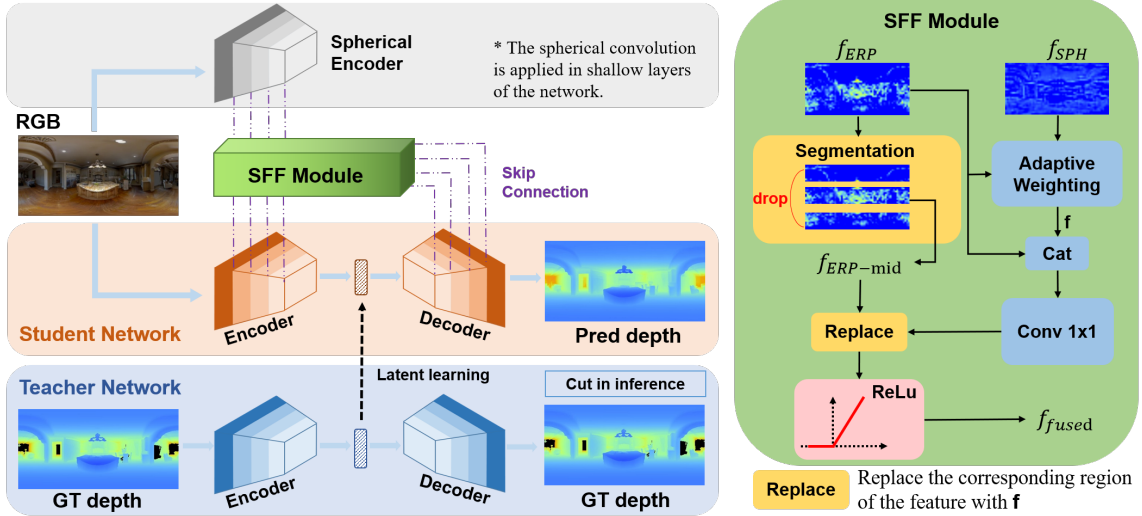
Figure 1: Overview of our network

[12] proposed a new framework for fusing features from different projections: ERP and Cubemap and demonstrate that the ERP features are more important for final ERP format depth prediction tasks. Differently, [18] designed a Cubemap Vision Transformers to extract distortion-free global features from the panorama and fuse them at multiple scales. For tangent patches based fusion methods, [19] proposed to estimate the depth from tangent patches and fuse the tangent patches to an ERP image. [20] introduce Local2Global Transformer, which aggregates local information within a panel and panel-wise global context. overhead. [21] introduced a panoramic transformer designed to exploit tangent patches within the spherical domain. [14] combined CNN and transformer to learn the holistic contextual information from the ERP and tangent patches and adopts a classification model for depth value prediction. [22] propose an equirectangular geometry-biased transformer.

In contrast with the closest researches[11, 12, 14], we propose a novel approach that fuses ERP and spherical representations. This integration can mitigate the defectives caused by ERP representation most effectively.

2.2 Spherical convolution

Spherical convolution is characterized by capturing and preserving the spatial information from panoramic images. Recently, [23] designed a Kernel Transformer to transfer the convolution kernels from perspective images to ERP images. [24] proposed to use spherical convolution to deal with the problem of weight sharing failure caused by video projection distortion. [25] employed spherical convolution to distill spatial-temporal 360 information. cite30 presented a spherical CNN that constructed by representing the sphere as a graph, and utilized the graph-based representation to define the standard CNN operations. These methods have provided evidence for the effectiveness of spherical convolutions in processing information from panoramic image. [26] design a distortion-aware Transformer to modulate ERP distortions continuously and self-adaptively. [27] proposed to utilizes a spherical polyhedron to represent omni-directional views to minimizes the variance of the spatial resolving power on the sphere surface.

2.3 Knowledge distillation

Knowledge distillation aims to enable the student model to mimic the behavior and performance of the teacher model. Knowledge distillation is first proposed by [28].

It is worth noting that [29] proposed that semantically similar inputs tend to elicit similar activation patterns in a trained network. Moreover, [30] demonstrated that knowledge distillation can be a powerful tool for reducing the size of large models without compromising their performance. These methods provide ample evidence of the effectiveness of knowledge distillation, in the field of deep learning. Moreover, some methods[31, 32, 33] have proved that the teacher-student model learning at the latent feature level is a feasible and effective approach.

In this paper we propose a network to exploit the inherent cues of depth images of training set and supervise the backbone network learning in a distilled knowledge way. Our proposed teacher-student model is different from the

existing methods[34, 35, 36] which only use depth map as ground truth in the loss function of the network output at training phase.

3 Proposed Methods

3.1 Overview

The proposed framework introduces a novel approach for monocular panoramic depth estimation. 1 shows the framework, which incorporates an ERP-based teacher-student model and employs spherical convolution for distortion elimination.

In our network, an ERP image serves as the input, and the predicted depth is output. The encoder utilizes a ConvNeXt-base pretrained model[37] to extract features from the input with channel numbers of [128, 256, 512, 1024]. Similarly, the spherical convolution encoder applies the proposed spherical convolution method to extract distortion-free high-dimensional features of corresponding sizes and channels in shallow networks. Besides, a skip connection structure (similar to [12])

is applied to enhance the interaction between the encoder and decoder and enrich the high-dimensional information of the image. Following an encoder-decoder architecture, the teacher network takes the ground truth depth image as input. In contrast to conventional depth estimation methods, our framework harnesses the benefits of knowledge distillation networks by employing a teacher network trained with ground truth to extract the inherent characteristics of the depth image. In decoder stage, the interpolation-based upsampling method is used to upsample the obtained features. Notably, We utilize a sub-pixel convolution[38] for final upsampling layers, which can minimize the impact of excessive manual factors on the results and enhance the spatial details.

3.2 Spherical convolution

3.2.1 Spherical kernel

One crucial aspect of performing convolution on a sphere is setting up the appropriate convolution kernel. Different with planar convolutions, convolution on a spherical surface possesses a distinctive characteristic: the kernels, whether rectangular or Gaussian, do not undergo translation but instead rotation on the sphere. Therefore, the problem of defective rotational invariance of convolution kernels on the sphere cannot be ignored. A sphere is inherently a perfectly axis-symmetric shape, and it appears as a circle from any viewpoint(See 2(a)). The existing methods can be classified as three approaches: 1. Using a conventional square kernel for the generated plane tangent to the central point of a spherical model. 2.using the points of a discrete spherical image originating from a geodesic dome. 3. network training for the offset of sampling points. Different from them, our sampling is directly carried out on a sphere, which eliminates the problems of disconnectivity and distortion in contrast to an ERP image representation, results in a more natural circular kernel in contrast to a square kernel applied to a tangent plane, and a relatively more uniform sampling in contrast to a discrete spherical image originated from a geodesic dome. And more reliable compared to methods that depend on network predictions. Inspired by conventional feature point detection[39], we introduced a circular convolution kernels. In contrast to computing rectangular convolution kernels from tangent planes[40][26], circular kernels align more closely with the essence of a sphere and have the ability to extend beyond image boundaries. Any point on the sphere can be considered as the center of an infinite number of circles. Therefore, we choose a point on the sphere and the closest outer circle around it as the convolution kernel(See 2(a)).

