Self-Supervised Monocular Depth Estimation with Self-Reference Distillation and Disparity Offset Refinement

Zhong Liu, Ran Li*, Shuwei Shao, Xingming Wu and Weihai Chen

Abstract—Monocular depth estimation plays a fundamental role in computer vision. Due to the costly acquisition of depth ground truth, self-supervised methods that leverage adjacent frames to establish a supervision signal have emerged as the most promising paradigms. In this work, we propose two novel ideas to improve self-supervised monocular depth estimation: 1) self-reference distillation and 2) disparity offset refinement. Specifically, we use a parameter-optimized model as the teacher updated as the training epochs to provide additional supervision during the training process. The teacher model has the same structure as the student model, with weights inherited from the historical student model. In addition, a multiview check is introduced to filter out the outliers produced by the teacher model. Furthermore, we leverage the contextual consistency between high-level and low-level features to obtain multiscale disparity offsets, which are used to refine the disparity output incrementally by aligning disparity information at different scales. The experimental results on the KITTI and Make3D datasets show that our method outperforms previous state-of-the-art competitors.

Index Terms—Monocular depth estimation, Self-supervised learning, Self-reference distillation, Disparity alignment, Multiview check.
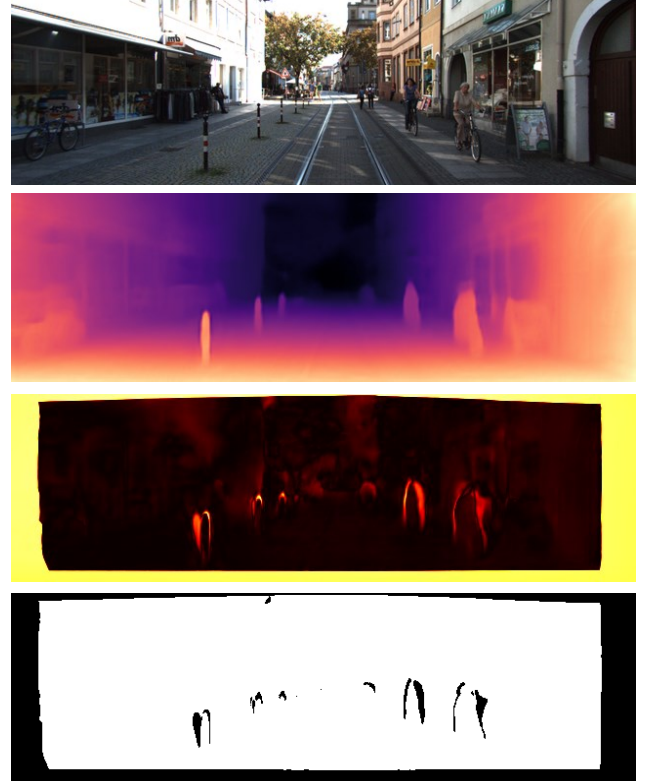
Fig. 1. Illustration of the depth map, error map and binary mask from the proposed method. The first row is the RGB image, and the second row is the depth map. The third row is the intermediate error map generated by the multiview check filter, where the places with large errors are red and yellowish. The fourth row is a binarized hard mask to filter out outliers.

I. INTRODUCTION

Perception of the 3D world is one of the main tasks in computer vision. However, in many cases it might be unfeasible to obtain access to depth information relying on expensive or complex sensors. Depth estimation from a single image has gained extensive attention and has been shown to be a practical technology with applications ranging from localization, navigation, autonomous driving, and robot grasping to 3D reconstruction.

In recent years, supervised monocular depth estimation has been widely studied and has made significant strides [1–5]. Supervised depth estimation is a mapping problem from pixel-level RGB information to depth. With the aid of CNN, self-attention and other mechanisms, depth estimation is performed based on image texture, color information, and surrounding image relationships. While supervised depth estimation has achieved excellent performance, RGB-D data is still constrained in abundance and variety when compared

with available RGB image and video data in the field. Furthermore, gathering a large number of accurate ground-truth datasets is a challenging task due to sensor noise and limited operating capabilities. Recent studies have identified a feasible alternative for performing depth estimator training in a self-supervised manner. The self-supervised method converts the depth estimation into an image synthesis using an intermediary variable (depth or disparity). For instance, the classical method Monodepth2 [6] trains the model to predict the appearance of a target image from the viewpoint of another image, by minimizing the photometric reconstruction loss.

Self-supervised monocular depth estimation relies on the assumption of static scenes and Lambertian surfaces to estimate both depth and relative pose. However, the assumption may
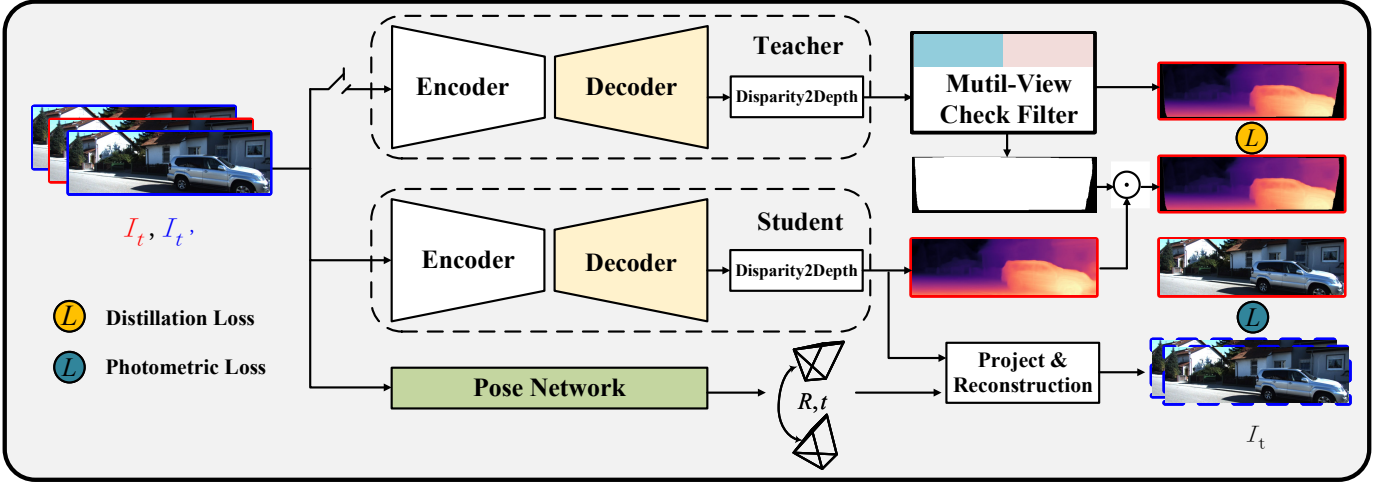
Fig. 2. Overview of the proposed framework. The model adopts an encoder-decoder architecture, in which the encoder is a commonly used backbone, such as ResNet [13] and Swin Transformer [14]. Our decoder outputs disparity, which is then converted to depth. The branch of the teacher model provides depth supervision to the student model after outliers are filtered out by a multiview check filter. In the architecture, the input sequence consists of I_t and $I_{t'}$, where $I_{t'} \in \{I_{t-1}, I_{t+1}\}$. I_t is used to output depth and $I_{t'}$ is used for reconstruction to generate $\widetilde{I}_t$.

not hold in many scenarios, leading to unstable unsupervised learning and local minimum issues in dynamic regions and non-Lambertian or low-textured surfaces. To mitigate this challenge, recent works [7–9] have introduced distillation techniques to facilitate training. These methods first need to build a well-behaved and sophisticated teacher model, and then the teacher model with frozen weights is used to distill the student model. The entire distillation process requires two stages of training to be completed separately, thus resulting in inefficient training. In addition, multiscale prediction structures are commonly used in dense prediction tasks now. If high-quality multiscale disparities can be generated, which help improve the depth estimation performance. Multiscale disparities are usually obtained by upsampling low-resolution disparity, but upsampling operations lose valuable information and cause disparity misalignment, bringing negative effects [10–12].

To address these issues, we propose two novel ideas to enhance self-supervised monocular depth estimation: 1) self-reference distillation and 2) disparity offset refinement. Specifically, to provide additional supervision during the training phase, we train a teacher model with a self-supervised approach during the initial epoch. The teacher model and its distilled counterpart, i.e., the student model, share an identical structure. With the increase of training epochs, the teacher model is continuously updated by inheriting the optimized parameter, so as to provide better depth supervision. However, the depth generated by the teacher model on all pixels is not necessarily reliable, and the edge area and the motion area will have relatively low confidence (Fig.1). We do not expect the teacher model to distill this kind of knowledge with large depth errors to the student model. To obtain better supervision signals, we introduce a multiview check filter to filter outliers in the depth map through two steps of forward projection and backprojection of the camera target view and the source view. For the misalignment of multiscale disparity maps, we introduce the disparity offset fields to refine the disparity output by leveraging the contextual consistency between high-

level and low-level features. The disparity offset fields allow the multiscale disparities to be aligned and enhance the depth prediction. The results can be found in the ablation experiment in Table II.

To summarize, the main contributions are listed as follows:
- We propose a novel self-supervised monocular depth estimation method employing distillation technique and disparity offset refinement to effectively improve the depth estimation performance.
- We propose self-reference distillation and introduce the multiview check technique to remove the depth outliers from the teacher model, implementing efficient single-stage online distillation learning.
- We leverage the contextual consistency in adjacent features to predict the disparity offset field. The aligned refinement for the disparity solves the disparity misalignment problem caused by the upsampling process.
- We conduct extensive experiments on the KITTI and Make3D datasets, demonstrating that our model outperforms existing state-of-the-art methods.

