Spherical Space Feature Decomposition for Guided Depth Map Super-Resolution

Zixiang Zhao[1,2] Jiangshe Zhang[1*] Xiang Gu[1] Chengli Tan[1]
Shuang Xu[3] Yulun Zhang[2] Radu Timofte[2,4] Luc Van Gool[2]

[1]Xi'an Jiaotong University [2]Computer Vision Lab, ETH Zürich
[3]Northwestern Polytechnical University [4]University of Würzburg

zixiangzhao@stu.xjtu.edu.cn, jszhang@mail.xjtu.edu.cn

Abstract

Guided depth map super-resolution (GDSR), as a hot topic in multi-modal image processing, aims to upsample low-resolution (LR) depth maps with additional information involved in high-resolution (HR) RGB images from the same scene. The critical step of this task is to effectively extract domain-shared and domain-private RGB/depth features. In addition, three detailed issues, namely blurry edges, noisy surfaces, and over-transferred RGB texture, need to be addressed. In this paper, we propose the Spherical Space feature Decomposition Network (SSDNet) to solve the above issues. To better model cross-modality features, Restormer block-based RGB/depth encoders are employed for extracting local-global features. Then, the extracted features are mapped to the spherical space to complete the separation of private features and the alignment of shared features. Shared features of RGB are fused with the depth features to complete the GDSR task. Subsequently, a spherical contrast refinement (SCR) module is proposed to further address the detail issues. Patches that are classified according to imperfect categories are input into the SCR module, where the patch features are pulled closer to the ground truth and pushed away from the corresponding imperfect samples in the spherical feature space via contrastive learning. Extensive experiments demonstrate that our method can achieve state-of-the-art results on four test datasets, as well as successfully generalize to real-world scenes. The code is available at https://github.com/Zhaozixiang1228/GDSR-SSDNet.

1. Introduction

Depth maps, as images that measure the distance of scene points from the sensor, are widely used in autonomous driving [38, 50, 87–89], pose estimation [75, 83], virtual reality [41, 51, 79, 80], and scene understanding [24, 53, 65, 96].

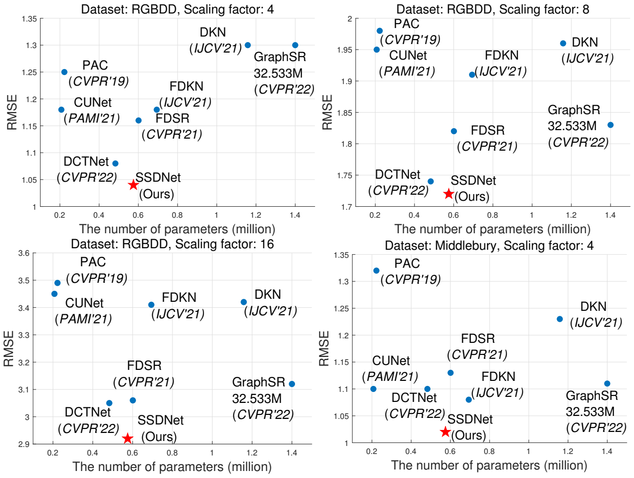

Figure 1: Our SSDNet achieves outstanding performance on the RGBDD dataset for ×4, ×8, and ×16 and Middlebury for ×4 while being computationally efficient.

However, the depth maps produced by current consumer-level depth sensors, *e.g.*, Time-of-Flight (ToF) and Kinect cameras, often have the disadvantages of low resolution and noise. These disadvantages are insufficient to meet the requirements of advanced computing vision tasks [52, 76, 93].

To obtain high-resolution depth maps, we naturally hope to accomplish depth map super-resolution (SR) by transferring mature SR models in the RGB image domain [13, 39, 86] to the depth SR task. However, a potential risk is that RGB SR models tend to focus on reconstructing high-frequency image information, such as details and texture. Conversely, for depth images, the objects' depth information is often textureless and piecewise, and more sensitive to unclear edges and noise. Therefore, it is unreasonable to directly apply RGB SR models to the depth SR task. On the other hand, while acquiring the depth map, it is relatively easy to obtain HR and noise-free RGB images in the same scene. Furthermore, there are statistical co-occurrences between the edges in RGB images and the discontinuities in

[*] Corresponding author.

depth maps [55]. Therefore, we hope to use the HR RGB images to provide edges and contour information missing in the depth map, and to fuse the multi-modal information to accomplish the LR depth image upsampling.

In the era of deep learning (DL), numerous methods have been utilized to learn the mapping between LR → HR depth maps. These methods succeeded to some extent in modeling the cross-modality features and reconstructing the contour and edge information. However, DL models which rely on natural priors will limit the model's flexibility. Models learning the LR → HR mapping via data-driven methods are difficult to interpret due to the unclear working mechanism [90]. Thus, effectively extracting and distinguishing domain-shared and domain-private RGB/depth features is still a challenge. While microscopically, the obtained depth images are often plagued by three detail issues: blurry edges, noisy surfaces, and over-transferred RGB texture, all of which affect the display quality of depth maps.

In response to addressing the above challenges, we expect to limit the solution space by constraining the extracted features and further improve the modeling of the dependencies between different modalities. Based on our observation, RGB images and depth maps contain shared features, such as depth map discontinuities and edge features of RGB objects, which can be aligned in the feature space. In addition, unique private features, such as the distance information of the depth map and the texture of RGB object surface, should be separated conversely. Thus, in the feature space, domain-shared and domain-private features are expected to separate and align respectively, while the distance between HR features and imperfect features with the above-mentioned detail issues should also be pushed away.

The above goal can be divided into two steps. First, extracting features through an effective encoder, and then selecting an appropriate distance measure to complete the alignment and separation of features. Considering that CNN-based architectures limit feature extraction capabilities due to content-independent convolution kernels and the lack of global information, we employ the Restormer blocks [84], which has been proved to effectively extract features in the low-level vision domain, as the basic unit of our encoder. For the choice of distance measure, we first think of the Euclidean distance, such as ℓ_1 or ℓ_2 distance. However, cross-modality features often contain different scales and orders of magnitude, and the Euclidean distance is easily affected by the scale, which makes it difficult to achieve our goal. Recently, with the development of spherical DL models [4, 7, 21, 22], the spherical feature learning is well known by virtue of its advantages over the Euclidean feature learning in many applications, *e.g*., domain adaptation [21]. Due to the distances between spherical features being regularized, the alignment and separation of features can be done more easily without losing the model's representational capacity.

According to the above analysis, we propose our *Spherical Space feature Decomposition Network* (SSDNet). The specific workflow is displayed in Fig. 2. Our contribution can be summarized in three-fold:

- We propose a spherical space feature decomposition framework to model the cross-modality features. The features extracted by the Restormer block are mapped to the spherical space for separation and alignment. This is the first time that the Transformer and spherical-space distance measure are applied to the GDSR task.
- The spherical contrast refinement module, cooperating with the imperfect patch classification and the corresponding contrastive learning branch, is proposed to address the possible detail issues in depth maps. This is also the first time that the contrastive learning technique has been used for the GDSR task.
- Experiments on four GDSR benchmarks and a real scene dataset demonstrate that our method can generate satisfactory HR depth maps in different scenarios and exhibits good generalization ability.

2. Related Work

In this section, we briefly introduce the GDSR task, and illustrate the Vision Transformer, spherical space DL and contrastive learning techniques utilized in SSDNet.

2.1. GDSR methods

Image super-resolution is a popular image processing and computer vision task with too many sub-categories, so we only discuss the GDSR task here. Conventional GDSR methods can be divided into local- [40, 43, 44, 46], global- [12, 17, 36, 48, 77] and learning-based [8, 19, 20, 73, 74] methods, *etc*. They extract cross-modality information by manually designed filters, optimizing equations, or sparse dictionary learning. With the rapid development of DL, GDSR is further promoted by the CNN-based methods [58, 60–62, 68, 81]. Generally, DL-based methods can be categorized in three groups, *i.e*., auto-encoder (AE) [23, 30, 35, 78, 90, 92], learnable filter [14, 31, 71], algorithm unfolding [10, 11, 54, 94] and unsupervised [9, 15] methods. AE-based methods learn cross-modality features via shared or private encoders and reconstruct HR depth maps with decoders. In the learnable filter group, the convolution kernels used to extract information are set to be context-dependent and spatially-variant, which improves the flexibility of the model. The algorithmic unfolding group establishes interpretable GDSR models by building a bridge between traditional optimization functions and deep learning. Unsupervised methods change the training paradigm that requires paired LR/HR images, thus solving the difficulty of obtaining paired training data.

2.2. Spherical deep learning

The works [4, 6, 7] introduce the geometrical neural network or the spherical convolution neural networks for analyzing the spherical signals that are rotation invariant. Gu *et al*. [22] propose the spherical multi-layer perceptron for aligning feature distributions of different domains in spherical feature space for domain adaptation. Our idea of spherical space feature decomposition is inspired by Gu *et al*. [22] that shows the advantages of spherical feature learning over the Euclidean feature learning for domain adaptation in the "high-level" image classification task. Different from [22], we tackle the "low-level" GDSR task by aligning/separating the shared/private features of paired RGB image and depth map, rather than aligning the feature distributions as in [22]. We further propose the spherical contrast refinement for tackling the issues of blurry edges, noisy surfaces, and over-transferred RGB texture for GDSR.