When performing convolutions on a sphere, it is necessary to take into account the curvature and topological structure of the sphere, which increases the complexity of the convolution process. With an increase in latitude spacing, the impact will become more pronounced. However, by selecting the closest outer circle as the convolution kernel, it can preserve the geometric properties of the image and minimizing the boundary effects. A circle encompasses infinite points, it is difficult for practical calculations. Considering that planar convolutions typically employ 3x3 convolution kernels, we select eight equidistant points on the circle, along with the central point, as the spherical convolution kernel, as depicted in 2(a). In contrast to traditional discrete spherical sampling methods, which may sacrifice local detail to ensure global coverage, our method independently computes the convolution kernel for each point based on its adjacent points. This approach eliminates the requirement for a global discrete grid, leading to higher precision and making it more suitable for pixel-level prediction tasks.

Computing the coordinates(x, y, z) of all pixels of an H×W ERP image projected onto a sphere, along with the corresponding coordinates on the outer circle, is a complex and time-consuming task. Therefore, we propose to define a basic spherical pattern, as illustrated in 2(a). Specifically, the outer circle is chosen as the basic spherical pattern at the North Pole of the sphere due to its unique geometric properties with coordinates (0, 0, 1).
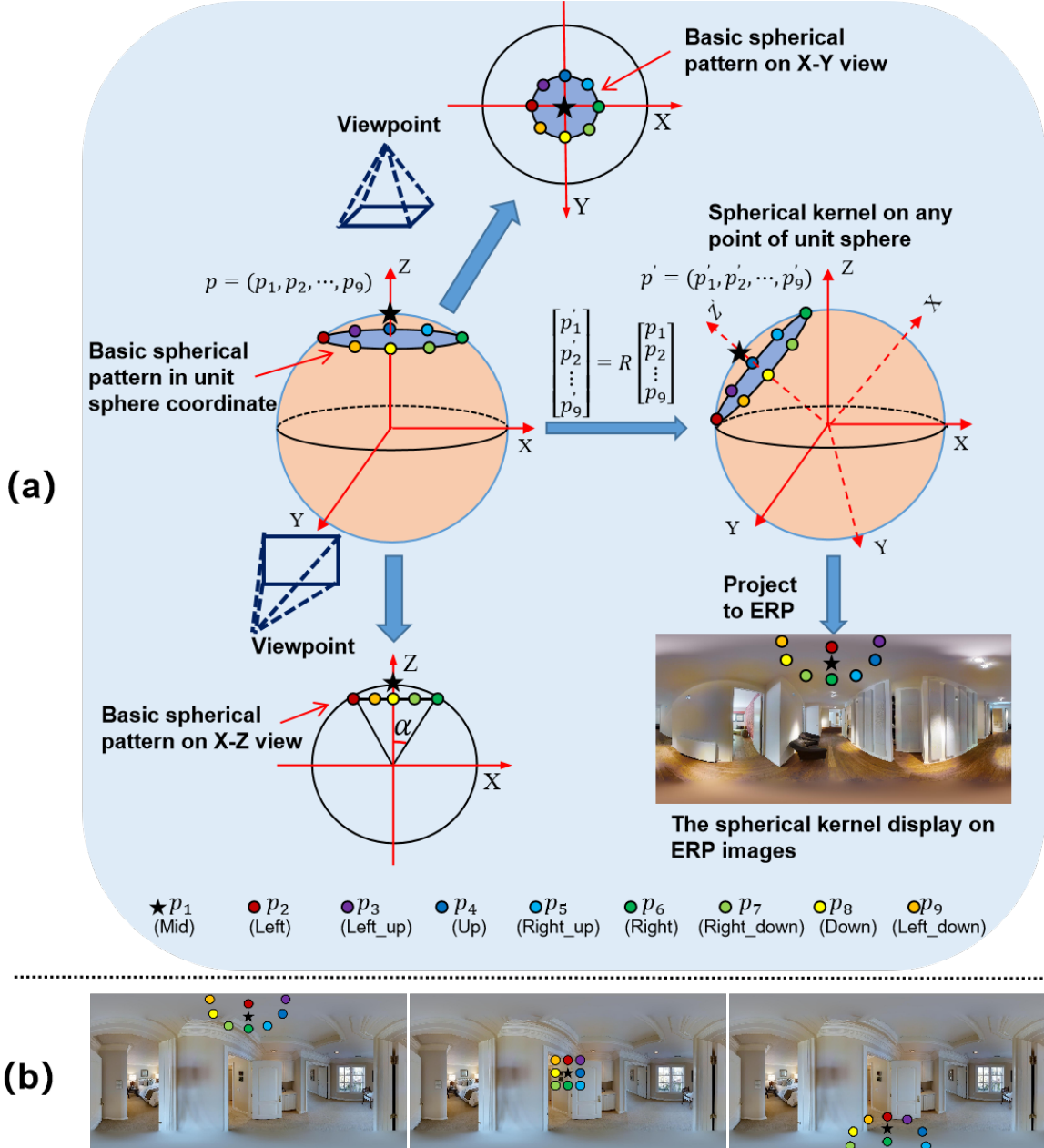
Figure 2: (a) Generation process of spherical convolution kernel. With a defined universal rotation matrix, spherical convolution kernels corresponding to different positions can be generated, which greatly reduces the computational cost. (b) Visualizing convolution kernels at the poles and equator positions in ERP images, which enable us to tackle distortion issues in distinct regions.