II. RELATED WORK

This section reviews the literature on monocular depth estimation and knowledge distillation.

A. Supervised monocular depth estimation

Depth estimation from a single image is an inherently ill-posed problem as pixels in the image can have numerous plausible depths. Saxena et al. [15] used a discriminatively trained Markov Random Field that incorporates multiscale local and global features and modeled both depths at individual points as well as the relation between depths at different points. Eigen et al. [16] introduced a multiscale architecture to make a coarse global depth prediction and progressively refine this prediction locally using two separate networks. A representative BTS method was proposed by [17] using local
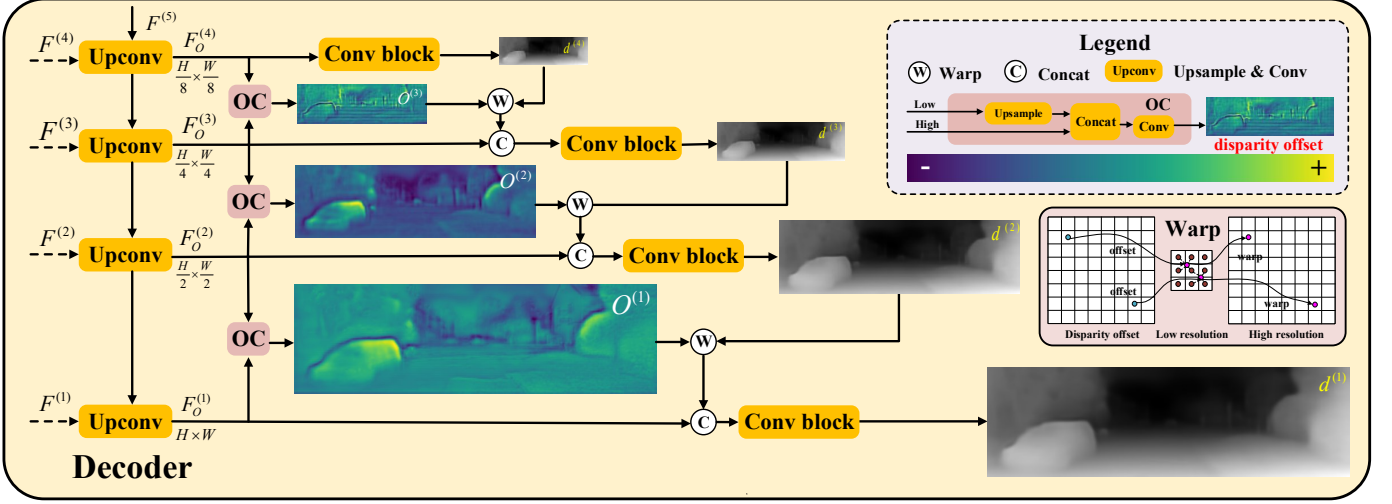
Fig. 3. Details of the decoder. The input of the decoder comes from the multiscale feature $F^{(i)}$ generated by the encoder, and the output is a disparity map of H×W size. The disparity output by the decoder is obtained by progressively upsampling the low-resolution disparity map. In this process, the offset calculation (OC) module provides a disparity offset field to fuse with the upsampled disparity to achieve disparity alignment. In the warp process, the high-resolution disparity map is the bilinear interpolation of the neighboring pixels in low-resolution disparity map, where the neighborhoods are defined according learned disparity offset.

planar guidance layers to guide the features to full resolution instead of standard upsampling layers during the decoding phase. Shariq et al. [3] divided the depth range into bins whose center value is estimated adaptively per image. The final depth values are estimated as linear combinations of the bin centers. Based on the different features generated by the encoder, Shao et al. [18] used the strategy of ensemble learning to obtain more robust depth prediction. Song et al. [2] adopted the Laplacian pyramid for resolving the problem of monocular depth estimation. By recovering depth residuals from encoded features in different levels of the Laplacian pyramid and summing up those predicted results progressively.

Various fully supervised methods based on deep learning have been continuously explored. However, all the above methods require high-quality ground-truth depth, which is costly to obtain.

B. Self-Supervised monocular depth estimation

While fully supervised approaches for depth estimation advance rapidly, the availability of precise depth labels becomes a significant problem. Hence, more recent self-supervised works provide alternatives to avoid the need for ground-truth depth annotations.

In monocular depth estimation, self-supervised approaches unify depth estimation and ego-motion estimation into one framework using view synthesis as a supervision signal. Zhou et al. [19] proposed an unsupervised learning framework for monocular depth estimation and camera motion estimation from unlabeled video sequences. Zou et al. [20] proposed leveraging geometric consistency as additional supervision signals for simultaneously training single-view depth prediction and optical flow estimation models using unlabeled video sequences based on brightness constancy and spatial smoothness priors. Furthermore, Godard et al. [6] proposed a classical method Monodepth2, and they adopted an auto-

masking scheme to filter out invalid pixels from moving objects and introduced a minimum reprojection loss to address occlusions. Based on Monodepth2, numerous current self-supervised monocular depth estimation approaches [21–23] are further researched. Liu et al. [24] proposed a domain-separated network for self-supervised depth estimation of all-day images. Michael et al. [25] presented a novel method for predicting accurate depths by exploiting wavelet decomposition. Shu et al. [26] exploited the point cloud consistency constraint to optimize view synthesis process. Jaehoon et al. [27] exploited semantic-aware depth features that integrate the semantic and geometric knowledge to overcome the limitations of the photometric loss. Vitor et al. [28] leveraged novel symmetrical packing and unpacking blocks to jointly learn to compress and decompress detail-preserving representations using 3D convolutions and implement a self-supervised monocular depth estimation method combining geometry with a new deep network. Akhil et al. [29] performed monocular depth estimation by virtual-world supervision and real-world SfM self-supervision. They compensate the SfM self-supervision limitations by leveraging virtual-world images with accurate semantic and depth supervision, and addressing the virtual-to-real domain gap. Other published methods were based on feature representation learning [30], competitive collaboration [31], edge, normal [32, 33], semantic segmentation [34, 35].

C. Knowledge Distillation

The concept of knowledge distillation was first proposed by [36] and made popular by [37]. Knowledge distillation aims to transfer knowledge from a teacher model to obtain a powerful and lightweight student model. The idea has been exploited for many computer vision tasks [38–40].

Recently, some works have attempted to exploit distillation for unsupervised depth estimation. Ren et al. [8] proposed an adaptive co-teaching framework for unsupervised depth
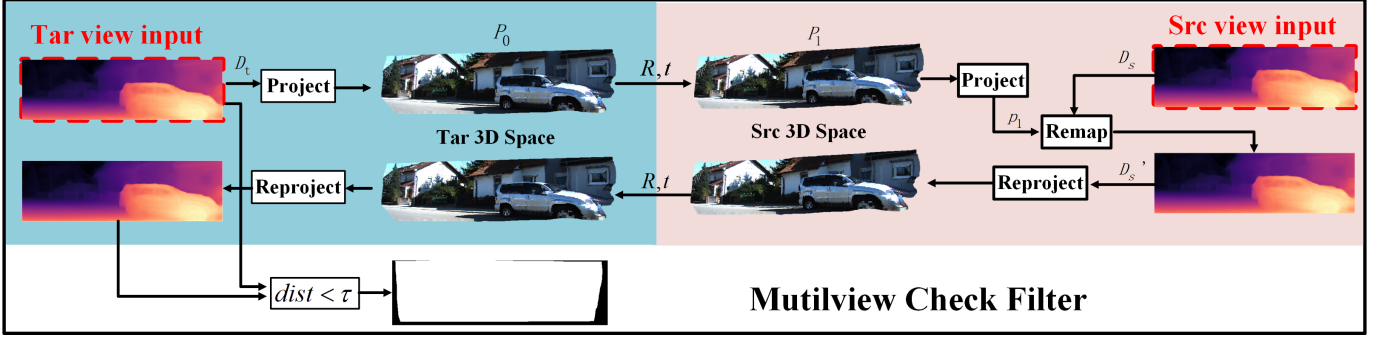
Fig. 4. Mutilview check filter. This process uses the target view and the source view to perform forward projection and backward reprojection. In the figure, dist represents a certain distance function and pixels exceeding the threshold are filtered out. The white part of the output mask is the reserved part, and the black part is filtered out. The tar view input corresponds to the Z_{ct} in Eq.11, and the detailed formulation of $dist < \tau$ can be seen in Eq.16.

estimation that enjoys the strengths of knowledge distillation and ensemble learning for more accurate depth estimation. Matteo et al. [9] proposed a new and peculiar self-teaching paradigm to model uncertainty and then used the uncertainty to assist knowledge distillation. Lyu et al. [7] adopted a large model to improve the accuracy of a lightweight model through two-stage training. These methods usually need to build a complex teacher model. The training of the teacher model and distillation processes are completely separated, thus resulting in relatively large time and computational costs.

Inspired by BAN [41] training a student model similarly parameterized as the teacher model and making the trained student be a teacher model in a new round, we propose our self-reference learning mode in monocular depth estimation. The self-training scheme [42] generates distillation labels for unlabeled data and trains the student model with these labels. Different from the two-stage distillation method used in previous work [7–9], our self-reference distillation achieves efficient single-stage online distillation.

III. METHOD

In this section, we first present the proposed network architecture. Then, we describe the implementation details of self-reference distillation, including the construction of a teacher-student distillation model and the implementation of a multiview check filter. Finally, we provide a detailed account of how we leverage contextual consistency between high-level and low-level features to obtain the disparity offset.