2.3. Transformer in vision

Transformer, proposed by Vaswani et al. [64], has now become a popular technology in computer vision. It has achieved dominance in classification [16, 42, 63], object detection [2, 95], segmentation [66, 91], *etc*. Simultaneously, it has also made progress in low-level vision [1, 18, 33, 67]. Chen *et al*. [5] proposed IPT based on the standard Transformer and multi-task learning. Liang *et al*. [37] proposed SwinIR whose core module inherits from Swin Transformer [42]. Recently, Restormer [84] improves the transformer block by the gated-Dconv network and multi-Dconv head transposed attention. By applying self-attention in the feature dimension, the ability of local-global representation learning is maintained while being more friendly to high-resolution input images.

2.4. Contrastive Learning

Contrastive learning, which aims to address the scarcity of source and target paired data and prioritize self-supervised representation learning [25, 28]. The primary objective is to bring samples close to their positive counterparts and push them away from negative ones within high-dimensional manifolds, and it seeks to increase the distance between samples belonging to different classes. Recently, contrastive learning has gained significant attention in computer vision, particularly in high-level tasks such as object detection [72], video segmentation [3], and object tracking [82]. It has also found applications in low-level tasks such as super-resolution [85], image translation [49], and image dehazing [70].

2.5. Comparison with existing approaches

The methods most related to our approach are feature-decomposition DL methods [10, 90], which propose the idea that cross-modality features contain the common and private information. In contrast, our method, for the first time, exploits the Transformer model's powerful ability to model globally dependent features. Meanwhile, the distance measure in spherical space is proved to decompose the cross-modality features more effectively than Euclidean distances. Additionally, instead of only focusing on optimizing the ℓ_2 loss between the reconstructed image and ground truth, we give classification-based fine-tuning for image patches under different imperfect conditions, thus further improving the performance of GDSR.

3. Method

In this section, we first present the specific architecture of our SSDNet. Then, comprehensive descriptions are provided for each module. Finally, the loss function and training details will also be addressed.

3.1. Overview

We can define the input LR depth map, RGB image and the HR depth map as $D_{LR} \in \mathbb{R}^{h \times w}$, $D_{HR} \in \mathbb{R}^{H \times W}$ and $R \in \mathbb{R}^{H \times W \times 3}$, where $\{H, W\}$ and $\{h, w\}$ are the height and width of input RGB and depth images, respectively. For our SSDNet network, it consists of five modules, namely, the *Encoders* for RGB and Depth images, the *Decoders* for RGB and Depth images, and the *Spherical Contrast Refinement* (SCR) module, which is denoted as $\mathcal{E}_{\mathcal{R}}(\cdot)$, $\mathcal{E}_{\mathcal{D}}(\cdot)$, $\mathcal{D}_{\mathcal{R}}(\cdot)$, $\mathcal{D}_{\mathcal{D}}(\cdot)$, and $\mathcal{S}(\cdot)$, respectively.

In general, as shown in Fig. 2, SSDNet utilizes $\mathcal{E}_{\mathcal{R}}$ and $\mathcal{E}_{\mathcal{D}}$ to extract features, which are then projected onto the spherical space for feature separation and alignment. Then, the depth features and shared RGB features are input into $\mathcal{D}_{\mathcal{R}}$ to obtain reconstructed depth maps, and RGB features are fed into $\mathcal{E}_{\mathcal{R}}$ to get reconstructed RGB images. Finally, the reconstructed depth maps are subsequently provided to the SCR module to refine details and output the final D_{HR}.

The basic unit we employed for feature extraction and image restoration is the Restormer block [84]. The reason we opted for the Restormer block in $\mathcal{E}_{\mathcal{R}}$, $\mathcal{E}_{\mathcal{D}}$, $\mathcal{D}_{\mathcal{R}}$ and $\mathcal{D}_{\mathcal{D}}$ is because it allows for the extraction of global features from high-resolution input images by utilizing self-attention across feature dimensions [84]. This approach enables the extraction of cross-modality shallow features without significantly increasing computational requirements. For details on the Restormer block architecture, please refer to the supplementary material or the original paper [84].

3.1.1 Spherical space transform

We first define the mapping based on Riemannian geometry between *Euclidean Space* feature maps and *Spherical Space* feature maps, *i.e.*, $\mathcal{LOG}(\cdot)$ and $\mathcal{EXP}(\cdot)$, which are employed in the feature decomposition. A schematic diagram for the

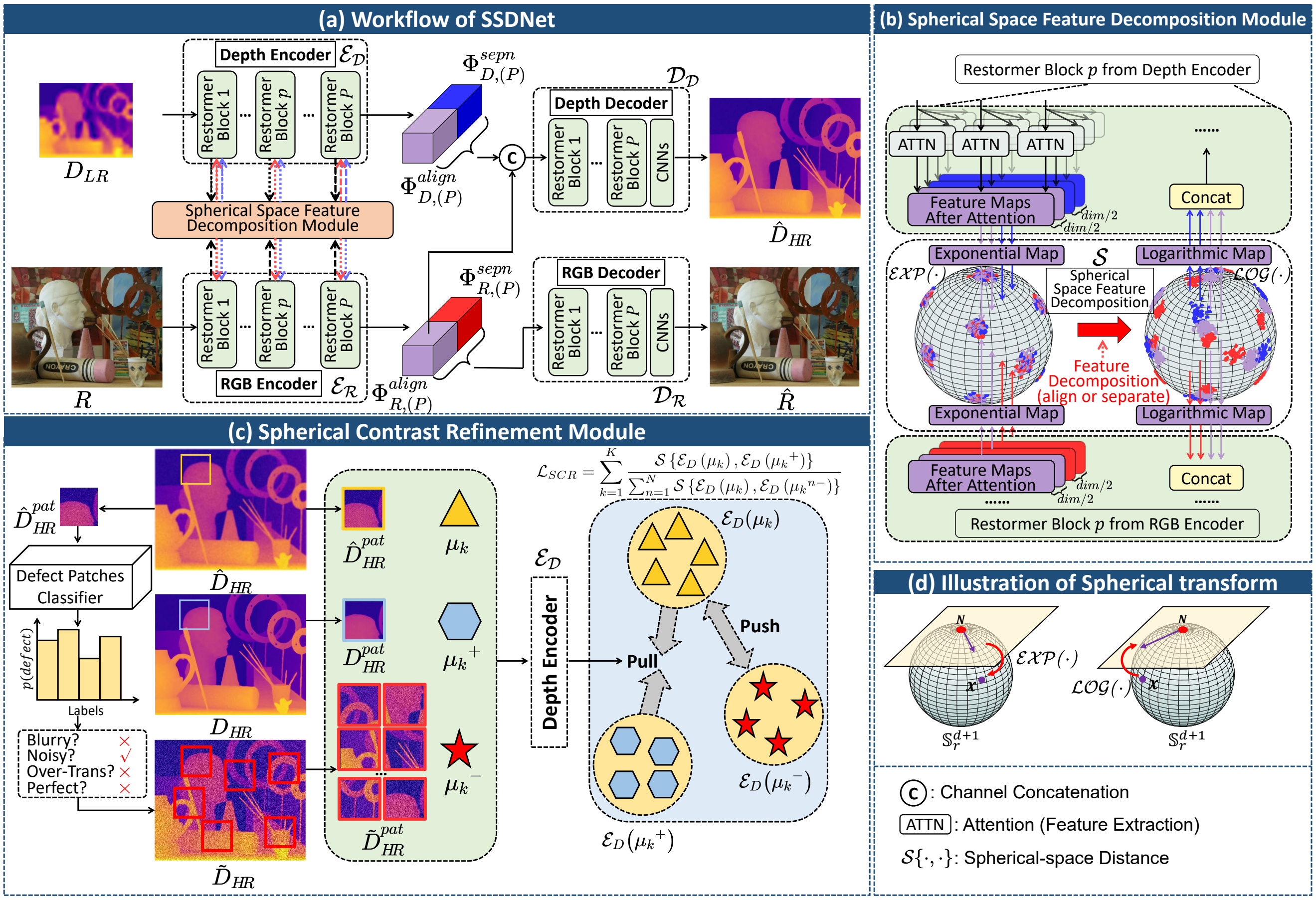
(a) Workflow of SSDNet
(b) Spherical Space Feature Decomposition Module
(c) Spherical Contrast Refinement Module
(d) Illustration of Spherical transform
D_{LR}
R
Depth Encoder $\mathcal{E}_{\mathcal{D}}$
RGB Encoder $\mathcal{E}_{\mathcal{R}}$
Restormer Block 1
Restormer Block p
Restormer Block P
Spherical Space Feature Decomposition Module
$\Phi^{sepn}_{D,(P)}$
$\Phi^{align}_{D,(P)}$
$\Phi^{sepn}_{R,(P)}$
$\Phi^{align}_{R,(P)}$
C
Depth Decoder $\mathcal{D}_{\mathcal{D}}$
RGB Decoder $\mathcal{D}_{\mathcal{R}}$
CNNs
$\hat{D}_{HR}$
$\hat{R}$
Restormer Block p from Depth Encoder
ATTN
Feature Maps After Attention
dim/2
Concat
Exponential Map
Logarithmic Map
$\mathcal{S}$
Spherical Space Feature Decomposition
$\mathcal{EXP}(\cdot)$
$\mathcal{LOG}(\cdot)$
Feature Decomposition (align or separate)
Restormer Block p from RGB Encoder
$\hat{D}^{pat}_{HR}$
Defect Patches Classifier
$p(defect)$
Labels
Blurry? ×
Noisy? ✓
Over-Trans? ×
Perfect? ×
D_{HR}
D^{pat}_{HR}
$\tilde{D}_{HR}$
$\tilde{D}^{pat}_{HR}$
μ_k
μ_k^{+}
μ_k^{-}
Depth Encoder
$\mathcal{E}_{\mathcal{D}}$
$\mathcal{L}_{SCR} = \sum_{k=1}^{K} \frac{\mathcal{S}\{\mathcal{E}_D(\mu_k), \mathcal{E}_D(\mu_k{}^{+})\}}{\sum_{n=1}^{N} \mathcal{S}\{\mathcal{E}_D(\mu_k), \mathcal{E}_D(\mu_k{}^{n-})\}}$
$\mathcal{E}_D(\mu_k)$
$\mathcal{E}_D(\mu_k{}^{+})$
$\mathcal{E}_D(\mu_k{}^{-})$
Pull
Push
N
x
$\mathbb{S}^{d+1}_r$
C: Channel Concatenation
ATTN: Attention (Feature Extraction)
$\mathcal{S}\{\cdot,\cdot\}$: Spherical-space Distance