An ERP image with H×W is projected onto a unit sphere$(r = 1)$, the distance between any two adjacent points on the equator is $\frac{2\pi}{W}$. Given the uniqueness of the sphere and the aspect ratio of the ERP image being 1:2, it follows that the distance between any two points on any circle centered at the sphere's center is also $\frac{2\pi}{W}$. As shown in 2, in the X−Z view, let α denote the distance between any point on the circle and the Z-axis and α is $\frac{2\pi}{W}$. The coordinates of the basic

spherical pattern are as 1 shows:

$$
\begin{aligned}
p_1 &= (0, 0, 1) \\
p_2, p_6 &= (0.sin(\alpha)r, \pm cos(\alpha)r) \\
p_3, p_9 &= (\pm sin(\frac{\pi}{4})sin(\alpha)r, cos(\frac{\pi}{4})sin(\alpha)r, cos(\alpha)r) \\
p_4, p_8 &= (\pm sin(\alpha)r, 0, cos(\alpha)r) \\
p_5, p_7 &= (sin(\frac{\pi}{4})sin(\alpha)r, \pm cos(\frac{\pi}{4})sin(\alpha)r, cos(\alpha)r)
\end{aligned}
\tag{1}
$$

where p_1 denotes the North Pole point: Mid , while $p_2 \cdots p_9$ represents the points forming the base-spherical pattern $Left, Left_{up}, Up, Right_{up}, Right, Right_{down}, Down$ and $Left_{down}$ respectively.

As 2 and 2(a) shows, by applying a same procedure of rotating the basic spherical pattern with a consistent rotation matrix R, we can effectively reposition the pattern on the sphere through spherical rotations. The employment of consistent rotation facilitates the generation of the convolution kernel at different positions, while ensuring that the distribution of points on the outer circle adheres to the original distribution of the basic spherical pattern in sphere. As shown in 2(b), the proposed spherical convolution kernel takes on different shapes in different regions of the image.

$$
\begin{bmatrix} p_1' \\ p_2' \\ \vdots \\ p_9' \end{bmatrix} = R \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_9 \end{bmatrix}
\tag{2}
$$

Where $p_1', p_2' \cdots p_9'$ represents the nine points that make up the spherical kernel: $Mid', Left', Left_{up}', Up', Right_{up}',$ $Right', Right_{down}', Down', Left_{down}'$. Note that the process does not induce any deformation to the spherical kernel.

3.2.2 Rotated matrix computation

It is imperative to ensure a consistent rotation pattern is used when rotating from the North Pole point to a given point on the sphere. Following 3, a point(x, y, z)on the sphere can be represented by (θ, φ):

$$
\theta = arccos(z) , \varphi = arctan2(y, x)
\tag{3}
$$

where $\theta \in [0, \pi]$ denotes the inclination between the positive half-axis of the Z-axis and a specific point, while $\varphi \in [0, 2\pi)$ represents the azimuthal angle between the projection of the point on the X-Y plane and the positive half-axis of the X-axis. Based on θ and φ, we could infer the rotation matrix R for each point. The spherical convolution kernel at any point on the sphere can be obtained through 2.

We calculate the matrix R for rotation around various axes based on the values of θ and φ. As depicted in 4, φ is partitioned into four distinct categories to accommodate different scenarios. As shown in 3(a),(b), when φ is greater or less than π, the sphere rotates around fixed axes according to 4. That is, it first rotates around the x-axis by θ, then around the z-axis, and does not rotate around the y-axis. This allows the proposed pattern to be rotated to the desired position, while maintaining the relative positions of points on the spherical pattern unchanged. As shown in 3(c),(d), when φ equals π or 0, the rotation solely occurs around the y-axis. This approach ensures that all points on the spherical surface rotate according to the same proposed rotation process.

$$
\begin{cases} yaw = \varphi - \frac{\pi}{2} \\ \quad pitch = 0 \\ \quad roll = -\theta \end{cases}, \varphi < \pi; \quad \begin{cases} yaw = 0 \\ pitch = -\theta, \varphi = \pi; \\ roll = 0 \end{cases}
$$
$$
\begin{cases} yaw = (\varphi - \pi) - \frac{\pi}{2} \\ \quad pitch = 0 \\ \quad roll = \theta \end{cases}, \varphi > \pi; \quad \begin{cases} yaw = 0 \\ pitch = \theta, \varphi = 0; \\ roll = 0 \end{cases}
\tag{4}
$$

where $yaw, pitch,$ and $roll$ denote the rotation angles around the Z-axis, Y-axis, and X-axis respectively. After obtaining the rotation angles, we can calculate the rotation matrices R_X, R_Y, R_Z for each direction, and the final
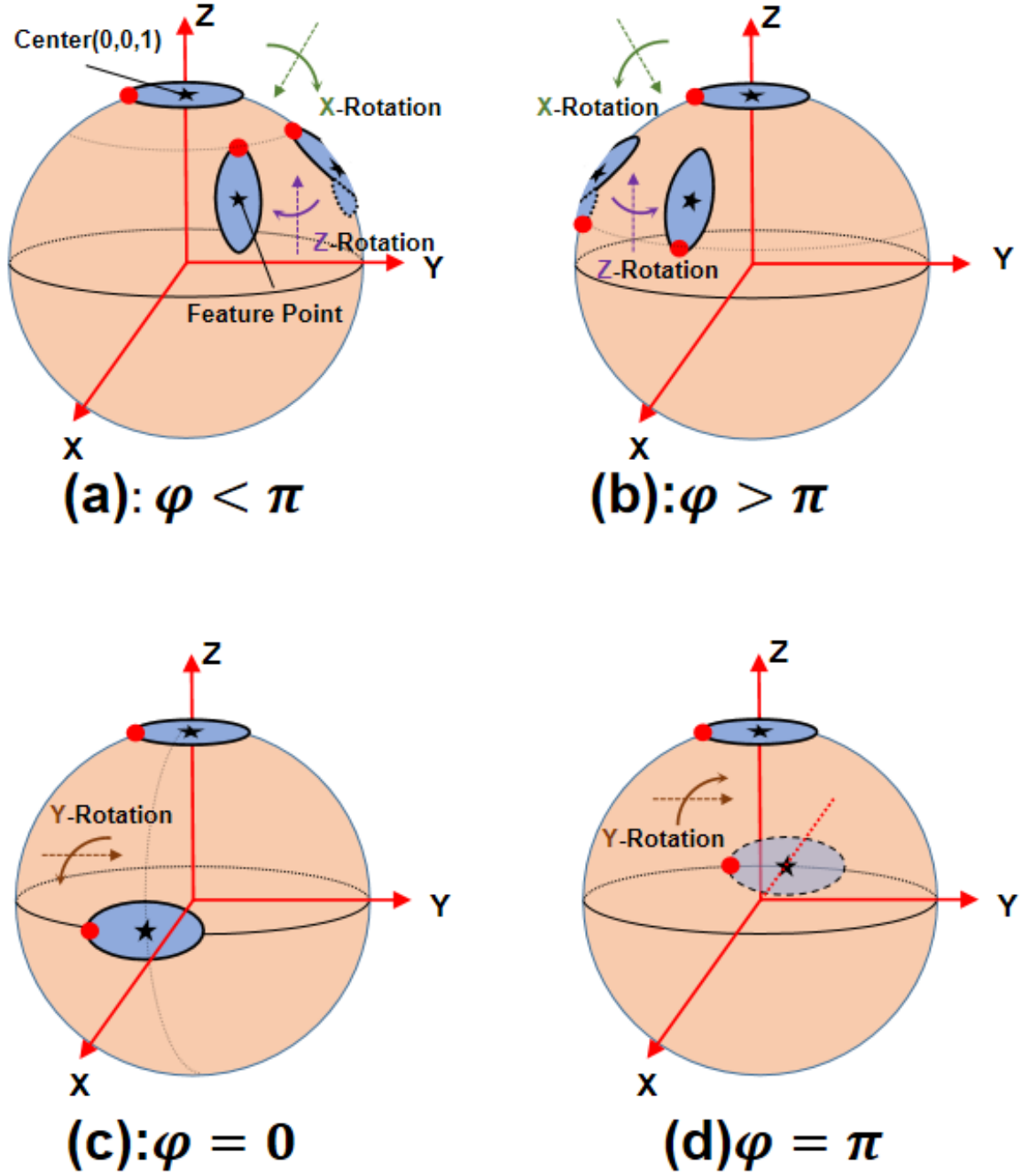
Figure 3: Rotation process