A. Motivation

In self-supervised monocular depth estimation, the depth and relative pose are estimated together, and these two inter-mediate variables are used to perform projection and repro-jection operations to synthesize images. Then the photometric error is minimized to train the model. Static scenes and Lambertian surfaces are important underlying assumptions for self-supervised depth estimation. However, dynamic regions, non-Lambertian surfaces or low-texture surfaces violate this assumption, causing unstable unsupervised training and local minima problems. To mitigate this challenge, recent works have introduced distillation techniques to facilitate training.

These methods first need to build a well-behaved and so-phisticated teacher model, and then the teacher model with frozen weights is used to distill the student model. The entire distillation process requires two stages of training to be completed separately, thus resulting in inefficient training. Therefore, we design efficient single-stage online distillation learning to further alleviate this problem. We first let the model optimize a set of parameters through self-supervision in the first training epoch. In the next epoch of training, the optimized parameters will be loaded for direct inference and generate depth pseudolabels. To produce higher-quality depth supervision, we propose a multiview check filter, which subjects depth pseudolabels to multiple views to filter out outliers.

In addition, multiscale prediction structures are commonly used in dense prediction tasks now. If high-quality multi-scale disparities can be generated, it will help improve the performance of depth estimation. Multiscale disparities are usually obtained by upsampling low-resolution disparity, but upsampling operations will lose valuable information and cause disparity misalignment, bringing negative effects [10–12]. Therefore, we leverage the contextual consistency be-tween high-level and low-level features to obtain the disparity offset and refine the disparity output incrementally based on the disparity offset to align disparity information at different scales.

B. Network Architecture

For self-supervised monocular depth estimation, the pro-posed method utilizes an encoder-decoder architecture with skip connections. In this architecture, the encoder transforms the input image into a latent space representation, while the decoder reconstructs the disparity from the latent space repre-sentation, as shown in Fig.2. After converting from disparity to depth, the output of the decoder becomes the model's final output. During the training process, the depth output by the depth network and the relative pose output by the pose network are used for view synthesis, and self-supervised training is achieved by optimizing the photometric loss. Additionally, the output depth of the teacher depth network is used as a pseudolabel to distill knowledge to the student model.
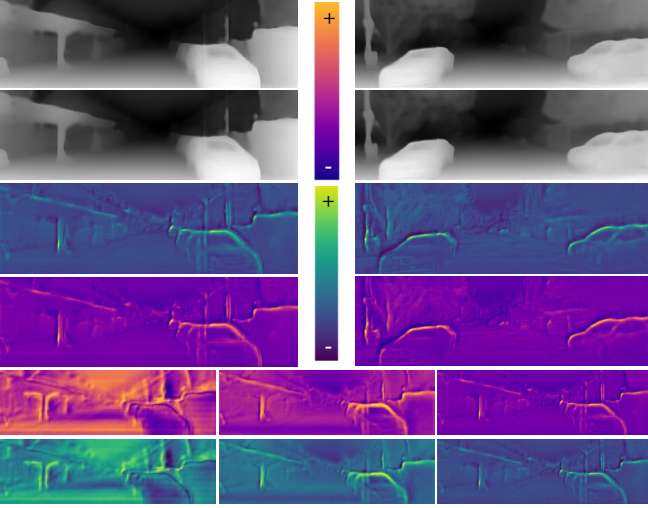
Fig. 5. Visualization of disparity offset. The first two rows are low-resolution disparity and high-resolution disparity. The third row and the fourth row are horizontal disparity offset and vertical disparity offset. The last two rows are different levels of disparity offset, and from left to right are high level to low level.

Encoder. The encoder plays a crucial role in effectively extracting features. Therefore, inspired by the outstanding performance of Transformers[43–46] in various vision tasks, we adopt an improved multipath Vision Transformer architecture MPViT [47], which leverages both the local connectivity of convolutions and the global context of the transformer. The encoder receives a single frame with resolution $H \times W$ as input and extracts features at five different scales, with resolutions $H \times W$, $\frac{H}{2} \times \frac{W}{2}$, $\frac{H}{4} \times \frac{W}{4}$, $\frac{H}{8} \times \frac{W}{8}$, and $\frac{H}{16} \times \frac{W}{16}$. These multiscale features are then directly accessed by the decoder through skip connections. According to the differences in layers and channels in the four scales, MPViT [47] has tiny, xsmall, small and base versions. We use MPViT-S (small) with a parameter size of 22.8M, equivalent to ResNet50 (25M) and Swin Transformer-T (28M). The layers of the 4 scales are set to [1,3,6,3], and the channels are set to [64, 128, 216, 288]. It should be noted that we actually use features of 5 scales, so we modify MPViT and add $\frac{H}{2} \times \frac{W}{2}$ scale features. The layers of this scale are set to 3, and the channels are set to 64.

Decoder. The multiscale features output by the encoder are used as input to the decoder, which progressively upsamples and convolves the feature maps to increase their resolution. The upsampled feature maps are then fused with the skip-connected encoding features and passed through the layers in a top-down manner, combining strong semantic information features and high-resolution features. Based on the features of two adjacent scales, the offset calculation module calculates the disparity offset field at each scale. The low-resolution disparity is refined with the assistance of the disparity offset and is progressively passed to the high-resolution scale to obtain the output disparity of the decoder. More specific details on how the offset calculation (OC) module calculates the disparity offset field from the features of adjacent scales will be discussed in the section on disparity alignment. The structure of the depth decoder is shown in Fig.3. The multi-

scale features generate three scale disparity offsets (O) in the figure. The final disparity output of the model is generated from the low-resolution disparity step-by-step upsampling, and the disparity map after each upsampling operation will be aligned according to O. After the disparity of the H×W scale is refined, the decoder outputs the disparity and then converts it to depth.

Pose Network. Our pose network uses the same lightweight architecture ResNet18[13] as [6, 7, 48, 49], taking a sequence of three frames as input to predict a 6-DoF relative pose between adjacent frames.

C. Self-Supervised Learning

Disparity alignment. Dense prediction tasks achieve a better performance with multiscale predictions. In depth estimation, multiscale disparities are usually obtained by upsampling low-resolution disparity. However, commonly used upsampling operations (e.g., bilinear interpolation) lose valuable information and cause disparity misalignment [10–12].

Inspired by various feature alignment works [10, 50, 51], we design an offset calculation module (OC) that utilizes the adjacent scale features to calculate the disparity offset field. The OC outputs allow effective disparity alignment of different scales and improve the prediction accuracy, which is provided evidence in the ablation study.

The OC module in the decoder receives two resolution features $\{F_o^{(i)}, F_o^{(i+1)}\}$. (1) The low-resolution feature $F_o^{(i+1)}$ is first upsampled to obtain features of the same resolution as $F_o^{(i)}$. (2) The resulting features are concatenated with the high-resolution feature $F_o^{(i)}$ to produce the disparity offset $O^{(i)}$, as shown in the offset calculation structure of the legend in Fig.3. (3) The disparity $d^{(i+1)}$ is warped by $O^{(i)}$ to obtain the refined disparity map (the high-resolution disparity map). In the warp process, the high-resolution disparity map is the bilinear interpolation of the neighboring pixels in low-resolution disparity map, where the neighborhoods are defined according learned disparity offset as shown in Fig.3. (4) Refined disparity is concatenated with $F_o^{(i)}$ to generate $d^{(i)}$. This process can be mathematically formulated as follows:

$$O^{(i)} = Conv(Cat(F_o^{(i)}, Upsample(F_o^{(i+1)}))), \quad (1)$$

$$d^{(i)} = Conv(Cat(F_o^{(i)}, Warp(O^{(i)}, d^{(i+1)}))), \quad (2)$$

In the legend of the figure, we mark that the disparity offset is positive or negative, which represents the direction of the disparity offset. The disparity offset includes the horizontal offset and vertical offset. In Fig.3, we only indicate the horizontal offset for convenience.

In the visualization of Fig.5, we show the disparity offset in the horizontal and vertical directions. The first two rows are low-resolution disparity and high-resolution disparity, respectively and the low-resolution disparity map needs to be aligned with the high-resolution disparity map by offsetting both horizontally and vertically. The third row and the fourth row are the horizontal disparity offset and vertical disparity offset. The last two rows are different levels of disparity offset, and from left to right are high level to low level. As the upsampling proceeds, high level features are propagated to

appropriate high-resolution positions following the guidance of disparity offset. Disparity offset have coarse-to-fine trends from high level to low level, so at the low level, the main salient regions are mainly located at the edges with rich details. In the ablation experiments, we verify the effectiveness of the disparity alignment and show the visualization results of predicted depth with disparity alignment and without disparity alignment in Fig.8. Our proposed DA is much clearer in the estimation of edge contours, which is consistent with the performance of the multiscale disparity offsets.