Figure 2: **(a)** The Architecture of our SSDNet method. First, the LR depth map is input into $\mathcal{E}_{\mathcal{D}}$, and the RGB image is input into $\mathcal{E}_{\mathcal{R}}$. The intermediate feature maps are projected onto the spherical space to complete the feature decomposition, *i.e*., alignment and separation. The shared & private features of the depth map, and the shared features of the RGB image are input into the $\mathcal{D}_{\mathcal{D}}$ to reconstruct HR depth map. The RGB features are fed into $\mathcal{D}_{\mathcal{R}}$ to reconstruct RGB image. **(b)** Illustration of spherical space feature decomposition. **(c)** The module for spherical contrast refinement. **(d)** The mapping between Euclidean Space and Spherical Space.

mappings is shown in Fig. 2d. The distance measure on the spherical space is also defined.

Definition 1 (Spherical exponential mapping)**.** Given a vector v in d-dimensional Euclidean Space, we define the $(d+1)$-dimensional vector $\bar{v}$ by $\bar{v} = (v, r)$ where r is a hyperparameter named radius. Then the spherical exponential transform [69] $\exp_N : T_N\mathbb{S}^{d+1}_r \rightarrow \mathbb{S}^{d+1}_r$ is defined as

$$\exp_N(v) = N\cos\theta + \bar{v}\frac{\sin\theta}{\theta}, \tag{1}$$

where $\mathbb{S}^{d+1}_r = \{x \in \mathbb{R}^{d+1} : \|x\| = r\} \subset \mathbb{R}^{d+1}$ is the d-dimensional spherical space, $N = (0, \cdots, 0, r) \in \mathbb{S}^{d+1}_r$ is the north pole, $\theta = \frac{\|\bar{v}\|}{r}$, and $T_N\mathbb{S}^{d+1}_r = \{(v, r) : v \in \mathbb{R}^d\}$ is the tangent space of $\mathbb{S}^{d+1}_r$. Further, the spherical exponential mapping $\mathcal{EXP} : \mathbb{R}^{h'\times w'\times d} \rightarrow \mathbb{R}^{h'\times w'\times (d+1)}$ for an Euclidean feature map Φ is defined as

$$\mathcal{EXP}(\Phi)[i,j,:] = \exp_N(\Phi[i,j,:]) \tag{2}$$

where $\Phi[i,j,:]$ is the feature in location (i,j).

Definition 2 (Spherical logarithmic mapping)**.** Given a spherical feature $x \in \mathbb{S}^{d+1}_r$ with $\|x\| = r$, we define the spherical logarithmic transform [69] $\log_N : \mathbb{S}^{d+1}_r \rightarrow T_N\mathbb{S}^{d+1}_r$ by

$$\log_N(x) = \frac{\psi}{\sin\psi}(x - N\cos\psi), \tag{3}$$

where $N = (0, \cdots, 0, r) \in \mathbb{S}^{d+1}_r$ is the north pole, $\psi = \operatorname{argcos}(N^T x / r^2)$. Further, the spherical logarithmic mapping $\mathcal{LOG} : \mathbb{R}^{h'\times w'\times (d+1)} \rightarrow \mathbb{R}^{h'\times w'\times d}$ for a spherical feature map Φ is defined by

$$\mathcal{LOG}(\Phi)[i,j,:] = \mathcal{H}\left(\log_N(\Phi[i,j,:])\right), \tag{4}$$

where $\mathcal{H} : T_N\mathbb{S}^{d+1}_r \rightarrow \mathbb{R}^d$ is defined by $\mathcal{H}((v,r)) = v$.

Definition 3 (Spherical space distance)**.** Given two spherical feature maps $\Phi_1, \Phi_2 \in \mathbb{R}^{h'\times w'\times (d+1)}$ with $\|\Phi_1[i,j,:]\| =$

$\|\Phi_2[i,j,:]\| = r$ for any i, j, the spherical space distance of Φ_1 and Φ_2 is defined as

$$\mathcal{S}\left\{\Phi_1, \Phi_2\right\} = \sum_{i=1}^{h'}\sum_{j=1}^{w'} 1 - \frac{1}{r^2}\Phi_1[i,j,:]^T\Phi_2[i,j,:]. \quad (5)$$

3.1.2 Encoder

We use the feature extraction of D_{LR} as an example to explain the procedure, and R can be carried out similarly to D_{LR} by changing the subscripts from D to R. First, a 3×3 convolution is used to obtain shallow features embedding $\Phi_D^{(0)}$. The main body of feature extraction consists of a cascade of P Restormer blocks, and we denote the p-th Restormer block in $\mathcal{E}_\mathcal{D}$ as $\mathcal{R}_D^{(p)}$, where $p = 1, \cdots, P$. The input of each $\mathcal{R}_D^{(p)}$ is represented by $\Phi_D^{(p-1)}$.

At the p-th step of feature extraction, $\Phi_D^{(p-1)}$ passes through $\mathcal{R}_D^{(p)}$ to obtain the preliminary extracted feature $\tilde{\Phi}_D^{(p)}$. According to the analysis for motivation in Sec. 1, after the multi-head self-attention mechanism, $\tilde{\Phi}_D^{(p)}$ should contain shared features in some channels and private features in other channels. Thus, we assume that features in the first $\frac{dim}{2}$ channels are shared and represent cross-modality information, while features in the latter $\frac{dim}{2}$ channels are private and represent the characteristics of their own modality. To achieve feature separation and alignment, $\tilde{\Phi}_D^{(p)}$ is mapped to the spherical space using the *spherical exponential mapping* $\mathcal{EXP}(\cdot)$, and then recovered in the feature domain by the *spherical logarithmic mapping* $\mathcal{LOG}(\cdot)$ after calculating the feature decomposition loss. Finally, the recovered features are re-concatenated along the channel dimension to obtain $\Phi_D^{(p)}$, which will be input to the next $\mathcal{R}_D^{(p+1)}$. The feature decomposition loss will be illustrated in subsequent sections. The total feature extraction process of p-th step is:

$$\tilde{\Phi}_D^{(p)} = \mathcal{R}_D^{(p)}\left(\Phi_D^{(p-1)}\right) \quad (6a)$$

$$\tilde{\Phi}_{D,(p)}^{align} = \tilde{\Phi}_D^{(p)}\left[0:\tfrac{dim}{2}\right], \tilde{\Phi}_{D,(p)}^{sepn} = \tilde{\Phi}_D^{(p)}\left[\tfrac{dim}{2}:dim\right] \quad (6b)$$

$$\Phi_{D,(p)}^{align} = \mathcal{LOG}\left(\mathcal{EXP}\left(\tilde{\Phi}_{D,(p)}^{align}\right)\right) \quad (6c)$$

$$\Phi_{D,(p)}^{sepn} = \mathcal{LOG}\left(\mathcal{EXP}\left(\tilde{\Phi}_{D,(p)}^{sepn}\right)\right) \quad (6d)$$

$$\Phi_D^{(p)} = Cat\left(\Phi_{D,(p)}^{align}, \Phi_{D,(p)}^{sepn}\right) \quad (6e)$$

where $\{\tilde{\Phi}_{D,(p)}^{align}, \tilde{\Phi}_{D,(p)}^{sepn}\}$ and $\{\Phi_{D,(p)}^{align}, \Phi_{D,(p)}^{sepn}\}$ are the aligned and separated features after/before calculating the feature decomposition loss, and $Cat(\cdot,\cdot)$ is the channel concatenation operator.

Eventually, the entire encoder feature extraction can be regarded as:

$$\Phi_D = \mathcal{E}_D\left(D_{LR}\right), \ \Phi_R = \mathcal{E}_R\left(R\right), \quad (7)$$

where Φ_D and Φ_R are abbreviated forms of $\Phi_D^{(P)}$ and $\Phi_D^{(P)}$. Then Φ_D and Φ_R will be used as the input to the decoder.