rotation matrix R are as shown in 6:

$$R_X = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(roll) & -sin(roll) \\ 0 & sin(roll) & cos(roll) \end{bmatrix};$$

$$R_Y = \begin{bmatrix} cos(pitch) & 0 & sin(pitch) \\ 0 & 1 & 0 \\ -sin(piych) & 0 & cos(pitch) \end{bmatrix};$$

$$R_Z = \begin{bmatrix} cos(yaw) & -sin(yaw) & 0 \\ sin(yaw) & cos(yaw) & 0 \\ 0 & 0 & 1 \end{bmatrix};$$

(5)

$$R = R_Z \bullet R_Y \bullet R_X \tag{6}$$

Utilizing R within the proposed basic spherical pattern enables the derivation of spherical convolution kernels that correspond to any position on the sphere.
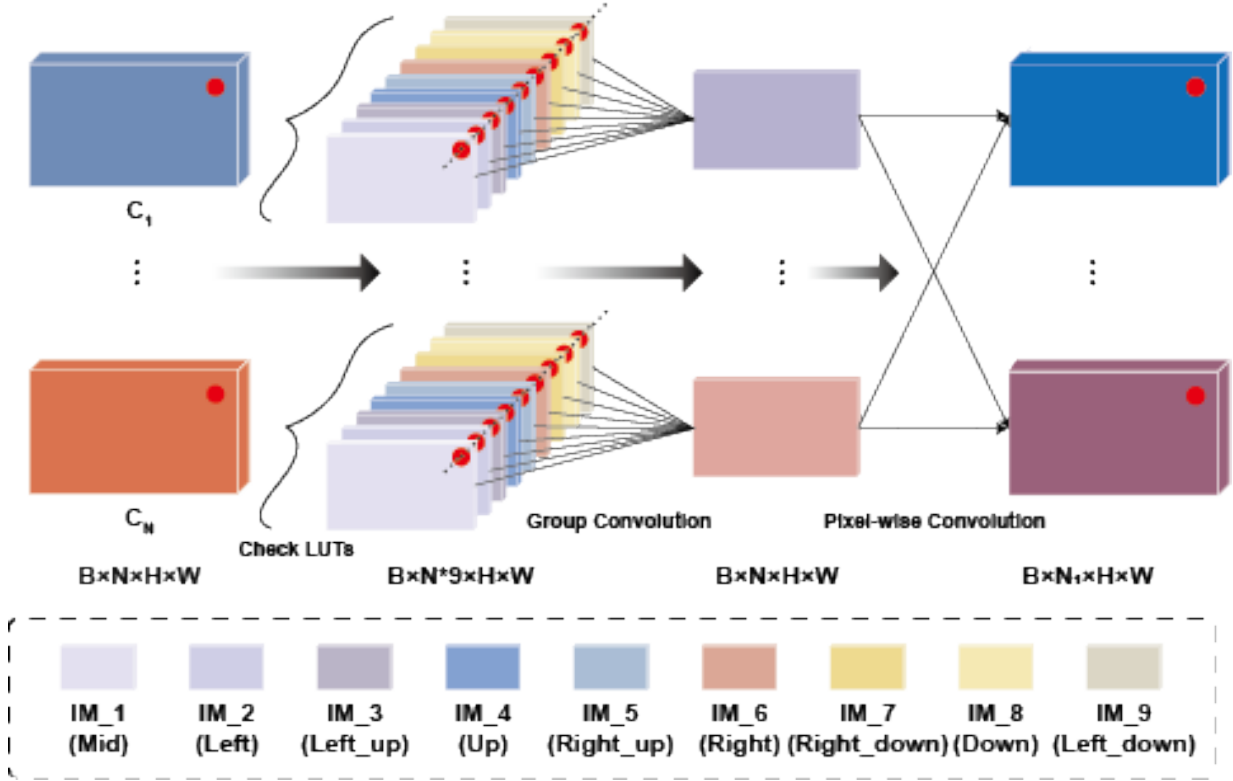

Figure 4: The relative position of the spherical convolution kernel for each pixel in the image is stored in the corresponding LUTs, which in turn maps the ERP image to nine sub-images. Then, after group convolution and pixel-wise convolution, an $R^{N_1 \times H \times W}$ feature map is obtained.

3.2.3 Separable spherical kernel convolution

For convolution, the network assigns unique weights to each channel's convolution kernel and accomplishes the convolution by moving these kernels over the image. However, on the sphere, convolution kernels are not translated but instead rotated. Previous methods employed grids to implement spherical convolution. While the grid-based approaches are constrained by the number of grids used, and may fail to achieve per-pixel division, which is detrimental for pixel-level tasks. To address this issue, we introduce a per-pixel separable spherical kernel convolution method. As shown in 4, We firstly maps spherical convolution kernels, centered at each pixel on the image, to the same position in different images. Subsequently, we conduct group convolution with a size of 1, where all pixels comprising the kernel in the convolution process are grouped together. This operation eliminates the need for additional padding to understand image boundaries. For the pixel-wise task, we believe that the introduced pixel-wise convolution strengthens the sensitivity for our network to inter dependencies among neighboring pixels on the sphere, enhancing the capacity of the network to perceive structural information in panoramic images.

specificly, we propose to use look-up tables(LUTs) to store the respective relative positions of the spherical convolution kernel at each point. For instance, LUT1 stores the positions of p_1 (i.e., the 'Mid' point) for each pixel of the image, which is the center of the proposed spherical convolution kernel, and LUT2 stores the relative positions of p_2 (i.e., the 'Left' point) in the spherical convolution kernel for each pixel of the image. With LUT2, we can obtain an image that is entirely composed of p_2 from a given image while maintaining the size of the original image. Similarly, LUT3 to LUT9 represent the positions of p_3 to p_9 in the spherical convolution kernel (see 4).