Photometric Loss. Self-supervised monocular depth estimation performs image synthesis by minimizing the photometric loss during training, so that the images synthesized from other views are close to the appearance of the target image, and the depth and relative pose required for the synthesized image are solved during this optimization process. I_t denotes the target image and $I_{t'}$ denotes the source image i.e. $I_{t'} \in \{I_{t-1}, I_{t+1}\}$. The relative pose of each source image's associated image is indicated as $T_{t \to t'}$ and the camera intrinsics are denoted as K. D_t is a depth map transformed from the disparity $d^{(1)}$. Similar to [6, 52, 53], we predict a dense depth map that minimizes the photometric reprojection loss $\mathcal{L}_{pe}$, where

$$\widetilde{I}_t = \pi(I_{t'}, K, D_t, T_{t \to t'}), \tag{3}$$

$$\mathcal{L}_{pe} = min\mathcal{F}(I_t, \widetilde{I}_t), \tag{4}$$

$$\mathcal{F}(I_t, \widetilde{I}_t) = \alpha \frac{1 - SSIM(I_t, \widetilde{I}_t)}{2} + 2\alpha|I_t - \widetilde{I}_t|, \tag{5}$$

Here π is a reconstruction function following [6, 52] and $\widetilde{I}_t \in \{I_{t-1 \to t}, I_{t+1 \to t}\}$. $\mathcal{F}$ is the weighted sum of the intensity difference term $\mathcal{L}_1$ [6, 52] and the structural similarity term $SSIM$ [54] and α is set to 0.85 as [6].

Smoothness Loss. As in [6, 52], we use edge-aware smoothness loss to encourage the smoothness property of inverse depth map:

$$\mathcal{L}_s = |\partial_x d_t^*|e^{-|\partial_x I_t|} + |\partial_y d_t^*|e^{-|\partial_y I_t|}, \tag{6}$$

where $d_t^* = d_t/\bar{d}_t$ is the mean-normalized inverse depth from [55] to discourage shrinking of the estimated depth.

D. Knowledge Distillation

Self-reference distillation. To mitigate the unstable unsupervised training and local minima problems, knowledge distillation is introduced in depth estimation [7–9], using the pseudo depth labels generated by a teacher model as supervision for a student model.

Our knowledge distillation process is finished in a single stage, online, as opposed to earlier work [56], which needs a two-stage training procedure: (1) finishing training a well-behaved and sophisticated teacher model and (2) the teacher model with frozen weights is used to distill the student model. In our method, the student branch of Fig.2 depicts the model going through self-supervised learning first. The model's weights are saved after the initial training epoch. The weights from the previous epoch of training are loaded online and utilized for inference starting with the second training epoch, which corresponds to the teacher branch in
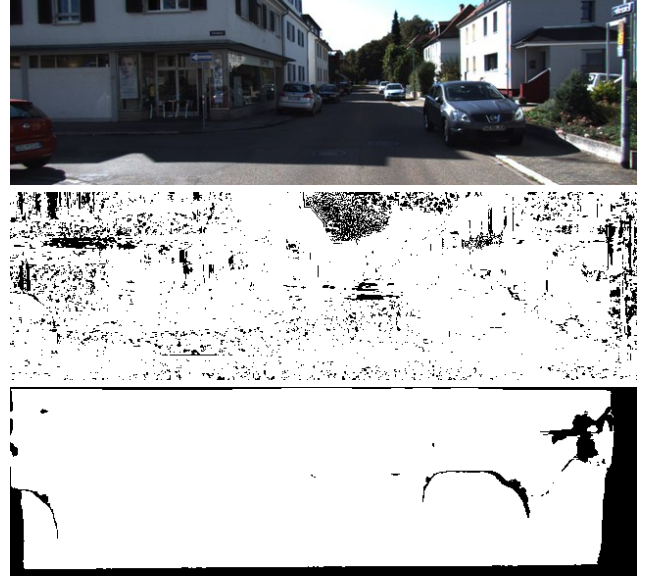

Fig. 6. Mask comparison. The second row is the auto mask, where the area with large photometric errors is masked and the third row is our filter mask, where the area with large depth errors is masked.

Fig.2. The student model is then distilled using the output of this inference as pseudo depth labels. In terms of network architecture, the teacher branch model is identical to the student branch model, with the only difference being the model weights. As shown in Fig.2, the decoder performs a disparity-to-depth conversion after outputting the disparity. To make more comprehensive use of information, we not only convert the depth of the H×W scale, but also convert the multiscale disparities into multiscale depth maps, which can provide more supervision information to distill the student model on multiple scales. Our distillation loss can be written as:

$$\mathcal{L}_d = \frac{1}{4}\sum_{i=1}^{4}(||D_{teacher} - \hat{D}_{student}||_1)_i. \tag{7}$$

Mutilview Check. Although teacher-student distillation provides supervision during training, the prediction of the teacher model at each pixel is not highly reliable. For example, the edge area will have relatively low confidence, resulting in some incorrect outlier points.

Like Monodepth2, we also employ masking strategy to boost performance. The auto-masking strategy of Monodepth2 calculates the minimum photometric error between RGB images to obtain the mask, which is used to re-weight the self-supervised loss term. In knowledge distillation, our aim is to remove out these points with large depth errors. The auto-masking technique does enhance performance by reducing motion scenes, however it masks out points with large photometric errors rather than large depth errors. The area with large photometric errors does not completely reflect the large depth errors as illustrated in Fig.6. Our needs cannot be satisfied by the auto masking strategy.

To achieve better distillation of depth information, we design a multiview check filter demonstrated in Fig.4, to filter outliers and offer a hard mask for teacher-student distillation.

Specifically, the multiview check filter receives the depth output from the model, which consists of the depth map from the target view and the depth map from the source view. According to the imaging principles of a pinhole camera, the correspondence between the pixel points of each perspective and the 3D spatial points can be mathematically described:

$$Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{K} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}, \qquad (8)$$

$$= \begin{bmatrix} \mathbf{K} & \mathbf{0} \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix}, \qquad (9)$$

$$\mathbf{K} = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad (10)$$

where $\mathbf{K}$ refers to camera intrinsic matrix, $\begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix}$ refers to camera extrinsic matrix. (X_w, Y_w, Z_w) and (X_c, Y_c, Z_c) represent points in the world coordinate and camera coordinate, respectively.

We use $\mathbf{p_0} = (u, v)$ to represent an arbitrary point in the target view and project it into the 3D space of the target view $\mathbf{P_0} = (X_{ct}, Y_{ct}, Z_{ct})$ through depth $\mathbf{D_t}(u, v) = Z_{ct}$, as shown in Eq.11.

$$\begin{bmatrix} X_{ct} \\ Y_{ct} \\ Z_{ct} \end{bmatrix} = \mathbf{K}^{-1} Z_{ct} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}, \qquad (11)$$

Combined with the relative pose $T_{t \to t'}$ output by the pose network, we can obtain the 3D representation $\mathbf{P_1} = (X_{cs}, Y_{cs}, Z_{cs})$ of $\mathbf{P_0}$ in the source view, as shown in Eq.12.

$$\begin{bmatrix} X_{cs} \\ Y_{cs} \\ Z_{cs} \end{bmatrix} = T_{t \to t'} \begin{bmatrix} X_{ct} \\ Y_{ct} \\ Z_{ct} \end{bmatrix} = T_{t \to t'} \mathbf{K}^{-1} Z_{ct} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}, \qquad (12)$$

The $\mathbf{P_1}$ is projected to the 2D point $\mathbf{p_1} = (u_1, v_1)$ of the source view through $Z_{cs} = \mathbf{D_s} = model(I_{t'})$ as shown in Eq.13.

$$Z_{cs} \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix} = \mathbf{K} \begin{bmatrix} X_{cs} \\ Y_{cs} \\ Z_{cs} \end{bmatrix}, \qquad (13)$$

Then, the source view depth map $\mathbf{D_s}$ is remapped to $\mathbf{D'_s}$ according to $\mathbf{p_1} = (u_1, v_1)$. Similar to the projection forward process mentioned above, we use $\mathbf{D'_s}$ to perform a reprojection process to obtain reprojected 3D point $\widetilde{\mathbf{P_0}}$, 2D point $\widetilde{\mathbf{p_0}}$ and depth map $\widetilde{\mathbf{D_r}}$ in the target view.

A reprojection error at $\widetilde{\mathbf{p_0}}$ is defined as e_{reproj}, and a geometric error e_{geo} is used to measure the relative depth error, which is formulated as [57]:

$$e_{reproj} = ||\widetilde{\mathbf{p_0}} - \mathbf{p_0}||_2, \qquad (14)$$

$$e_{geo} = \frac{|\mathbf{D_r}(\mathbf{p_0}) - \mathbf{D_r}(\widetilde{\mathbf{p_0}})|}{\mathbf{D_r}(\mathbf{p_0})}, \qquad (15)$$

The valid subset of pixels for the filter mask is determined by Eq.16, where α and β are hyperparameters. We set $\alpha = \beta = 4$.

$$\{\mathbf{p_0}\}_i = \{\mathbf{p_0}|e_{reproj} < \alpha \bar{e}_{reproj}, e_{geo} < \beta \bar{e}_{geo}\}, \qquad (16)$$

$\{\mathbf{p_0}\}_i$ represents the set of multiview checks between the target view and the i-th source view, and the intersection of the sets calculated under all source views is the filter mask. We show the e_{geo} and the filter mask in Fig.1 and Fig.9. In Fig.9, the error map on the third row is e_{geo}, and the mask on the fourth row is the filter mask. In the error map, darker areas represent smaller relative errors (black areas), and lighter areas have greater errors (red, yellowish areas). In knowledge distillation, it is not expected that teacher model to distill inaccurate depth information to the student model. Inaccurate depth information is mainly distributed on the border around the image and a small part is distributed in the middle area as shown in Fig.9. These outliers with relatively large errors are harmful to the student model if they also participate in knowledge distillation. The most direct filtering method is to use Eq.16 to generate a binary hard mask. It is easy to find that the filter mask and error map in Fig.9 may not necessarily correspond exactly. In the picture on the left, the points in the yellow area (with relatively large errors) are basically filtered out, while in the picture on the right, only part of the points in the yellow area are filtered out. This is caused by two reasons: one is caused by the binary hard mask, which will be filtered out only when it reaches a certain threshold, and the other is related to the setting of the threshold. Different image scenes require different thresholds (α and β) for truncation, but for ease of implementation, we choose a unified α and β.