3.1.3 Decoder

According to Sec. 3.1.2, we obtained features Φ_D and Φ_R, each containing shared information across $\frac{dim}{2}$ channels and private information across the other $\frac{dim}{2}$ channels. The features that we consider helpful for the HR depth reconstruction task are the full depth features Φ_D and the shared RGB features Φ_R^{sepn}. Therefore, we concatenate Φ_D and Φ_R^{sepn} in channel dimension and input them to $\mathcal{D}_\mathcal{D}$ to obtain the reconstructed depth image $\hat{D}_{HR}$, and input Φ_R to $\mathcal{D}_\mathcal{R}$ to obtain the reconstructed RGB image $\hat{R}$, which is:

$$\hat{D}_{HR} = \mathcal{D}_\mathcal{D}\left(Cat(\Phi_D, \Phi_R\left[0:\tfrac{dim}{2}\right])\right), \ \hat{R} = \mathcal{D}_\mathcal{R}(\Phi_R). \quad (8)$$

3.1.4 Spherical Contrast Refinement module

After decoding, we have obtained a preliminary estimation of the HR depth map $\hat{D}_{HR}$. However, it potentially has some minor issues, such as blurry edges, noisy surfaces and over-transferred RGB texture. Hence, in the *Spherical Contrast Refinement* (SCR) module, we target the imperfect issues in $\hat{D}_{HR}$ by contrastive learning, and the diagram is shown in Fig. 2c.

Defect patches classifier. Firstly, we artificially synthesize the "imperfect image dataset" using the training set. For example, for an $m \times m$ image patch from the ground truth in the training set, we can add random noise to make the patch *noisy*, apply Gaussian blur to it *blurry*, and add the same-location RGB image to make it *texture over-transferred*. Patches that are not processed can be regarded as *perfect*. In this way, we obtain a dataset with labels "noisy", "blurry", "texture over-transferred" and "perfect" which can be used to train a *defect patches classifier* (DPC) based on ResNet34 [26].

Positive and negative samples. After obtaining the decoder output $\hat{D}_{HR}$ in Eq. (8), we randomly crop it into an $m \times m$ patch $\hat{D}_{HR}^{pat}$, and input it into the well-trained DPC to obtain the imperfect type of $\hat{D}_{HR}^{pat}$. After getting the imperfect label, similar to the operation for making the "imperfect image dataset", we can transform the corresponding ground truth D_{HR} into an imperfect depth map $\tilde{D}_{HR}$. Then, we randomly crop N $m \times m$ patches from $\tilde{D}_{HR}$ that are different from the position of $\hat{D}_{HR}^{pat}$, and the corresponding imperfect patches $\tilde{D}_{HR}^{pat}$, named *negative samples*, are generated. Meanwhile, we refer to $\hat{D}_{HR}^{pat}$ as the *anchor* and the same-location ground truth D_{HR}^{pat} as the *positive sample*. The k-th positive, anchor and negative samples are represented by ${\mu_k}^+$, μ_k, and ${\mu_k}^-$, respectively.

Spherical contrast refinement. We input ${\mu_k}^+$, μ_k, and ${\mu_k}^-$ into $\mathcal{E}_D$ and get the contrastive refinement loss $\mathcal{L}_{SCR}$:

$$\mathcal{L}_{SCR} = \sum_{k=1}^{K} \frac{\mathcal{S}\left\{\mathcal{E}_D\left(\mu_k\right), \mathcal{E}_D\left({\mu_k}^+\right)\right\}}{\sum_{n=1}^{N}\mathcal{S}\left\{\mathcal{E}_D\left(\mu_k\right), \mathcal{E}_D\left({\mu_k}^{n-}\right)\right\}}, \quad (9)$$

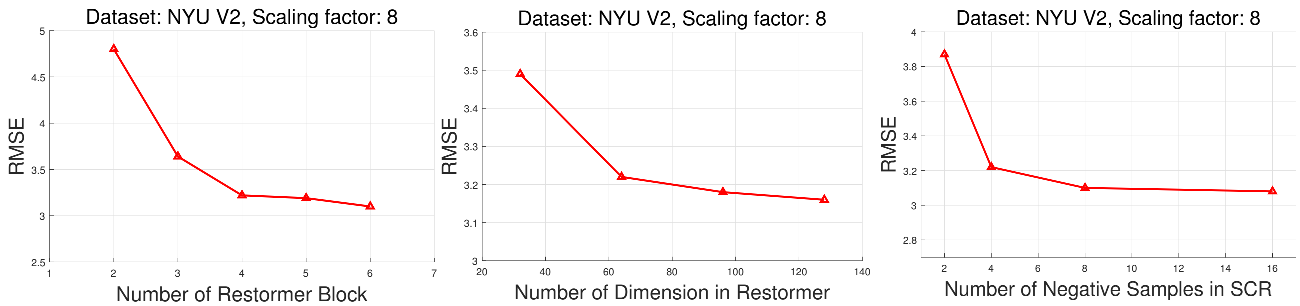

Figure 3: Exhibition for different settings of SSDNet. Results are evaluated on validation set of NYU V2 for GDSR × 8.

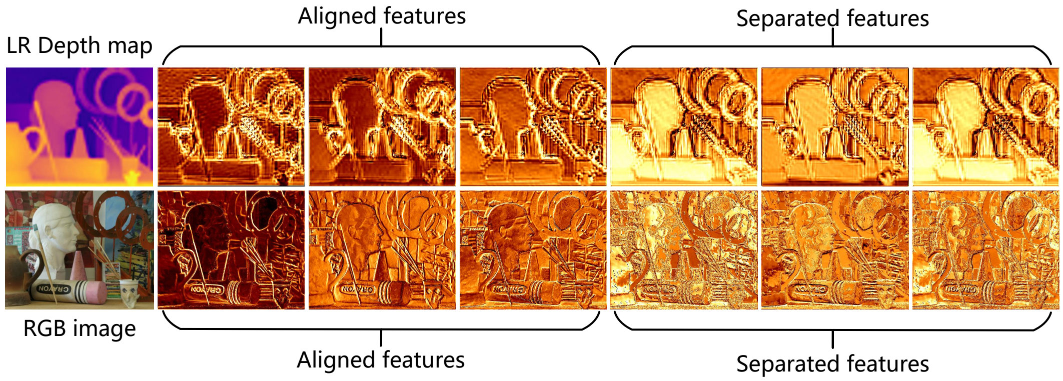

Figure 4: Visulization of the aligned and separated features.

where $\mathcal{S}\{\cdot,\cdot\}$ is the spherical space distance in Definition 3, and ${\mu_k}^{n-}$ is the nth negative sample in ${\mu_k}^{-}$. Through gradient descent, we minimize the distance between ${\mu_k}^{+}$ and μ_k features and maximize that between μ_k and ${\mu_k}^{-}$ features. This process fine-tunes $\mathcal{E}_D$. However, since the increased training cost for the SCR module, we incorporate it into the regular network training process every few iterations to strike a balance between training efficiency and effectiveness.

3.2. Training loss

The training loss in this paper comprises several components: the depth map reconstruction loss $\mathcal{L}_{pixel}^{D}$, the RGB reconstruction loss $\mathcal{L}_{pixel}^{R}$, the feature decomposition loss $\mathcal{L}_{dec}$, and the spherical contrastive refinement loss $\mathcal{L}_{SCR}$. We describe each loss separately next.

$\mathcal{L}_{pixel}^{D}$ ensures that the estimated depth map $\hat{D}_{HR}$ output by our SSDNet is close to the ground truth depth map D_{HR}. $\mathcal{L}_{pixel}^{R}$ ensures that the output $\hat{R}$ is close to the input R. Although theoretically unrelated to the depth map super-resolution task, this loss item is used to guarantee that the semantic information from the RGB image is involved in the shared RGB features, rather than simply generating random noise that is approximate to the depth features to meet the feature decomposition. Specifically,

$$\mathcal{L}_{pixel}^{D}=\sum_{k=1}^{K}\|\hat{D}_{HR}^{(k)}–D_{HR}^{(k)}\|_2^2,\ \mathcal{L}_{pixel}^{R}=\sum_{k=1}^{K}\|\hat{R}^{(k)}–R^{(k)}\|_2^2. \quad (10)$$

Regarding the feature decomposition loss $\mathcal{L}_{dec}$, we utilize the *spherical space distance* to enhance the similarity between shared features while reducing the similarity between separated features. We define the specific structure of the feature decomposition loss $\mathcal{L}_{dec}$ as follows:

$$\mathcal{L}_{dec}=\mathcal{L}_{align}-(1-\mathcal{L}_{sepn})^2, \quad (11)$$

where

$$\mathcal{L}_{sepn}=\sum_{p=1}^{P}\mathcal{S}\left\{\Phi_{D,(p)}^{sepn},\Phi_{R,(p)}^{sepn}\right\},\mathcal{L}_{align}=\sum_{p=1}^{P}\mathcal{S}\left\{\Phi_{D,(p)}^{align},\Phi_{R,(p)}^{align}\right\}. \quad (12)$$

Unlike ℓ_2 distance, the spherical distance captures relative differences without being influenced by scale. For 1D feature maps with three consecutive pixels: $f_1=[0.4,0.5,0.6]$, $f_2=[4,5,6]$, $f_3=[0.6,0.5,0.4]$, f_1 and f_2 exhibit similar pixel-wise increase structures and potentially similar extracted features, while f_3 differs. Thus, $\{f_1,f_2\}$ should be closer in distance and aligned, while $\{f_1,f_3\}$ should be separated. However, $\ell_2\{f_1,f_2\}=7.90$, $\ell_2\{f_1,f_3\}=0.28$ while $\mathcal{S}\{f_1,f_2\}=7.7\times10^{-8}$, $\mathcal{S}\{f_1,f_3\}=0.02$. Therefore, the spherical distance is a better choice.