As depicted in 4, once the LUTs are available, a feature $R^{N \times H \times W}$ can be mapped to $R^{N*9 \times H \times W}$ sub-features from IM_1 to IM_9. Based on it, the group convolution with a kernel size of 1 can be employed to equivalently replace
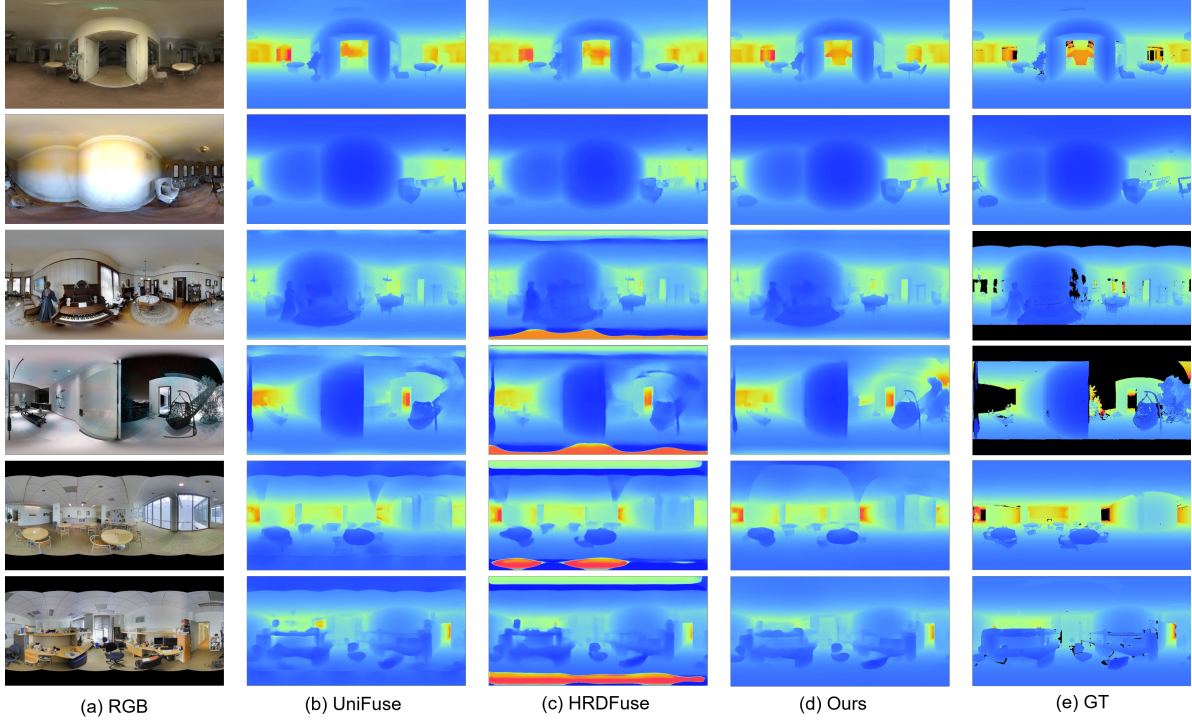
Figure 5: Results of qualitative comparison on 3D60 (top), Matterport3D (middle) and Stanford2D3D (bottom).

the original kernel convolution. Finally, a pixel-wise convolution is conducted to expand the channels in the spherical convolution, and a feature$\in R^{N_1 \times H \times W}$ (where N_1 can be arbitrarily set) can be obtained.

3.3 Segmentation feature fusion module

The feature f_{ERP} obtained through planar convolution exhibits distortion at the poles, while the feature f_{sph} obtained through the proposed spherical convolution method is distortion-free. The effectiveness and importance of planar convolution has been demonstrated in this task[12], research on the reliability of spherical convolution in deeper layers remains limited. To exploit the advantages of both convolutions, we propose to integrate f_{sph} into f_{ERP}. Given the minimal curvature near the equator in panoramic images, the distortion in this region is negligible. Therefore, planar convolution in this area is reasonable. In the panoramic image domain, researchers typically assume significant distortion in the upper and lower thirds, with almost no distortion in the central part. Leveraging the well-established method of planar convolution enables effective feature extraction from panoramic images, we preserve the features extracted through planar convolution near the equator. This strategy enables us to fully harness the benefits of planar convolution in extracting rich features from the image while avoiding the potential adverse impact of spherical convolution in deep layers. As shown in 1, we segment f_{ERP} into three equal parts and discard the features at the North and South poles, retaining the features in the middle part ($f_{ERP-mid}$). To fuse f_{ERP} and f_{sph}, we propose an adaptive weight fusion scheme, where we perform adaptive fusion on the two features to obtain an initial fused feature f:

$$f = w_0 \times f_{ERP} + w_1 \times f_{sph} \tag{7}$$

where w_0 and w_1 are learnable parameters. Then, we prioritize f_{ERP} as the primary carrier and perform a concatenation operation between f and f_{ERP}. Subsequently, a convolution layer is used to achieve fusion and retain the feature $f_{ERP-mid}$ extracted near the equator. Lastly, a simple non-linear activation is applied to obtain the final fused feature f_{fused}. The fused feature f_{fused} effectively combines the superior features extracted by $f_{ERP-mid}$ near the equator with the distortion-free features f_{sph}.

3.4 teacher network

The proposed teacher-student network, as depicted in 1, aims to incorporate more depth information into the network by utilizing ground truth depth and compensating for the shortcomings of spherical convolution in deep layers.