In fact, we also consider using a soft mask as a weight to balance the supervision loss term. We use the error map directly as a soft mask, i.e. $M = 1 - e_{geo}$ to aid the distillation. However, the soft masking method impairs the distillation. Because the supervisory signal is not truncated in the part with large error, this method also brings negative optimization, as shown in Table II of the ablation experiment results.

Distillation Loss. After the depth pseudo labels produced by the teacher model are filtered by a multiview check filter, we apply the resulting filter masks to our distillation loss. The modified distillation loss can be formulated as:

$$\mathcal{L}_d = \frac{1}{4} \sum_{i=1}^{4} (M||D_{teacher} - \hat{D}_{student}||_1)_i, \qquad (17)$$

where M is the filter mask.

Total Loss The total loss in training consists of three parts, photometric loss $\mathcal{L}_{pe}$, smoothing loss $\mathcal{L}_s$ and distillation loss $\mathcal{L}_d$, which are calculated at four scales.

$$\mathcal{L}_{total} = \frac{1}{4} \sum_{i=1}^{4} (\mu \mathcal{L}_{pe} + \lambda \mathcal{L}_s + \gamma \mathcal{L}_d)_i, \qquad (18)$$

with λ set to 10^{-3} and γ set to 0.1. Similar to previous works[6, 22], we apply a per-pixel binary mask, i.e. $\mu \in \{0, 1\}$, which is formulated as:

$$\mu = [min\mathcal{F}(I_t, \widetilde{I}_t) < min\mathcal{F}(I_t, I_{t'})], \qquad (19)$$

where [] is the Iverson bracket.

Method	Train	Backbone	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Eigen [16]	D	-	0.203	1.548	6.307	0.282	0.702	0.890	0.890
Liu [58]	D	VGG16	0.201	1.584	6.471	0.273	0.680	0.898	0.967
Klodt [59]	D*M	ResNet50	0.166	1.490	5.998	-	0.778	0.919	0.966
AdaDepth [60]	D*	ResNet50	0.167	1.257	5.578	0.237	0.771	0.922	0.971
DVSO [61]	DS	ResNet50	0.097	0.734	4.442	0.187	0.888	0.958	0.980
SVSM FT [62]	DS	VGG16	**0.094**	**0.626**	4.252	0.177	0.891	0.965	0.984
Guo [63]	DS	VGG16	0.096	0.641	**4.095**	**0.168**	**0.892**	**0.967**	**0.986**
Monodepth2 [6]	M	ResNet18	0.115	0.903	4.863	0.193	0.877	0.959	0.981
MonoDEVSNet [29]	M	ResNet18	0.116	0.836	4.735	-	0.860	0.954	-
R-MSFM3 [64]	M	ResNet18	0.114	0.815	4.712	0.193	0.876	0.959	0.981
R-MSFM6 [64]	M	ResNet18	0.112	0.806	4.704	0.191	0.878	0.960	0.981
SAFENet [27]	M+Se	ResNet18	0.112	0.788	4.582	0.187	0.878	0.963	0.983
VC-Depth [65]	M	ResNet18	0.112	0.816	4.715	0.190	0.880	0.960	0.982
Ours	M	ResNet18	0.111	0.762	4.619	0.186	0.877	0.961	0.983
Shu [26]	M	ResNet50	0.129	0.976	4.958	0.203	0.848	0.951	0.979
Mono-Uncertainty [9]	M	ResNet50	0.111	0.863	4.756	0.188	0.881	0.961	0.982
PackNet†[28]	M	PackNet	0.108	0.727	4.426	0.184	0.885	0.963	0.983
Johnston et al. [66]	M	ResNet101	0.106	0.861	4.699	0.185	0.889	0.962	0.982
MonoFormer [67]	M	Res50+ViT	0.106	0.839	4.627	0.185	0.884	0.962	0.983
CADepth [21]	M	ResNet50	0.105	0.769	4.535	0.181	0.892	0.964	0.983
Ours	M	ResNet50	0.106	0.718	4.520	0.180	0.886	0.964	0.983
DIFFNet [23]	M	HRNet18	0.102	0.749	4.445	0.179	0.897	0.965	0.983
MonoViT [22]	M	MPViT-S	**0.099**	0.708	4.372	0.175	**0.900**	**0.967**	0.984
Ours	M	MPViT-S	**0.099**	**0.659**	**4.314**	**0.174**	0.898	**0.967**	**0.985**
Monodepth2(1024×320) [6]	M	ResNet18	0.115	0.882	4.701	0.190	0.879	0.961	0.982
R-MSFM3(1024×320) [64]	M	ResNet18	0.112	0.773	4.581	0.189	0.879	0.960	0.982
R-MSFM6(1024×320) [64]	M	ResNet18	0.108	0.748	4.470	0.185	0.889	0.963	0.982
Ours(1024×320)	M	ResNet18	0.106	0.673	4.379	0.180	0.886	0.965	0.984
DCNDepth(1024×320) [68]	M	ResNet50	0.104	0.720	4.494	0.181	0.888	0.965	0.984
CADepth(1024×320) [21]	M	ResNet50	0.102	0.734	4.407	0.178	0.898	0.966	0.984
Ours(1024×320)	M	ResNet50	0.102	0.653	4.381	0.178	0.898	0.966	**0.985**
DIFFNet(1024×320) [23]	M	HRNet18	0.097	0.722	4.345	0.174	0.907	0.967	0.984
MonoViT(1024×320) [22]	M	MPViT-S	**0.096**	0.714	4.292	0.172	**0.908**	0.968	0.984
Ours(1024×320)	M	MPViT-S	**0.096**	**0.635**	**4.158**	**0.171**	0.905	**0.969**	**0.985**

TABLE I

QUANTITATIVE RESULTS. Comparison of our method to existing methods on KITTI 2015 [69] using the Eigen split. The best results in each category are in bold. The resolutions we used for training and testing the models were 640×192 and 1024×320 (marked in the table). In the training approach, Se stands for training with semantic labels, D for depth supervision, D* for auxiliary depth supervision, and M for mono self-supervision. † refers to the model pretained on Cityscapes [70].

IV. EXPERIMENTS

A. Implementation Details

We use PyTorch to implement our model and the backbone includes ResNet18, ResNet50, Swin Transformer-T, MPViT-S, pretrained on the ImageNet1K dataset [71]. There are a total of 20 training epochs. In the first epoch, γ in the $\mathcal{L}_{total}$ item is set to 0, and from the second epoch of training, we load the model saved in the last epoch for teacher-student distillation and the $\mathcal{L}_{total}$ item is set to 0.1. The initial learning rate for the optimizer we employ, AdamW [72], is set to 1e-4. Inputs with a resolution of 640×192 are trained with a single A5000 GPU, and the batch size is set to 12. Inputs with a resolution of 1024×320 adopt distributed data parallel training, requiring 4 GPUs, and the batch size of a single GPU is set to 4.

B. Datasets

KITTI [69]. The KITTI dataset provides 61 scenes from cities, residential areas, roads, and campuses, utilizing a typical image size of 1242×375. We follow the data split of Eigen et al. [16] and Zhou et al.'s [19] preprocessing. There are 39,810 monocular triplets for training and 4,424 for validation. In the comparative experiment with other methods, we conduct on the test set [16] containing 697 images, which provides 652 depth ground-truth labels. We report results using the per-image median ground-truth scaling [19] during evaluation.

Make3D [74]. The Make3D dataset is an outdoor dataset with a scene similar to KITTI with a fixed image size of 1704×2272, containing a training set of 400 image-depth pairs and a test set of 134 image-depth pairs, which is generally used as a generalization test for monocular depth estimation. Following previous preprocessing [6, 22] on a center crop with a 2×1 ratio, we test the performance of different solutions [6, 22, 49].