When optimizing $\mathcal{L}_{dec}$, since the value range of spherical distance $\mathcal{S}\{\cdot,\cdot\}$ is $[0,2]$, we have $\mathcal{L}_{align}\rightarrow0$, $\mathcal{L}_{sepn}\rightarrow1$. According to Definition 3, the derivation gives: $\mathcal{C}\{\Phi_{D,(p)}^{align},\Phi_{R,(p)}^{align}\}\rightarrow1$ and $\mathcal{C}\{\Phi_{D,(p)}^{sepn},\Phi_{R,(p)}^{sepn}\}\rightarrow0$, where $\mathcal{C}$ is the cosine similarity (the last term of Eq. (5)). When

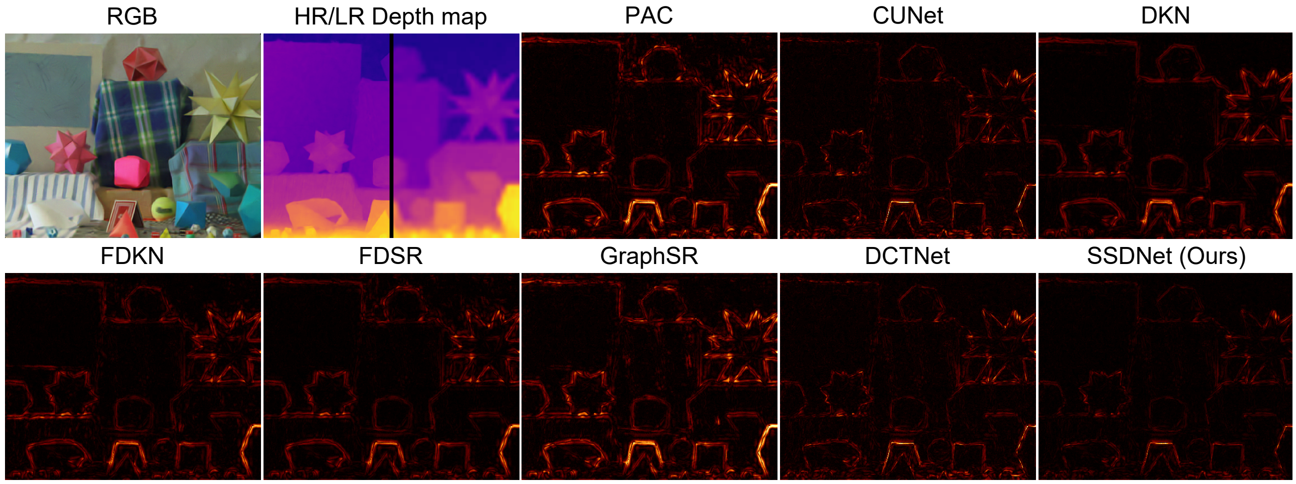

Figure 5: Visual error maps for test image in the Middlebury dataset for 8× super-resolution.

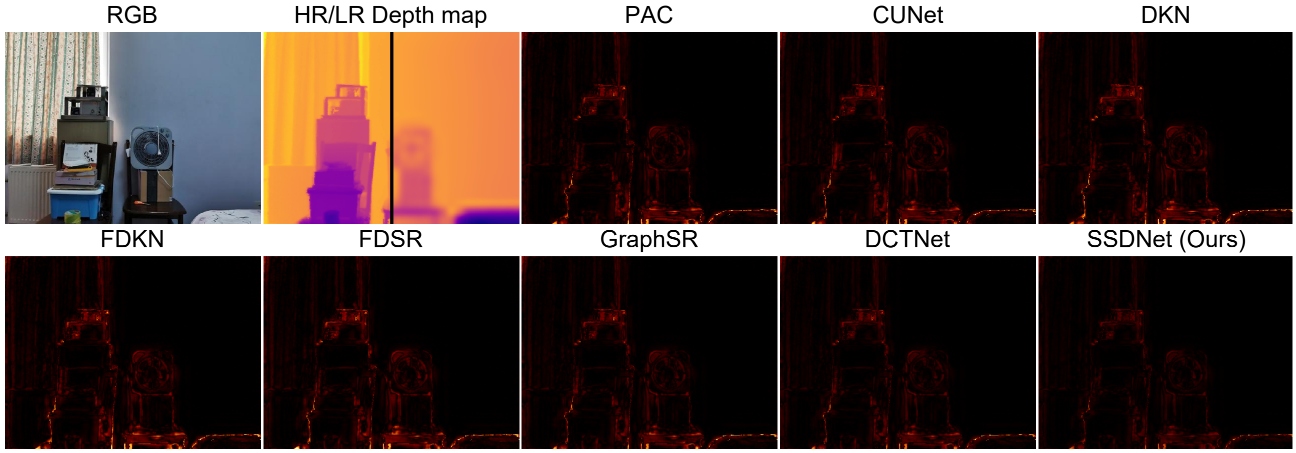

Figure 6: Visual error maps for test image in the RGBDD dataset for 16× super-resolution.

$\mathcal{C}=1$, vectors have an angle of 0, indicating higher similarity. If $\mathcal{C}=0$, the vectors are orthogonal, signifying no correlation. By forcing $\mathcal{C}=1$ or $\mathcal{C}=0$ for aligned/private features mapped to spherical space, we ensure the enhancement/reduction of similarity, which achieves our goal of feature decomposition.

By combining Eqs. (9) to (11), we obtain the total loss used for training, which can be expressed as follows:

$$\mathcal{L}_{total} = \mathcal{L}^{D}_{pixel} + \alpha_1 \mathcal{L}^{R}_{pixel} + \alpha_2 \mathcal{L}_{dec} + \alpha_3 \mathcal{L}_{SCR}. \quad (13)$$

Note that $\alpha_3 = 0$ if the current epoch does not require the spherical contrast refinement.

4. Experiment

This section will conduct a comprehensive set of experiments that aim to showcase the effectiveness of our model in addressing the GDSR task. Moreover, we will also provide evidence that substantiates the soundness of our SSDNet.

4.1. Setup

Datasets. Our experiments follow the protocol established in [27, 31, 90]. To evaluate our model, we employ four popular benchmarks: Middlebury [29, 56], Lu [45], NYU v2 [57], and RGBDD [27]. Our training and validation sets consist of the first 1000 images of NYU v2 dataset [57], divided into a 9:1 ratio. The last 449 pairs in NYU v2 are utilized as the test dataset. Middlebury [29, 56] (30 pairs), Lu [45] (6 pairs), and RGBDD [27] (405 pairs) are also used as test sets to evaluate the depth map SR ability across different scenes and objects. To furthermore demonstrate the generalization ability of our model to unknown scenarios, we incorporate the*real-world branch* of the RGBDD dataset, which comprises 2215/405 pairs of RGB-D images for training/test sets, respectively. Lastly, we use the root-mean-square error (RMSE) metric to evaluate the super-resolution effect of our model.

Implementation details. In our experiments, we apply bicu-

Methods	Middlebury [29, 56]			NYU V2 [57]			Lu [45]			RGBDD [27]		
	×4	×8	×16	×4	×8	×16	×4	×8	×16	×4	×8	×16
PAC [59]	1.32	2.62	4.58	1.89	3.33	6.78	1.20	2.33	5.19	1.25	1.98	3.49
CUNet [11]	1.10	2.17	4.33	1.92	3.70	6.78	0.91	2.23	4.99	1.18	1.95	3.45
DKN [31]	1.23	2.12	4.24	1.62	3.26	6.51	0.96	2.16	5.11	1.30	1.96	3.42
FDKN [31]	1.08	2.17	4.50	1.86	3.58	6.96	0.82	2.10	5.05	1.18	1.91	3.41
FDSR [27]	1.13	2.08	4.39	1.61	3.18	5.86	1.29	2.19	5.00	1.16	1.82	3.06
GraphSR [8]	1.11	2.12	4.43	1.79	3.17	6.02	0.92	2.05	5.15	1.30	1.83	3.12
DCTNet [90]	1.10	2.05	4.19	**1.59**	3.16	**5.84**	0.88	1.85	**4.39**	1.08	1.74	3.05
Ours	**1.02**	**1.91**	**4.02**	1.60	**3.14**	5.86	**0.80**	**1.82**	4.77	**1.04**	**1.72**	**2.92**

Table 1: Quantitative comparisons among the SOTA methods and our SSDNet in test datasets. The best and second best values are highlighted by **bold** and underline, respectively.