Datasets	Method	Abs Rel↓	Sq Rel↓	RMSE↓	RMSE(log)↓	δ_1↑	δ_2↑	δ_3↑
Standford2D3D	FCRN[41]	- / 0.1837	- / -	- / 0.5774	- / -	- / 0.7230	- / 0.9207	- / 0.9731
	BiFuse with fusion[11]	- / 0.1209	- / -	- / 0.4142	- / -	- / 0.8660	- / 0.9580	- / 0.9860
	UniFuse with fusion[12]	- / 0.1114	- / -	- / 0.3691	- / 0.2322	- / 0.8711	- / 0.9664	- / 0.9882
	OmniFusion (2-iter)[19]	0.0950 / -	0.0491 / -	0.3474 / -	0.1599 / -	0.8988 / -	0.9769 / -	0.9924 / -
	PanoFormer*[21]	- / 0.1131	- / 0.0723	- / 0.3557	- / 0.2454	- / 0.8808	- / 0.9623	- / 0.9855
	SphereDepth[5]	- / -	- / -	- / 0.4512	- / -	- / 0.8666	- / 0.9642	- / 0.9863
	PanelNet[20]	- / -	- / -	0.2933 / -	- / -	0.9242 / -	0.9796 / -	0.9915 / -
	HRDFuse[14]	0.0935 / -	0.0508 / -	0.3106 / -	0.1422 / -	0.9140 / -	0.9798 / -	0.9927 / -
	Ours	**0.0926 / 0.0940**	**0.0487 / 0.0541**	0.3058 / **0.3269**	**0.1396 / 0.1417**	0.9188 / **0.9143**	**0.9804 / 0.9808**	**0.9931 / 0.9921**
	Teacher Network	0.0086 / 0.0093	0.0013 / 0.0021	0.0608 / 0.0758	0.0214 / 0.0270	0.9983 / 0.9994	0.9997 / 0.9994	0.9999 / 0.9998
3D60	FCRN[41]	- / 0.0699	- / 0.2833	- / -	- / -	- / 0.9532	- / 0.9905	- / 0.9966
	BiFuse with fusionp[11]	- / 0.0615	- / -	- / 0.2440	- / -	- / 0.9699	- / 0.9927	- / 0.9969
	UniFuse with fusion[12]	- / 0.0466	- / -	- / 0.1968	- / 0.0725	- / 0.9835	- / 0.9965	- / 0.9987
	OmniFusion (2-iter)[19]	0.0430 / -	0.0114 / -	0.1808 / -	0.0735 / -	0.9859 / -	0.9969 / -	0.9989 / -
	ODE-CNN[23]	- / 0.0467	- / 0.0124	- / 0.1728	- / 0.0793	- / 0.9814	- / 0.9967	- / 0.9889
	SphereDepth[5]	- / 0.0550	- / 0.1145	- / 0.2364	- / -	- / 0.9743	- / 0.9944	- / 0.9978
	HRDFuse[14]	0.0358 / -	0.0100 / -	0.1555 / -	0.0592 / -	0.9894 / -	0.9973 / -	0.9990 / -
	Ours	0.0394 / **0.0379**	0.0101 / **0.0105**	0.1560 / **0.1687**	0.0604 / **0.0602**	**0.9897 / 0.9901**	**0.9975 / 0.9975**	**0.9990 / 0.9991**
	Teacher Network	0.0081 / 0.0051	0.0005 / 0.0004	0.0401 / 0.0380	0.0143 / 0.0054	0.9996 / 0.9996	0.9999 / 0.9999	0.9999 / 0.9999
Matterport3D	FCRN[41]	0.2409	-	0.6704	-	0.7703	0.9174	0.9617
	BiFuse with fusion[11]	0.2048	-	0.6259	-	0.8452	0.9319	0.9632
	UniFuse with fusion[12]	0.1063	-	0.4941	0.1613	0.8897	0.9623	0.9831
	OmniFusion (2-iter)*[19]	0.1007	0.0969	0.4435	0.1664	0.9143	0.9666	0.9844
	PanoFormer*[21]	0.0904	0.0764	0.4470	0.1650	0.8816	0.9661	0.9878
	SphereDepth[5]	-	-	0.5922	-	0.8620	0.9519	0.9770
	PanelNet[20]	-	-	0.4528	-	0.9123	0.9703	0.9856
	HRDFuse[14]	0.0967	0.0936	0.4433	0.1642	0.9162	0.9669	0.9844
	Ours	0.0941	**0.0723**	**0.4396**	**0.1402**	0.9110	**0.9712**	**0.9904**
	Teacher Network	0.0186	0.0049	0.1262	0.0162	0.9954	0.9991	0.9997

Table 1: Quantitative comparison with other methods. Bold indicates that our method performs the best. -/-: On the left side of /, it indicates that the dataset processing estimates a depth of 8 meters, while on the right side of /, it indicates that the dataset processing estimates a depth of 10 meters. *:It indicates that due to the absence of a pre-trained model, its metrics are derived from the latest SOTA model, [14].

The teacher network takes the ground truth depth as input, generating the latent features in the deepest layer, which acts as guidance for the student model. By leveraging the inherent characteristics of the teacher model, we can enrich the depth information contained in the latent features of the student model, thereby improving the network's performance in depth estimation. It is important to note that the teacher network is discarded during the final inference. During the training of the teacher model, we employ the commonly used Burhu loss[41] as the loss function for depth estimation tasks.

4 Experiments

4.1 Datesets, Metrics and Implimentation details

Datesets: In this paper, We conducted experiments on three benchmark datasets that are widely used for this tasks: 3D60[15], Matterport3D[16], and Stanford2D3D[17] datasets. Stanford2D3D and Matterport3Dare real-world datasets. While 3D60[15] is composed of two synthetic datasets: SUNCG[42] and SceneNet[43] and two real-world datasets: Stanford2D3D and Matterport3D. Note that there are some rendering issues[12] with the 3D60, and some anomalies may occur in this task.

Metrics: Following previous work[12, 14], we adopt standard evaluation metrics for evaluation: Absolute Relative Error (Abs Rel), Squared Relative Error (Sq Rel), Root Mean Squared Error (RMSE), Root Mean Squared Error in logarithmic space (RMSE(log)) and accuracy with a threshold δ_t, where $t \in \{1.25, 1.25^2, 1.25^3\}$.

Implimentation details: Our network was trained using the Adam optimizer, a batch size of 1, and a learning rate of 1×10^{-4} on a TITAN RTX 24G. We trained our model for only 30 epochs for Matterport3D, 3D60 and 20 epochs for Stanford2D3D. Moreover, we adopt augmentation techniques, random color adjustment, and left-right-flipping, random yaw rotation in the training phase.

4.2 comparision with state of the art

1 presents a comparative analysis between our method and existing methods for depth estimation. Notably, Some methods like [14] and [19] differ from conventional depth estimation methods in terms of data processing for the Stanford2D3D and 3D60 datasets. Specifically, the training data and testing data have a maximum depth of 8 meters for these two datasets, while traditional methods like [12] [41] and [5] have a maximum depth of 10 meters. In order to analyze the results more comprehensively and to adequately compare our method with other methods, we evaluated the two different depth estimation results(8m and 10m) for our method. For the Matterport3D dataset, all existing methods

Base	S-Conv	Teacher	SFF	Abs Rel↓	Sq Rel↓	RMSE↓	$\delta_1 \uparrow$
✓				0.1125	0.0599	0.3434	0.8870
✓	✓			0.1050	0.0564	0.3239	0.9066
✓	✓	✓		0.0968	0.0546	0.3124	0.9084
✓	✓		✓	0.0986	0.0507	0.3131	0.9156
✓	✓	✓	✓	**0.0926**	**0.0487**	**0.3058**	**0.9188**

Table 2: Ablation study for different combinations of independent components.

have the same maximum depth value of 10. Here we clarify that due to the unavailability of pre-trained models for some methods (e.g., Omnifuse does not provide a pre-trained model for the Matterport3D dataset, and Panoformer, PanelNet and HRDFuse does not provide any pre-trained models), for fair comparisons, we collected publicly available experimental data of competitors from the comparisons made by the latest SOTA depth estimation model HRDfuse.