C. Quantitative Evaluation

In the comparison experiment, we evaluate images with two resolutions 640×192 and 1024×320 on the KITTI [69] dataset, adopting the standard metrics (Abs Rel, Sq Rel,
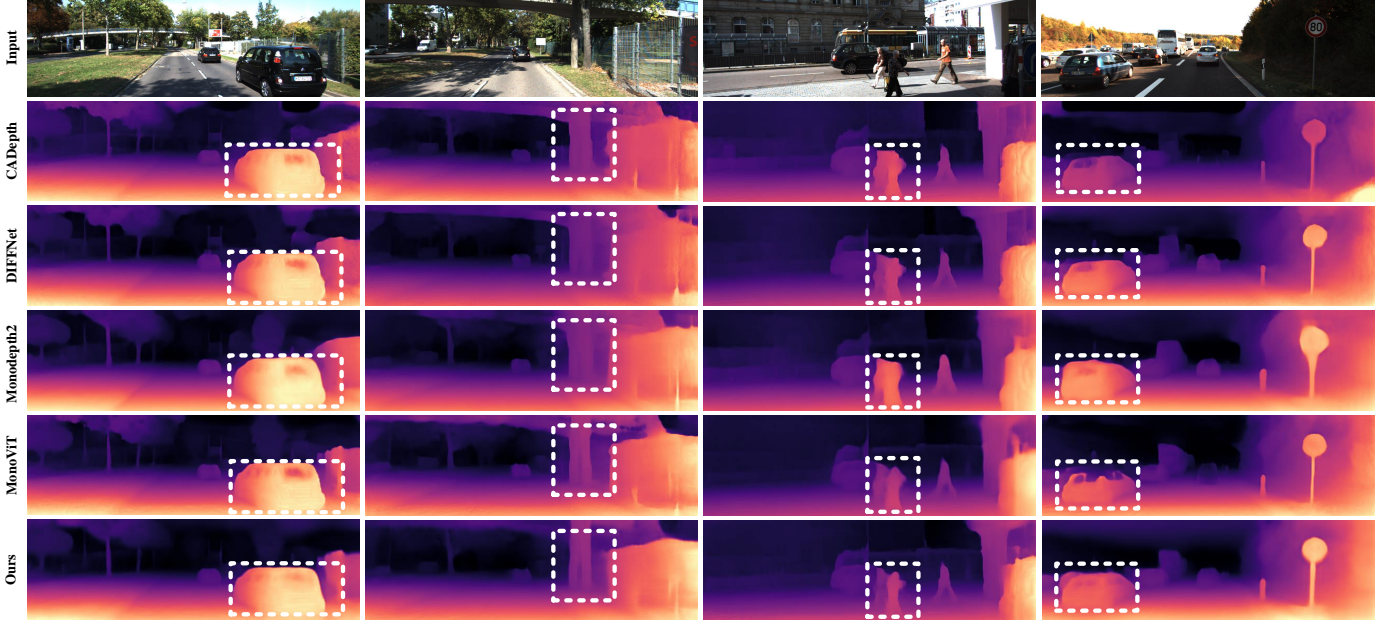
Fig. 7. Qualitative results on the KITTI [69]. Our proposed model in the last row produces much superior depth maps than others, which are reflected in the quantitative results in Table I.

Method	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Baseline	0.103	0.740	4.458	0.179	0.896	0.966	0.983
Baseline+DA	0.101	0.725	4.450	0.178	0.896	0.966	0.983
Baseline+SRD	0.100	0.708	4.357	0.176	0.897	0.966	0.984
Baseline+SRD+DA	**0.099**	0.673	4.333	0.175	**0.898**	0.966	0.984
Baseline+SRD+DA+MVC (**full**)	**0.099**	**0.659**	**4.314**	**0.174**	**0.898**	**0.967**	**0.985**
Ours (w/o mask)	**0.099**	0.673	4.333	0.175	**0.898**	0.966	0.984
Ours (w/auto mask [6])	0.102	0.707	4.364	0.175	0.895	**0.967**	0.984
Ours (w/self-discoverd mask [73])	0.102	0.667	4.356	0.175	0.892	0.966	**0.985**
Ours (w/soft mask)	0.102	0.665	4.351	0.174	0.890	0.965	**0.985**
Ours (w/hard mask)	**0.099**	**0.659**	**4.314**	**0.174**	**0.898**	**0.967**	**0.985**
Baseline (ResNet18)	0.115	0.903	4.863	0.193	0.877	0.959	0.981
Ours (ResNet18)	0.111	0.762	4.619	0.186	0.877	0.961	0.983
Baseline (ResNet50)	0.110	0.831	4.642	0.187	0.883	0.962	0.982
Ours (ResNet50)	0.106	0.718	4.520	0.180	0.886	0.964	0.983
Baseline (Swin-T)	0.109	0.814	4.636	0.185	0.888	0.963	0.982
Ours (Swin-T)	0.105	0.686	4.493	0.178	0.885	0.964	**0.985**
Baseline (MPViT-S)	0.103	0.740	4.458	0.179	0.896	0.966	0.983
Ours (MPViT-S)	**0.099**	**0.659**	**4.314**	**0.174**	**0.898**	**0.967**	**0.985**

TABLE II

ABLATION RESULTS. The baseline model is Monodepth2 [6], which is replaced backbone by MPViT-S [47] .DA denotes disparity alignment, SRD denotes self-reference distillation, and MVC denotes mutilview check.

RMSE, RMSE log, $\delta_1 < 1.25$, $\delta_2 < 1.25^2$, $\delta_3 < 1.25^3$) proposed in [16] and we cap depth to 80 m per standard practice [52]. $\delta < t$: % of $\mathbf{d}$ satisfies $\left(\max \left(\frac{\widehat{\mathbf{d}}}{\mathbf{d}}, \frac{\mathbf{d}}{\widehat{\mathbf{d}}} \right) = \delta < t \right)$ for $t = 1.25, 1.25^2, 1.25^3$.

- $AbsRel = \frac{1}{|\mathbf{T}|} \sum_{\widehat{\mathbf{d}} \in \mathbf{T}} \left| \widehat{\mathbf{d}} - \mathbf{d} \right| / \mathbf{d}$,
- $SqRel = \frac{1}{|\mathbf{T}|} \sum_{\widehat{\mathbf{d}} \in \mathbf{T}} \left\| \widehat{\mathbf{d}} - \mathbf{d} \right\|^2 / \mathbf{d}$,
- $RMSE = \sqrt{\frac{1}{|\mathbf{T}|} \sum_{\widehat{\mathbf{d}} \in \mathbf{T}} \left\| \widehat{\mathbf{d}} - \mathbf{d} \right\|^2}$,

- $RMSElog = \sqrt{\frac{1}{|\mathbf{T}|} \sum_{\widehat{\mathbf{d}} \in \mathbf{T}} \left\| \log_{10} \widehat{\mathbf{d}} - \log_{10} \mathbf{d} \right\|^2}$,

We compare the results of several variants of our model trained with different types of supervision. In Table I, Se denotes training with semantic labels, D for depth supervision, D* for auxiliary depth supervision, and M for mono self-supervision. On evaluation metrics, our model (MPViT-S) outperforms most methods (e.g. [16, 58–60]) when compared to the depth supervision method. Other metrics outperform the DVSO [61], with the exception of the Abs Rel metric, which is slightly lower. According to the experimental findings,
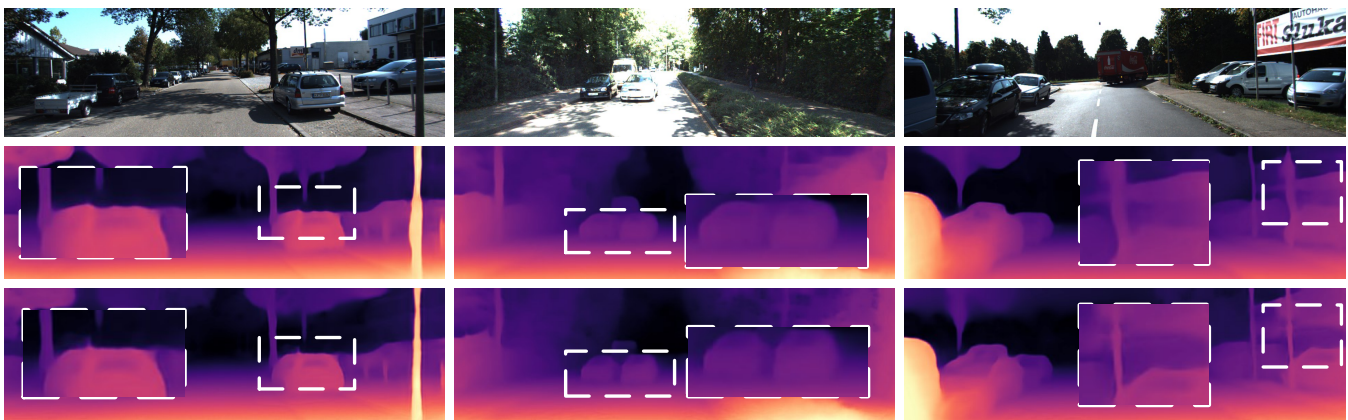
Fig. 8. Visualization results of w/ DA and w/o DA. From the top to bottom, the RGB images, the predicted depth maps without DA (disparity alignment) and the predicted depth maps with DA. The area inside the white box clearly shows that the depth prediction performance is better with DA than without DA.

Distillation method	Traning time	Params	FLOPs	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
w/oDistillation	42.3h	27.9M	20.6G	0.101	0.725	4.450	0.178	0.896	0.966	0.983
w/[9]	74.7h	55.8M	34.8G	0.100	0.695	4.373	0.175	0.897	0.966	0.984
w/[8]	75.9h	54.5M	38.0G	0.100	0.695	4.337	0.176	0.897	0.965	0.983
w/[7]	92.5h	107.6M	46.8G	0.100	0.675	**4.302**	**0.174**	0.896	0.966	0.984
w/SRD (Ours)	**43.4h**	**50.5M**	**23.5G**	**0.099**	**0.659**	4.314	**0.174**	0.898	**0.967**	**0.985**

TABLE III

COMPARISON OF THE RESULTS OF DIFFERENT DISTILLATION METHODS. [7–9] is a two-stage distillation, and SRD (self-reference distillation) is a single-stage online distillation. The Params and FLOPs are calculated according to the input through a forward function of model.
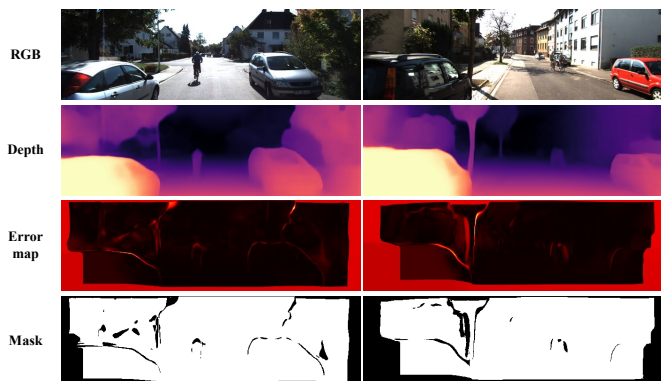

Fig. 9. Visualization of the filter mask. In the fourth row, the black area is filtered out, and the white area is retained.

our method closes the gap between supervised and self-supervised monocular depth estimation. In comparison with self-supervised depth estimation methods, ours has a considerable improvement on most evaluation metrics compared to our baseline model Monodepth2 [6]. Our accuracy is superior to that of the newly proposed approaches, such as MonoFormer [67], CADepth [21], and DIFFNet [23] in all metrics. We also contrast the most advanced approach currently available, MonoViT [22]. Our results on Abs Rel and δ_1 are comparable, but we perform better on other measures, particularly Sq Rel. On a higher resolution of 1024×320, the accuracy of the model has been further improved. Our monocular depth estimation method demonstrates superior performance compared to state-of-the-art approaches, as evidenced by the results presented in

Table I.