Dataset: RGBDD in real-world					
Methods	RMSE	Methods	RMSE	Methods	RMSE
SVLRM [47]	8.05	DJF [34]	7.90	DJFR [35]	8.01
DKN [31]	7.38	FDKN [31]	7.50	FDSR [27]	7.50
DCTNet [90]	7.37	SSDNet	**7.32**		
FDSR* [27]	5.49	DCTNet* [90]	5.43	SSDNet*	**5.38**

Table 2: Quantitative comparisons among the SOTA methods and SSDNet on *real-world branch* of RGBDD. The best and second best RMSEs are highlighted by **bold** and underline. Model* means that the model has been fine-tuned on the real-world training data.

Methods	PAC	CUN	DKN	FDKN	FDSR	GSR	DCT	Ours
Time (s)	0.20	0.23	0.21	0.04	0.01	0.92	0.08	0.10

Table 3: Results of time-consuming comparison for generating a 640× 480 HR depth map.

bic down-sampling to the HR depth maps to synthesize the corresponding LR depth maps. During the pre-processing stage, the training samples undergo resizing to 128×128. We train the network with a mini-batch size of 56 for 100 epochs, employing the Adam [32] optimizer with an initial learning rate of 5×10^{-3}. For training and testing, we employ a PC featuring eight NVIDIA GeForce RTX 3090 GPUs. α_1, α_2 and α_3 in Eq. (13) set to 10^{-2}, 10^{-3} and 10^{-2} in order to balance the magnitudes of each term in the loss. The SCR fine-tuning is performed once every 10 epochs of training. Further details are provided in the *supplementary material*.

4.2. Empirical validation experiment

Impact of network hyper-parameters. In our proposed SSDNet, the number of Restormer Blocks P, and number of dimensions in Restormer C, and the number of negative samples in SCR N are crucial factors in enhancing the super-resolution performance. We conduct experiments on the NYU validation set to study the impact of different combinations of P, C, N. Initially, we fix any two parameters at $C = 64$, $P = 4$, and $N = 8$, respectively, and evaluate the prediction quality of the unfixed parameter at $P = 2, 3, 4, 5, 6$, $C = 32, 64, 96, 128$, and $N = 2, 4, 8, 16$. The results are summarized in Fig. 3. We observe that the performance is limited when $P < 4$, and redundant parameter increases do not yield corresponding improvements when $P > 4$. Similarly, increasing C to more than 64 does not produce significant benefits but instead increased the training cost and computational expense. For the SCR module, the increase in training burden does not match the improvement in effectiveness when N exceeds 8. Therefore, to ensure a balance between performance and computational cost, we choose $P = 4, C = 64, N = 8$ for subsequent experiments.

Visualization of feature decomposition. Aligned and separated features are visualized in Fig. 4. Aligned features extract shared properties (edges/contours) from the depth/RGB image pair. Separated features capture modality-specific details (textures in RGB images, smooth depth distance in depth maps). The visualization aligns with our motivation.

4.3. Comparison with SOTA methods

This section aims to evaluate the performance of our SSDNet on several popular benchmarks in Sec. 4.1, and compare our results with the state-of-the-art methods, including PAC [59], CUNet [11], DKN [31], FDKN [31], FDSR [27], GraphSR [8], and DCTNet [90].

Qualitative Comparison. We present comparisons of the error maps with ground truth depth maps among SOTA methods in Figs. 5 and 6. Intuitively, the depth predictions of SSDNet exhibit lower prediction errors and are closer to the ground truth images, especially in terms of perception and restoration for the edges and contours. Further visual comparisons are shown in the supplementary material.

Quantitative results for traditional testsets. The quantitative results with scaling factors of 4, 8, and 16 on four test sets are shown in Tab. 1. Compared with existing methods that achieve good results on certain datasets or super-resolution factors, our SSDNet produces satisfactory predic-

	Encoder&Decoder	$\mathcal{L}_{dec}$	Form of $\mathcal{L}_{dec}$	$\mathcal{L}^{R}_{pixel}$	SCR Module	×4	×8	×16
Exp. I	Shared	√	$\mathcal{S}$	√	√	1.23	2.10	3.49
Exp. II	Private	×	$\mathcal{S}$	√	√	1.17	1.94	3.27
Exp. III	Private	√	ℓ_2	√	√	1.15	1.83	3.06
Exp. IV	Private	√	$\mathcal{S}$	×	√	1.16	1.89	3.29
Exp. V	Private	√	$\mathcal{S}$	√	×	1.18	1.87	3.36
Ours	Private	√	$\mathcal{S}$	√	√	**1.04**	**1.72**	**2.92**

Table 4: Results of ablation experiments on the RGBDD test set. **Bold** indicates the best score in terms of RMSE.

tions on multiple datasets and various super-resolution scales. This indicates that our model has a better super-resolution performance than previous methods.

Quantitative results for real-world branch. In addition, following [27, 90], we use the pre-trained 4× model to directly test on the *real-world branch* of the RGBDD dataset to explore the generalization ability in real-world scenarios. We conduct all testing without any additional fine-tuning, and the quantitative results are presented in Tab. 2. Furthermore, we also perform targeted training and testing on this dataset. Regardless of whether fine-tuning or not, SSDNet outperforms previous methods in RMSE, highlighting its powerful generalization ability in handling unknown scenes.

Parameter & running time comparison. We demonstrate the relationship between the number of model learnable parameters *vs*. RMSE on the RGBDD dataset for ×4, ×8, and ×16 and Middlebury for ×4 in Fig. 1. Our model exhibits a clear advantage over existing methods with relatively fewer parameters. The time-consuming comparison is presented in Tab. 3. They both demonstrate the efficiency of our method for the GDSR task, showing the potential for developing lightweight networks and practical applications in the future.

4.4. Ablation Studies

In this section, we validate the design rationality of our SSDNet through ablation experiments based on the RGBDD testset, and present the results in Tab. 4.

SSDNet architecture. In Exp. I, we share the encoders and decoders for RGB and depth images, *i.e*., a unified encoder and decoder are used instead of $\{\mathcal{E}_D, \mathcal{E}_R\}$ and $\{\mathcal{D}_D, \mathcal{D}_R\}$. The number of Restormer blocks is increased to ensure that the learnable parameter number is comparable.

Spherical feature decomposition. In Exp. II, we eliminate the entire feature decomposition module, *i.e*., $\mathcal{L}_{dec}$ in Eq. (11) will not be employed. In Exp. III, we change the distance measure for $\mathcal{L}_{sepn}$ and $\mathcal{L}_{align}$ in Eq. (12) from spherical space distance to Euclidean distance, *i.e*., ℓ_2-loss.

Reconstruction RGB. In Exp. IV, We eliminate $\mathcal{L}^{R}_{pixel}$ in Eq. (10) and explore the change for feature extraction capability without the restriction of reconstructing RGB.

Spherical contrast refinement. In Exp. V, we remove the SCR module and use $\hat{D}_{HR}$ in Eq. (8) as the final output of our model.

Analysis. Tab. 4 shows that changing settings leads to performance degradation, validating our model design.
Exp. I: Unified encoder/decoder ignores modality-specific features, hindering efficient cross-modal feature extraction and causing the largest degradation.
Exp. II: The absence of $\mathcal{L}_{dec}$ prevents decomposing the shared/private features.
Exp. III: ℓ_2 distance partially decomposes features but is less effective than $\mathcal{S}$.
Exp. IV: $\mathcal{L}^{R}_{pixel}$ preserves semantic information in RGB and avoids excessive adaptation of feature separation/alignment.
Exp. V: Removal of SCR leads to mentioned detailed issues and performance degradation.

5. Conclusion

We propose a Spherical Space feature Decomposition network (SSDNet) for guided depth map super-resolution, where a Restormer-based encoder is used to extract the global features of the inputs, and the intermediate features will be mapped to the spherical space to complete the separation/alignment of shared/private features, respectively. Finally, a Restormer-based decoder is to reconstruct the HR depth map. The spherical contrast refinement module is then employed to address the possible detail issues, *e.g*., blurry edges, noisy surfaces and over-transferred RGB texture. The satisfactory output results and the lightweight size demonstrate the superiority of our approach.

Acknowledgement

This work has been supported by the National Key Research and Development Program of China under grant 2018AAA0102201, the National Natural Science Foundation of China under Grant 61976174 and 12201497, Shaanxi Fundamental Science Research Project for Mathematics and Physics under Grant 22JSQ033, the Fundamental Research Funds for the Central Universities under Grant D5000220060, and partly supported by the Alexander von Humboldt Foundation.