As 1 shows, Our method performs well compared to SOTA methods[11, 12, 19, 14, 5, 21, 20] on several benchmark datasets. On the Stanford2D3D dataset, our method outperforms Unifuse by 17.5% (Abs Rel) and 12.91% (RMSE), outperforms Omnifuse by 2.59%(Abs Rel) and 13.6%(RMSE),our method outperforms Panlenet by 0.082% (δ_1) and 0.16% (δ_3), and outperforms HRDfuse by 0.972%(abs rel), 1.57% (RMSE) and 0.525% (δ_1). On the 3D60 dataset, our method outperforms Unifuse by 22.96% (abs rel) and 16.66% (RMSE), outperforms Omnifuse by 9.14% (abs rel), outperforms ODE-CNN by 23.22%(Abs Rel) and 2.43%(RMSE), and outperforms HRDfuse with 0.03% (δ_1), while also demonstrating competitive results on other metrics with HRDFuse. Furthermore, it is observed that the method introduced in this paper achieves a slightly superior of accuracy in comparison to HRDFuse. On the Matterport3D dataset, our method outperforms Unifuse by 18.38% (Abs Rel) and 12.37% (RMSE), outperforms Omnifuse by 7% (Abs Rel), outperforms PanelNet by 0.485% (δ_3) and over 3% (RMSE), and outperforms HRDfuse by 2.76% (Abs Rel), 29.46% (Sq Rel), 0.841% (RMSE) and 0.61%(δ_3). In 5, since HRDFuse does not provide any pre-trained models, we retrained the model to the official Settings for visualization, and we qualitatively compare our method with UniFuse and HRDFuse, and our method outperforms them.

4.3 ablation study

4.3.1 ablation study of each component

We conducted a series of incremental experiments to assess the effectiveness of each component, as illustrated in 2. The ablation experiment was performed with maximum depth of 8 meters on the Stanford2D3D. We used only the planar convolution method for depth estimation as the baseline. Subsequently, we added the proposed spherical convolution method, teacher network, and SFF module sequentially. As shown in 2, the performance of the planar convolution model was adversely affected by distortion. With the introducing of proposed the spherical convolution method, the performance improved by 6.21% (Sq Rel). However, we only used a simple concatenation method for fusion, which significantly reduced fusion effectiveness, while the performance has greatly improved by 11.24% (Sq Rel) since we utilized the SFF module. Moreover, we assessed the effectiveness of the teacher network by incorporating it into the network without the SFF module, resulting in a 3.3% (Sq Rel) improvement. Finally, when all components were used, the performance achieved the maximum improvement of 23.00%(Sq Rel). The experimental results illustrate that each proposed component plays a pivotal role in this task, notably elevating the overall performance of the network.

4.3.2 weight of fusion

We performed ablation experiments on the weights of SFF module, as presented in 3. We assigned fixed weight ratios of 1:0, 0:1, and 0.5:0.5, in addition to using adaptive weights. The experimental results demonstrate that the adaptive weights outperform the other three fixed weight ratios. Overall, the results provide further evidence of the effectiveness and reliability of the proposed SFF module.

ERP feature	Spherical feature	Abs Rel↓	Sq Rel↓	RMSE↓	$\delta_3 \uparrow$
0.5	0.5	0.1045	0.0535	0.3112	0.9930
0	1	0.0928	0.0510	0.3059	0.9927
1	0	0.1011	0.0533	0.3139	0.9929
Adaptive weighting		**0.0926**	**0.0487**	**0.3058**	**0.9931**

Table 3: The Ablation study on the weight of SFF module.

5 Conclusions and future work

In this paper, we propose a method of depth estimation of a monocular panoramic image. To the best of our knowledge, it is the first of fusing equirectangular and spherical representations so as to mitigate the effect of the disconnectivity and distortion of ERP images, and supervise the student network to learn the inherent cues of depth images of training set via a teacher-student model. The experiments shows the effectiveness of the proposed method. Since depth estimation is a basic technique for image understanding, we believe the proposed method can find a lot of applications, such as visual surveillance, robot navigation and so on. It is also our future work to do.

References

[1] Hong-Xiang Chen, Kunhong Li, Zhiheng Fu, Mengyi Liu, Zonghao Chen, and Yulan Guo. Distortion-aware monocular depth estimation for omnidirectional images. *IEEE Signal Processing Letters*, 28:334–338, 2021.

[2] Keisuke Tateno, Nassir Navab, and Federico Tombari. Distortion-aware convolutional filters for dense prediction in panoramic images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 707–722, 2018.

[3] Chuanqing Zhuang, Zhengda Lu, Yiqun Wang, Jun Xiao, and Ying Wang. Acdnet: Adaptively combined dilated convolution for monocular panorama depth estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 3653–3661, 2022.

[4] Giovanni Pintore, Marco Agus, Eva Almansa, Jens Schneider, and Enrico Gobbetti. Slicenet: deep dense depth estimation from a single indoor panorama using a slice-based representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11536–11545, 2021.

[5] Qingsong Yan, Qiang Wang, Kaiyong Zhao, Bo Li, Xiaowei Chu, and Fei Deng. Spheredepth: Panorama depth estimation from spherical domain. In *2022 International Conference on 3D Vision (3DV)*, pages 1–10. IEEE, 2022.

[6] Nikolaos Zioulis, Antonis Karakottas, Dimitrios Zarpalas, Federico Alvarez, and Petros Daras. Spherical view synthesis for self-supervised 360 depth estimation. In *2019 International Conference on 3D Vision (3DV)*, pages 690–699. IEEE, 2019.

[7] Yuyan Li, Zhixin Yan, Ye Duan, and Liu Ren. Panodepth: A two-stage approach for monocular omnidirectional depth estimation. In *2021 International Conference on 3D Vision (3DV)*, pages 648–658. IEEE, 2021.

[8] Yu-Chuan Su and Kristen Grauman. Learning spherical convolution for fast features from 360 imagery. *Advances in Neural Information Processing Systems*, 30, 2017.

[9] Shang-Ta Yang, Fu-En Wang, Chi-Han Peng, Peter Wonka, Min Sun, and Hung-Kuo Chu. Dula-net: A dual-projection network for estimating room layouts from a single rgb panorama. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3363–3372, 2019.