We provide a comparison chart of the visualized outcomes, as seen in Fig.7, to present our results more intuitively. We indicate with the white dotted box where our method performs better than other methods. For example, in the visualization results, it can be found that MonoViT [22] does not perform well in the depth estimation of car mirrors, and Monodepth2 [6] does not perform well in overlapping pedestrian occlusion areas.

D. Generalization Evaluation

We conduct generalization experiments on the Make3D dataset [74]. Our model is directly used for the test of the make3D test set without any fine-tuning after training in the KITTI dataset [69]. When evaluating metrics, we maintain consistent data preprocessing on a center crop of 2×1 ratio with [6, 21–23].

In our generalization testing experiments, we conduct experiments on a test set with 134 samples. We compare some supervised methods with state-of-the-art unsupervised methods. In the generalization of Make3D, our method performs better than the supervised method developed by [16] and [58]. Other metrics are marginally inferior when compared to Laina's supervised method [75], but our method performs better on the Sq Rel evaluation metric. Our method achieves the best results compared to self-supervised monocular depth estimation methods, and our Sq Rel metric shows the greatest improvement, which is reflected not only in the generalized experiments, but also in the ablation experiments. The results in Table IV demonstrated that, in terms of generalization
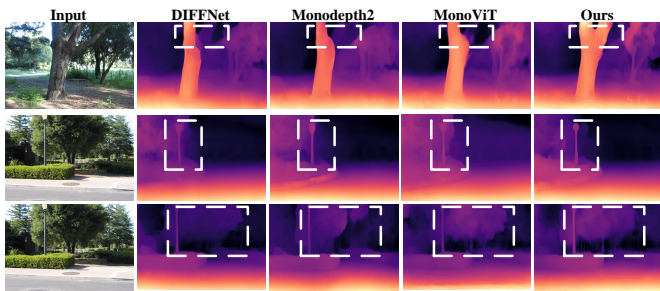
Fig. 10. Qualitative comparison on the Make3D [74] dataset. Predictions by DIFFNet [23], Monodepth2 [6], MonoViT[22] and ours.

Method	Abs Rel	Sq Rel	RMSE	log_{10}
Eigen†[16]	0.428	5.079	8.389	0.149
Liu†[58]	0.475	6.562	10.05	0.165
Laina†[75]	**0.204**	**1.840**	**5.683**	**0.084**
Monodepth [52]	0.544	10.94	11.760	0.193
Zhou [19]	0.383	5.321	10.470	0.478
DDVO [55]	0.387	4.720	8.090	0.204
Monodepth2 [6]	0.322	3.589	7.417	0.163
CADepth [21]	0.319	3.564	7.152	0.158
HR-Depth [7]	0.305	2.944	6.857	0.157
MonoViT [22]	0.286	2.758	6.623	0.147
Ours	**0.252**	**1.583**	**5.833**	**0.114**

TABLE IV

GENERALIZATION RESULTS. The method marked (†) denotes the depth supervision method, and the other methods are self-supervised.

capacity, our model is much superior to the existing state-of-the-art approaches. Fig.10 displays a depiction of the generalization visualization. We mark the areas worthy of attention with white dotted boxes, and it can also be seen from the visualizations that our method outperforms the others.

E. Ablation Study

To evaluate the effectiveness of each module of the model, we conduct ablation experiments on the KITTI [69] dataset. Our benchmark model is Monodepth2 [6], but for a fair experimental comparison, we replace the backbone network of Monodepth2 from ResNet [13] with MPViT-S [47]. Based on the benchmark model, we conduct ablation experiments on three modules, namely, self-reference distillation (SRD), multiview check (MVC) and disparity alignment (DA). The experimental results confirm the effectiveness of the three modules we proposed. From the results in Table II, it can be found that the self-reference distillation has a significant improvement in the Sq Rel metric.

To verify the effectiveness of the proposed mask (hard mask), we compare against other masking strategies, including the proposed mask (soft mask), the self-discovered mask [73] and the auto mask [6]. The soft mask and the discovered mask are similar in form, but the difference is that the soft mask is used to re-weight depth supervision loss in distillation and the discovered mask is used to re-weight photometric

loss in self-supervised learning. From the experimental results, the strategy of re-weighting the loss improves the Sq Rel metric, but other metrics decrease. The auto mask and the proposed mask (hard mask) both use a binarized mask. The former sets mask to only include the loss of pixels where the reprojection error of the warped image is lower than of the original, unwrapped source image, which is used to remove the motion information in the self-supervised training. The latter filters out outliers with large depth errors output by the teacher model during the distillation process. The distribution of these two masks is not consistent. The former is sparsely distributed in the whole image, while the latter is mainly distributed in the border around the image, and a small part is in the middle area of the image. Compared with these three masks, our proposed mask (hard mask) achieves the best performance among all metrics, as shown in the Table II.

We compare different backbone networks. We use ResNet [13], Swin Transformer [14], and MPViT [47] as the backbone networks to train the model. We selected several models with relatively small and similar numbers of parameters. These models include ResNet18 with 11.7M parameters, ResNet50 with 25 M parameters, Swin Transformer-T with 28 M parameters, and MPViT-S with 22.8 M parameters. The experimental findings in Table II reveal that our method has considerably enhanced most metrics when compared to the baseline model using the same backbone network, and the model using MPViT as the backbone network has achieved the greatest performance outcomes.

In addition, we compare different distillation method [7–9] to verify the effectiveness of our self-reference distillation. We compare the computational cost and quantitative results of these methods as shown in Table III. In terms of computational costs, we compare the training time, the parameters and the FLOPs, where the parameters and the FLOPs are calculated according to the input through a forward function of model. [7–9] are two-stage distillations, which need to complete the training of the teacher model first, and then distill the student model. The entire distillation process requires two stages of training to be completed separately, increasing the training time, while our single-stage online distillation method basically does not increase training time. [7] builds complex teacher model to distill, so the parameters and FLOPs are large. [8] uses ensemble strategy but for the fair comparison, it remains consistent with other methods using a single teacher model. The parameters of [8, 9] are comparable to our method but their FLOPs are still larger than ours. Overall, our method performs better in quantitative results with minimal computational costs.

V. CONCLUSION

In this paper, we propose a novel self-supervised monocular depth estimation method, employing self-reference distillation to provide depth supervision signals for the student model and introducing a multiview check filter to filter outliers in depth maps. In addition, we propose the disparity offset to solve the disparity misalignment problem caused by the upsampling process. Extensive experiments are carried out on

two challenging datasets, including the KITTI and Make3D datasets. The experimental results emphasize the effectiveness and strong generalization of our method.

REFERENCES

[1] Jiancai Huang, Zhaohui Jiang, Weihua Gui, Zunhui Yi, Dong Pan, Ke Zhou, and Chuan Xu. Depth estimation from a single image of blast furnace burden surface based on edge defocus tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(9):6044–6057, 2022.

[2] Minsoo Song, Seokjae Lim, and Wonjun Kim. Monocular depth estimation using laplacian pyramid-based depth residuals. *IEEE transactions on circuits and systems for video technology*, 31(11):4381–4393, 2021.

[3] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Adabins: Depth estimation using adaptive bins. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4009–4018, 2021.

[4] Weihao Yuan, Xiaodong Gu, Zuozhuo Dai, Siyu Zhu, and Ping Tan. New crfs: Neural window fully-connected crfs for monocular depth estimation. *arXiv preprint arXiv:2203.01502*, 2022.

[5] Xuyang Meng, Chunxiao Fan, Yue Ming, and Hui Yu. Cornet: Context-based ordinal regression network for monocular depth estimation. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(7):4841–4853, 2022.

[6] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3828–3838, 2019.

[7] Xiaoyang Lyu, Liang Liu, Mengmeng Wang, Xin Kong, Lina Liu, Yong Liu, Xinxin Chen, and Yi Yuan. Hr-depth: High resolution self-supervised monocular depth estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 2294–2301, 2021.

[8] Weisong Ren, Lijun Wang, Yongri Piao, Miao Zhang, Huchuan Lu, and Ting Liu. Adaptive co-teaching for unsupervised monocular depth estimation. In *European Conference on Computer Vision*, pages 89–105. Springer, 2022.

[9] Matteo Poggi, Filippo Aleotti, Fabio Tosi, and Stefano Mattoccia. On the uncertainty of self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3227–3237, 2020.

[10] Shihua Huang, Zhichao Lu, Ran Cheng, and Cheng He. Fapn: Feature-aligned pyramid network for dense image prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 864–873, 2021.