References

[1] Jiezhang Cao, Yawei Li, Kai Zhang, and Luc Van Gool. Video super-resolution transformer. *CoRR*, abs/2106.06847, 2021. 3

[2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, pages 213–229. Springer, 2020. 3

[3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, pages 9630–9640. IEEE, 2021. 3

[4] Efrain Castillo-Muñiz and Eduardo Bayro-Corrochano. Geometric spherical networks for visual data processing. In *IJCNN*, 2012. 2, 3

[5] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. In *CVPR*, pages 12299–12310. CVF / IEEE, 2021. 3

[6] Taco Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Convolutional networks for spherical signals. *arXiv preprint arXiv:1709.04893*, 2017. 3

[7] Taco S. Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Spherical CNNs. In *ICLR*, 2018. 2, 3

[8] Riccardo de Lutio, Alexander Becker, Stefano D'Aronco, Stefania Russo, Jan D. Wegner, and Konrad Schindler. Learning graph regularisation for guided super-resolution. In *CVPR*, pages 1969–1978. IEEE, 2022. 2, 8

[9] Riccardo de Lutio, Stefano D'Aronco, Jan Dirk Wegner, and Konrad Schindler. Guided super-resolution as pixel-to-pixel transformation. In *ICCV*, pages 8828–8836. IEEE, 2019. 2

[10] Xin Deng and Pier Luigi Dragotti. Deep coupled ISTA network for multi-modal image super-resolution. *IEEE Trans. Image Process.*, 29:1683–1698, 2020. 2, 3

[11] Xin Deng and Pier Luigi Dragotti. Deep convolutional neural network for multi-modal image restoration and fusion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 43(10):3333–3348, 2021. 2, 8

[12] James Diebel and Sebastian Thrun. An application of markov random fields to range sensing. In *NeurIPS*, pages 291–298, 2005. 2

[13] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *ECCV*, pages 184–199. Springer, 2014. 1

[14] Jiangxin Dong, Jinshan Pan, Jimmy S. Ren, Liang Lin, Jinhui Tang, and Ming-Hsuan Yang. Learning spatially variant linear representation models for joint filtering. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(11):8355–8370, 2022. 2

[15] Xiaoyu Dong, Naoto Yokoya, Longguang Wang, and Tatsumi Uezato. Learning mutual modulation for self-supervised cross-modal super-resolution. In *ECCV*, 2022. 2

[16] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 3

[17] David Ferstl, Christian Reinbacher, René Ranftl, Matthias Rüther, and Horst Bischof. Image guided depth upsampling using anisotropic total generalized variation. In *ICCV*, pages 993–1000, 2013. 2

[18] Zhicheng Geng, Luming Liang, Tianyu Ding, and Ilya Zharkov. RSTT: real-time spatial temporal transformer for space-time video super-resolution. *CoRR*, abs/2203.14186, 2022. 3

[19] Shuhang Gu, Shi Guo, Wangmeng Zuo, Yunjin Chen, Radu Timofte, Luc Van Gool, and Lei Zhang. Learned dynamic guidance for depth image reconstruction. *IEEE Trans. Pattern Anal. Mach. Intell.*, 42(10):2437–2452, 2019. 2

[20] Shuhang Gu, Wangmeng Zuo, Shi Guo, Yunjin Chen, Chongyu Chen, and Lei Zhang. Learning dynamic guidance for depth image enhancement. In *CVPR*, pages 712–721, 2017. 2

[21] Xiang Gu, Jian Sun, and Zongben Xu. Spherical space domain adaptation with robust pseudo-label loss. In *CVPR*, 2020. 2

[22] Xiang Gu, Jian Sun, and Zongben Xu. Unsupervised and semi-supervised robust spherical space domain adaptation. *IEEE Trans. Pattern Anal. Mach. Intell.*, pages 1–1, 2022. 2, 3

[23] Chunle Guo, Chongyi Li, Jichang Guo, Runmin Cong, Huazhu Fu, and Ping Han. Hierarchical features driven residual learning for depth map super-resolution. *IEEE Trans. Image Process.*, 28(5):2545–2557, 2019. 2

[24] Saurabh Gupta, Ross B. Girshick, Pablo Andrés Arbeláez, and Jitendra Malik. Learning rich features from RGB-D images for object detection and segmentation. In *ECCV*, pages 345–360. Springer, 2014. 1

[25] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, pages 9726–9735. CVF / IEEE, 2020. 3

[26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 5

[27] Lingzhi He, Hongguang Zhu, Feng Li, Huihui Bai, Runmin Cong, Chunjie Zhang, Chunyu Lin, Meiqin Liu, and Yao Zhao. Towards fast and accurate real-world depth super-resolution: Benchmark dataset and baseline. In *CVPR*, pages 9229–9238, 2021. 7, 8, 9

[28] Olivier J. Hénaff. Data-efficient image recognition with contrastive predictive coding. In *ICML*, volume 119, pages 4182–4192. PMLR, 2020. 3

[29] Heiko Hirschmüller and Daniel Scharstein. Evaluation of cost functions for stereo matching. In *CVPR*, 2007. 7, 8

[30] Tak-Wai Hui, Chen Change Loy, and Xiaoou Tang. Depth map super-resolution by deep multi-scale guidance. In *ECCV*, pages 353–369. Springer, 2016. 2

[31] Beomjun Kim, Jean Ponce, and Bumsub Ham. Deformable kernel networks for joint image filtering. *Int. J. Comput. Vis.*, 129(2):579–600, 2021. 2, 7, 8

[32] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2014. 8

[33] Wenbo Li, Xin Lu, Jiangbo Lu, Xiangyu Zhang, and Jiaya Jia. On efficient transformer and image pre-training for low-level vision. *CoRR*, abs/2112.10175, 2021. 3

[34] Yijun Li, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep joint image filtering. In *ECCV*, pages 154–169. Springer, 2016. 8

[35] Yijun Li, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan

Yang. Joint image filtering with deep convolutional networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41(8):1909–1923, 2019. 2, 8

[36] Yu Li, Dongbo Min, Minh N. Do, and Jiangbo Lu. Fast guided global interpolation for depth and motion. In *ECCV*, pages 717–733. Springer, 2016. 2

[37] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *ICCVW*, pages 1833–1844. IEEE, 2021. 3

[38] Miao Liao, Feixiang Lu, Dingfu Zhou, Sibo Zhang, Wei Li, and Ruigang Yang. DVI: depth guided video inpainting for autonomous driving. In *ECCV*, pages 1–17. Springer, 2020. 1

[39] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *CVPRW*, 2017. 1

[40] Ming-Yu Liu, Oncel Tuzel, and Yuichi Taguchi. Joint geodesic upsampling of depth images. In *CVPR*, pages 169–176, 2013. 2

[41] Xianming Liu, Deming Zhai, Rong Chen, Xiangyang Ji, Debin Zhao, and Wen Gao. Depth super-resolution via joint color-guided internal and external regularizations. *IEEE Trans. Image Process.*, 28(4):1636–1645, 2019. 1

[42] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, pages 9992–10002. IEEE, 2021. 3

[43] Jiajun Lu and David A. Forsyth. Sparse depth super resolution. In *CVPR*, pages 2245–2253, 2015. 2

[44] Jiangbo Lu, Keyang Shi, Dongbo Min, Liang Lin, and Minh N. Do. Cross-based local multipoint filtering. In *CVPR*, pages 430–437, 2012. 2

[45] Si Lu, Xiaofeng Ren, and Feng Liu. Depth enhancement via low-rank matrix completion. In *CVPR*, pages 3390–3397, 2014. 7, 8

[46] Dongbo Min, Jiangbo Lu, and Minh N. Do. Depth video enhancement based on weighted mode filtering. *IEEE Trans. Image Process.*, 21(3):1176–1190, 2012. 2

[47] Jinshan Pan, Jiangxin Dong, Jimmy S. J. Ren, Liang Lin, Jinhui Tang, and Ming-Hsuan Yang. Spatially variant linear representation models for joint filtering. In *CVPR*, pages 1702–1711, 2019. 8

[48] Jaesik Park, Hyeongwoo Kim, Yu-Wing Tai, Michael S. Brown, and In-So Kweon. High quality depth map upsampling for 3d-tof cameras. In *ICCV*, pages 1623–1630, 2011. 2

[49] Taesung Park, Alexei A. Efros, Richard Zhang, and Jun-Yan Zhu. Contrastive learning for unpaired image-to-image translation. In *ECCV (9)*, volume 12354 of *Lecture Notes in Computer Science*, pages 319–345. Springer, 2020. 3

[50] Wanli Peng, Hao Pan, He Liu, and Yi Sun. IDA-3D: instance-depth-aware 3d object detection from stereo vision for autonomous driving. In *CVPR*, pages 13012–13021, 2020. 1

[51] Haotong Qin, Zhongang Cai, Mingyuan Zhang, Yifu Ding, Haiyu Zhao, Shuai Yi, Xianglong Liu, and Hao Su. Bipointnet: Binary neural network for point clouds. In *ICLR*, 2021. 1

[52] Haotong Qin, Ruihao Gong, Xianglong Liu, Xiao Bai, Jingkuan Song, and Nicu Sebe. Binary neural networks: A survey. *Pattern Recognition*, 2020. 1

[53] Haotong Qin, Ruihao Gong, Xianglong Liu, Mingzhu Shen, Ziran Wei, Fengwei Yu, and Jingkuan Song. Forward and backward information retention for accurate binary neural networks. In *ICCV*, 2020. 1

[54] Gernot Riegler, David Ferstl, Matthias Rüther, and Horst Bischof. A deep primal-dual network for guided depth super-resolution. In *BMVC*. BMVA Press, 2016. 2

[55] Gernot Riegler, Matthias Rüther, and Horst Bischof. Atgv-net: Accurate depth super-resolution. In *ECCV*, pages 268–284. Springer, 2016. 2

[56] Daniel Scharstein and Chris Pal. Learning conditional random fields for stereo. In *CVPR*, 2007. 7, 8

[57] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from RGBD images. In *ECCV*, pages 746–760. Springer, 2012. 7, 8

[58] Xibin Song, Yuchao Dai, Dingfu Zhou, Liu Liu, Wei Li, Hongdong Li, and Ruigang Yang. Channel attention based iterative residual learning for depth map super-resolution. In *CVPR*, pages 5630–5639, 2020. 2

[59] Hang Su, Varun Jampani, Deqing Sun, Orazio Gallo, Erik G. Learned-Miller, and Jan Kautz. Pixel-adaptive convolutional neural networks. In *CVPR*, pages 11166–11175, 2019. 8

[60] Baoli Sun, Xinchen Ye, Baopu Li, Haojie Li, Zhihui Wang, and Rui Xu. Learning scene structure guidance via cross-task knowledge transfer for single depth super-resolution. In *CVPR*, pages 7792–7801, 2021. 2

[61] Jiaxiang Tang, Xiaokang Chen, and Gang Zeng. Joint implicit image function for guided depth super-resolution. In *ACM Multimedia*, pages 4390–4399. ACM, 2021.