[10] Hsien-Tzu Cheng, Chun-Hung Chao, Jin-Dong Dong, Hao-Kai Wen, Tyng-Luh Liu, and Min Sun. Cube padding for weakly-supervised saliency prediction in 360 videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1420–1429, 2018.

[11] Fu-En Wang, Yu-Hsuan Yeh, Min Sun, Wei-Chen Chiu, and Yi-Hsuan Tsai. Bifuse: Monocular 360 depth estimation via bi-projection fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 462–471, 2020.

[12] Hualie Jiang, Zhe Sheng, Siyu Zhu, Zilong Dong, and Rui Huang. Unifuse: Unidirectional fusion for 360 panorama depth estimation. *IEEE Robotics and Automation Letters*, 6(2):1519–1526, 2021.

[13] Marc Eder, Mykhailo Shvets, John Lim, and Jan-Michael Frahm. Tangent images for mitigating spherical distortion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12426–12434, 2020.

[14] Hao Ai, Zidong Cao, Yan-Pei Cao, Ying Shan, and Lin Wang. Hrdfuse: Monocular 360deg depth estimation by collaboratively learning holistic-with-regional depth distributions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13273–13282, 2023.

[15] Nikolaos Zioulis, Antonis Karakottas, Dimitrios Zarpalas, and Petros Daras. Omnidepth: Dense depth estimation for indoors spherical panoramas. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 448–465, 2018.

[16] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*, 2017.

[17] Iro Armeni, Sasha Sax, Amir R Zamir, and Silvio Savarese. Joint 2d-3d-semantic data for indoor scene understanding. *arXiv preprint arXiv:1702.01105*, 2017.

[18] Jiayang Bai, Shuichang Lai, Haoyu Qin, Jie Guo, and Yanwen Guo. Glpanodepth: Global-to-local panoramic depth estimation. *arXiv preprint arXiv:2202.02796*, 2022.

[19] Yuyan Li, Yuliang Guo, Zhixin Yan, Xinyu Huang, Ye Duan, and Liu Ren. Omnifusion: 360 monocular depth estimation via geometry-aware fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2801–2810, 2022.

[20] Haozheng Yu, Lu He, Bing Jian, Weiwei Feng, and Shan Liu. Panelnet: Understanding 360 indoor environment via panel representation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 878–887, 2023.

[21] Zhijie Shen, Chunyu Lin, Kang Liao, Lang Nie, Zishuo Zheng, and Yao Zhao. Panoformer: Panorama transformer for indoor 360 depth estimation. In *European Conference on Computer Vision*, pages 195–211. Springer, 2022.

[22] Ilwi Yun, Chanyong Shin, Hyunku Lee, Hyuk-Jae Lee, and Chae Eun Rhee. Egformer: Equirectangular geometry-biased transformer for 360 depth estimation. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6078–6089, 2023.

[23] Ming Li, Xuejiao Hu, Jingzhao Dai, Yang Li, and Sidan Du. Omnidirectional stereo depth estimation based on spherical deep network. *Image and Vision Computing*, 114:104264, 2021.

[24] Jie Li, Ling Han, Chong Zhang, Qiyue Li, and Zhi Liu. Spherical convolution empowered viewport prediction in 360 video multicast with limited fov feedback. *ACM Transactions on Multimedia Computing, Communications and Applications*, 19(1):1–23, 2023.

[25] Chenglei Wu, Ruixiao Zhang, Zhi Wang, and Lifeng Sun. A spherical convolution approach for learning long term viewport prediction in 360 immersive video. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 14003–14040, 2020.

[26] Fanghua Yu, Xintao Wang, Mingdeng Cao, Gen Li, Ying Shan, and Chao Dong. Osrt: Omnidirectional image super-resolution with distortion-aware transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13283–13292, 2023.

[27] Yeonkun Lee, Jaeseok Jeong, Jongseob Yun, Wonjune Cho, and Kuk-Jin Yoon. Spherephd: Applying cnns on a spherical polyhedron representation of 360deg images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9181–9189, 2019.

[28] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[29] Frederick Tung and Greg Mori. Similarity-preserving knowledge distillation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1365–1374, 2019.

[30] Lucas Beyer, Xiaohua Zhai, Amélie Royer, Larisa Markeeva, Rohan Anil, and Alexander Kolesnikov. Knowledge distillation: A good teacher is patient and consistent. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10925–10934, 2022.

[31] Zhendong Yang, Zhe Li, Xiaohu Jiang, Yuan Gong, Zehuan Yuan, Danpei Zhao, and Chun Yuan. Focal and global knowledge distillation for detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4643–4652, 2022.

[32] Linfeng Zhang, Xin Chen, Xiaobing Tu, Pengfei Wan, Ning Xu, and Kaisheng Ma. Wavelet knowledge distillation: Towards efficient image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12464–12474, 2022.

[33] Sungsoo Ahn, Shell Xu Hu, Andreas Damianou, Neil D Lawrence, and Zhenwen Dai. Variational information distillation for knowledge transfer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9163–9171, 2019.

[34] Wenze Hu, Xue Dong, Ning Liu, and Yuanfeng Chen. Lumde: Light-weight unsupervised monocular depth estimation via knowledge distillation. *Applied Sciences*, 12(24):12593, 2022.

[35] Junjie Hu, Chenyou Fan, Hualie Jiang, Xiyue Guo, Yuan Gao, Xiangyong Lu, and Tin Lun Lam. Boosting lightweight depth estimation via knowledge distillation. In *International Conference on Knowledge Science, Engineering and Management*, pages 27–39. Springer, 2023.

[36] Yiran Wang, Xingyi Li, Min Shi, Ke Xian, and Zhiguo Cao. Knowledge distillation for fast and accurate monocular depth estimation on mobile devices. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2457–2465, 2021.

[37] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[38] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P. Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1874–1883, 2016.

[39] Jianfeng Li, Shigang Li, Tong Chen, and Yiguang Liu. Tracking on full-view image for camera motion estimation based on spherical model. In *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 569–574. IEEE, 2017.

[40] Benjamin Coors, Alexandru Paul Condurache, and Andreas Geiger. Spherenet: Learning spherical representations for detection and classification in omnidirectional images. In *Proceedings of the European conference on computer vision (ECCV)*, pages 518–533, 2018.

[41] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *3D Vision (3DV), 2016 Fourth International Conference on*, pages 239–248. IEEE, 2016.

[42] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1746–1754, 2017.

[43] Ankur Handa, Viorica Pătrăucean, Simon Stent, and Roberto Cipolla. Scenenet: An annotated model generator for indoor scene understanding. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5737–5743. IEEE, 2016.