[62] Qi Tang, Runmin Cong, Ronghui Sheng, Lingzhi He, Dan Zhang, Yao Zhao, and Sam Kwong. Bridgenet: A joint learning network of depth map super-resolution and monocular depth estimation. In *ACM Multimedia*, pages 2148–2157. ACM, 2021. 2

[63] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, volume 139, pages 10347–10357. PMLR, 2021. 3

[64] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pages 5998–6008, 2017. 3

[65] Jiakai Wang, Aishan Liu, Xiao Bai, and Xianglong Liu. Universal adversarial patch attack for automatic checkout using perceptual and attentional bias. *IEEE Trans. Image Process.*, 31:598–611, 2021. 1

[66] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *ICCV*, pages 548–558. IEEE, 2021. 3

[67] Zhendong Wang, Xiaodong Cun, Jianmin Bao, and Jianzhuang Liu. Uformer: A general u-shaped transformer for image restoration. *CoRR*, abs/2106.03106, 2021. 3

[68] Yang Wen, Bin Sheng, Ping Li, Weiyao Lin, and David Dagan Feng. Deep color guided coarse-to-fine convolutional network

cascade for depth image super-resolution. *IEEE Trans. Image Process.*, 28(2):994–1006, 2019. 2

[69] Richard C. Wilson, Edwin R. Hancock, Elzbieta Pekalska, and Robert P. W. Duin. Spherical and hyperbolic embeddings of data. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(11):2255–2269, 2014. 4

[70] Haiyan Wu, Yanyun Qu, Shaohui Lin, Jian Zhou, Ruizhi Qiao, Zhizhong Zhang, Yuan Xie, and Lizhuang Ma. Contrastive learning for compact single image dehazing. In *CVPR*, pages 10551–10560. CVF / IEEE, 2021. 3

[71] Huikai Wu, Shuai Zheng, Junge Zhang, and Kaiqi Huang. Fast end-to-end trainable guided filter. In *CVPR*, pages 1838–1847, 2018. 2

[72] Enze Xie, Jian Ding, Wenhai Wang, Xiaohang Zhan, Hang Xu, Peize Sun, Zhenguo Li, and Ping Luo. Detco: Unsupervised contrastive learning for object detection. In *ICCV*, pages 8372–8381. IEEE, 2021. 3

[73] Jun Xie, Rogério Schmidt Feris, and Ming-Ting Sun. Edge-guided single depth image super resolution. *IEEE Trans. Image Process.*, 25(1):428–438, 2016. 2

[74] Jun Xie, Rogério Schmidt Feris, Shiaw-Shian Yu, and Ming-Ting Sun. Joint super resolution and denoising from a single depth image. *IEEE Trans. Multim.*, 17(9):1525–1537, 2015. 2

[75] Fu Xiong, Boshen Zhang, Yang Xiao, Zhiguo Cao, Taidong Yu, Joey Tianyi Zhou, and Junsong Yuan. A2J: anchor-to-joint regression network for 3d articulated pose estimation from a single depth image. In *ICCV*, pages 793–802. IEEE, 2019. 1

[76] Ruikang Xu, Mingde Yao, and Zhiwei Xiong. Zero-shot dual-lens super-resolution. In *CVPR*, pages 9130–9139, 2023. 1

[77] Jingyu Yang, Xinchen Ye, Kun Li, Chunping Hou, and Yao Wang. Color-guided depth recovery from RGB-D data using an adaptive autoregressive model. *IEEE Trans. Image Process.*, 23(8):3443–3458, 2014. 2

[78] Yuxiang Yang, Qi Cao, Jing Zhang, and Dacheng Tao. CODON: on orchestrating cross-domain attentions for depth super-resolution. *Int. J. Comput. Vis.*, 130(2):267–284, 2022. 2

[79] Mingde Yao, Dongliang He, Xin Li, Fu Li, and Zhiwei Xiong. Towards interactive self-supervised denoising. *IEEE Transactions on Circuits and Systems for Video Technology*, 2023. 1

[80] Mingde Yao, Dongliang He, Xin Li, Zhihong Pan, and Zhiwei Xiong. Bidirectional translation between uhd-hdr and hd-sdr videos. *IEEE Transactions on Multimedia*, 2023. 1

[81] Xinchen Ye, Baoli Sun, Zhihui Wang, Jingyu Yang, Rui Xu, Haojie Li, and Baopu Li. Pmbanet: Progressive multi-branch aggregation network for scene depth super-resolution. *IEEE Trans. Image Process.*, 29:7427–7442, 2020. 2

[82] En Yu, Zhuoling Li, and Shoudong Han. Towards discriminative representation: Multi-view trajectory contrastive learning for online multi-object tracking. In *CVPR*, pages 8824–8833. IEEE, 2022. 3

[83] Shanxin Yuan, Guillermo Garcia-Hernando, Björn Stenger, Gyeongsik Moon, Ju Yong Chang, Kyoung Mu Lee, Pavlo Molchanov, Jan Kautz, Sina Honari, Liuhao Ge, Junsong Yuan, Xinghao Chen, Guijin Wang, Fan Yang, Kai Akiyama, Yang Wu, Qingfu Wan, Meysam Madadi, Sergio Escalera, Shile Li, Dongheui Lee, Iason Oikonomidis, Antonis A. Argyros, and Tae-Kyun Kim. Depth-based 3d hand pose estimation: From current achievements to future goals. In *CVPR*, pages 2636–2645, 2018. 1

[84] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *CVPR*, pages 5718–5729. IEEE, 2022. 2, 3

[85] Jiahui Zhang, Shijian Lu, Fangneng Zhan, and Yingchen Yu. Blind image super-resolution via contrastive representation learning. *CoRR*, abs/2107.00708, 2021. 3

[86] Kai Zhang, Luc Van Gool, and Radu Timofte. Deep unfolding network for image super-resolution. In *CVPR*, pages 3214–3223, 2020. 1

[87] Zixiang Zhao, Haowen Bai, Jiangshe Zhang, Yulun Zhang, Shuang Xu, Zudi Lin, Radu Timofte, and Luc Van Gool. Cddfuse: Correlation-driven dual-branch feature decomposition for multi-modality image fusion. In *CVPR*, pages 5906–5916, June 2023. 1

[88] Zixiang Zhao, Haowen Bai, Jiangshe Zhang, Yulun Zhang, Kai Zhang, Shuang Xu, Dongdong Chen, Radu Timofte, and Luc Van Gool. Equivariant multi-modality image fusion. *arXiv preprint arXiv:2305.11443*, 2023.

[89] Zixiang Zhao, Haowen Bai, Yuanzhi Zhu, Jiangshe Zhang, Shuang Xu, Yulun Zhang, Kai Zhang, Deyu Meng, Radu Timofte, and Luc Van Gool. DDFM: denoising diffusion model for multi-modality image fusion. *CoRR*, abs/2303.06840, 2023. 1

[90] Zixiang Zhao, Jiangshe Zhang, Shuang Xu, Zudi Lin, and Hanspeter Pfister. Discrete cosine transform network for guided depth map super-resolution. In *CVPR*, pages 5687–5697. IEEE, 2022. 2, 3, 7, 8, 9

[91] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip H. S. Torr, and Li Zhang. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *CVPR*, pages 6881–6890. CVF / IEEE, 2021. 3

[92] Zhiwei Zhong, Xianming Liu, Junjun Jiang, Debin Zhao, Zhiwen Chen, and Xiangyang Ji. High-resolution depth maps imaging via attention-based hierarchical multi-modal fusion. *IEEE Trans. Image Process.*, 31:648–663, 2022. 2

[93] Zhiwei Zhong, Xianming Liu, Junjun Jiang, Debin Zhao, and Xiangyang Ji. Guided depth map super-resolution: A survey. *ACM Computing Surveys*, 2023. 1

[94] Man Zhou, Keyu Yan, Jinshan Pan, Wenqi Ren, Qi Xie, and Xiangyong Cao. Memory-augmented deep unfolding network for guided image super-resolution. *Int. J. Comput. Vis.*, 131(1):215–242, 2023. 2

[95] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: deformable transformers for end-to-end object detection. In *ICLR*, 2021. 3

[96] Michael Zollhöfer, Matthias Nießner, Shahram Izadi, Christoph Rhemann, Christopher Zach, Matthew Fisher, Chenglei Wu, Andrew W. Fitzgibbon, Charles T. Loop, Christian Theobalt, and Marc Stamminger. Real-time non-rigid reconstruction using an RGB-D camera. *ACM Trans. Graph.*, 33(4):156:1–156:12, 2014. 1