[11] Xiangtai Li, Ansheng You, Zhen Zhu, Houlong Zhao, Maoke Yang, Kuiyuan Yang, Shaohua Tan, and Yunhai Tong. Semantic flow for fast and accurate scene parsing. In *European Conference on Computer Vision*, pages 775–793. Springer, 2020.

[12] Wencheng Han, Junbo Yin, Xiaogang Jin, Xiangdong Dai, and Jianbing Shen. Brnet: Exploring comprehensive features for monocular depth estimation. In *European Conference on Computer Vision*, pages 586–602. Springer, 2022.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[14] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021.

[15] Ashutosh Saxena, Sung Chung, and Andrew Ng. Learning depth from single monocular images. *Advances in neural information processing systems*, 18, 2005.

[16] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *Advances in neural information processing systems*, 27, 2014.

[17] Jin Han Lee, Myung-Kyu Han, Dong Wook Ko, and Il Hong Suh. From big to small: Multi-scale local planar guidance for monocular depth estimation. *arXiv preprint arXiv:1907.10326*, 2019.

[18] Shuwei Shao, Ran Li, Zhongcai Pei, Zhong Liu, Weihai Chen, Wentao Zhu, Xingming Wu, and Baochang Zhang. Towards comprehensive monocular depth estimation: Multiple heads are better than one. *IEEE Transactions on Multimedia*, 2022.

[19] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1851–1858, 2017.

[20] Yuliang Zou, Zelun Luo, and Jia-Bin Huang. Df-net: Unsupervised joint learning of depth and flow using cross-task consistency. In *Proceedings of the European conference on computer vision (ECCV)*, pages 36–53, 2018.

[21] Jiaxing Yan, Hong Zhao, Penghui Bu, and YuSheng Jin. Channel-wise attention-based network for self-supervised monocular depth estimation. In *2021 International Conference on 3D Vision (3DV)*, pages 464–473. IEEE, 2021.

[22] Chaoqiang Zhao, Youmin Zhang, Matteo Poggi, Fabio Tosi, Xianda Guo, Zheng Zhu, Guan Huang, Yang Tang, and Stefano Mattoccia. Monovit: Self-supervised monocular depth estimation with a vision transformer. *arXiv preprint arXiv:2208.03543*, 2022.

[23] Hang Zhou, David Greenwood, and Sarah Taylor. Self-supervised monocular depth estimation with internal feature fusion. *arXiv preprint arXiv:2110.09482*, 2021.

[24] Lina Liu, Xibin Song, Mengmeng Wang, Yong Liu, and Liangjun Zhang. Self-supervised monocular depth estimation for all day images using domain separation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12737–12746, 2021.

[25] Michaël Ramamonjisoa, Michael Firman, Jamie Watson, Vincent Lepetit, and Daniyar Turmukhambetov. Single image depth prediction with wavelet decomposition. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11084–11093, 2021.

[26] Shu Chen, Zhengdong Pu, Xiang Fan, and Beiji Zou. Fixing defect of photometric loss for self-supervised monocular depth estimation. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(3):1328–1338, 2021.

[27] Jaehoon Choi, Dongki Jung, Donghwan Lee, and Changick Kim. Safenet: Self-supervised monocular depth estimation with semantic-aware feature extraction. *arXiv preprint arXiv:2010.02893*, 2020.

[28] Vitor Guizilini, Rares Ambrus, Sudeep Pillai, Allan Raventos, and Adrien Gaidon. 3d packing for self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2485–2494, 2020.

[29] Akhil Gurram, Ahmet Faruk Tuna, Fengyi Shen, Onay Urfalioglu, and Antonio M López. Monocular depth estimation through virtual-world supervision and real-world sfm self-supervision. *IEEE Transactions on Intelligent Transportation Systems*, 23(8):12738–12751, 2021.

[30] Jaime Spencer, Richard Bowden, and Simon Hadfield. Defeat-net: General monocular depth via simultaneous unsupervised representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14402–14413, 2020.

[31] Anurag Ranjan, Varun Jampani, Lukas Balles, Kihwan Kim, Deqing Sun, Jonas Wulff, and Michael J Black. Competitive collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation. In *Proceedings*

of the IEEE/CVF conference on computer vision and pattern recognition, pages 12240–12249, 2019.

[32] Zhenheng Yang, Peng Wang, Yang Wang, Wei Xu, and Ram Nevatia. Lego: Learning edge with geometry all at once by watching videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 225–234, 2018.

[33] Zhenheng Yang, Peng Wang, Wei Xu, Liang Zhao, and Ramakant Nevatia. Unsupervised learning of geometry with edge-aware depth-normal consistency. *arXiv preprint arXiv:1711.03665*, 2017.

[34] Vitor Guizilini, Rui Hou, Jie Li, Rares Ambrus, and Adrien Gaidon. Semantically-guided representation learning for self-supervised monocular depth. *arXiv preprint arXiv:2002.12319*, 2020.

[35] Marvin Klingner, Jan-Aike Termöhlen, Jonas Mikolajczyk, and Tim Fingscheidt. Self-supervised monocular depth estimation: Solving the dynamic object problem by semantic guidance. In *European Conference on Computer Vision*, pages 582–600. Springer, 2020.

[36] Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541, 2006.

[37] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015.

[38] Yuncheng Li, Jianchao Yang, Yale Song, Liangliang Cao, Jiebo Luo, and Li-Jia Li. Learning from noisy labels with distillation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1910–1918, 2017.

[39] Guobin Chen, Wongun Choi, Xiang Yu, Tony Han, and Manmohan Chandraker. Learning efficient object detection models with knowledge distillation. *Advances in neural information processing systems*, 30, 2017.

[40] Saurabh Gupta, Judy Hoffman, and Jitendra Malik. Cross modal distillation for supervision transfer. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2827–2836, 2016.

[41] Tommaso Furlanello, Zachary Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born again neural networks. In *International Conference on Machine Learning*, pages 1607–1616. PMLR, 2018.

[42] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10687–10698, 2020.

[43] Zeyu Cheng, Yi Zhang, and Chengkai Tang. Swin-depth: Using transformers and multi-scale fusion for monocular-based depth estimation. *IEEE Sensors Journal*, 21(23):26912–26920, 2021.

[44] Guanglei Yang, Hao Tang, Mingli Ding, Nicu Sebe, and Elisa Ricci. Transformer-based attention networks for continuous pixel-wise prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16269–16279, 2021.

[45] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12179–12188, 2021.

[46] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 568–578, 2021.

[47] Youngwan Lee, Jonghee Kim, Jeffrey Willette, and Sung Ju Hwang. Mpvit: Multi-path vision transformer for dense prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7287–7296, 2022.

[48] Jiaxing Yan, Hong Zhao, Penghui Bu, and YuSheng Jin. Channel-wise attention-based network for self-supervised monocular depth estimation. In *2021 International Conference on 3D Vision (3DV)*, pages 464–473. IEEE, 2021.

[49] Hang Zhou, David Greenwood, and Sarah Taylor. Self-supervised monocular depth estimation with internal feature fusion. *arXiv preprint arXiv:2110.09482*, 2021.

[50] Zilong Huang, Yunchao Wei, Xinggang Wang, Wenyu Liu, Thomas S Huang, and Humphrey Shi. Alignseg: Feature-aligned segmentation networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(1):550–557, 2021.

[51] Adriano Cardace, Pierluigi Zama Ramirez, Samuele Salti, and Luigi Di Stefano. Shallow features guide unsupervised domain adaptation for semantic segmentation at class boundaries. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1160–1170, 2022.

[52] Clément Godard, Oisin Mac Aodha, and Gabriel J. Brostow. Unsupervised monocular depth estimation with left-right consistency. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6602–6611, 2017.

[53] Ravi Garg, Vijay Kumar Bg, Gustavo Carneiro, and Ian Reid. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *European conference on computer vision*, pages 740–756. Springer, 2016.

[54] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[55] Chaoyang Wang, José Miguel Buenaposada, Rui Zhu, and Simon Lucey. Learning depth from monocular videos using direct methods. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2022–2030, 2018.

[56] Andra Petrovai and Sergiu Nedevschi. Exploiting pseudo labels in a self-supervised learning framework for improved monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1578–1588, 2022.

[57] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European conference on computer vision (ECCV)*, pages 767–783, 2018.

[58] Fayao Liu, Chunhua Shen, Guosheng Lin, and Ian Reid. Learning depth from single monocular images using deep convolutional neural fields. *IEEE transactions on pattern analysis and machine intelligence*, 38(10):2024–2039, 2015.

[59] Maria Klodt and Andrea Vedaldi. Supervising the new with the old: learning sfm from sfm. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 698–713, 2018.

[60] Jogendra Nath Kundu, Phani Krishna Uppala, Anuj Pahuja, and R Venkatesh Babu. Adadepth: Unsupervised content congruent adaptation for depth estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2656–2665, 2018.

[61] Nan Yang, Rui Wang, Jorg Stuckler, and Daniel Cremers. Deep virtual stereo odometry: Leveraging deep depth prediction for monocular direct sparse odometry. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 817–833, 2018.

[62] Yue Luo, Jimmy Ren, Mude Lin, Jiahao Pang, Wenxiu Sun, Hongsheng Li, and Liang Lin. Single view stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 155–163, 2018.

[63] Xiaoyang Guo, Hongsheng Li, Shuai Yi, Jimmy Ren, and Xiaogang Wang. Learning monocular depth by distilling cross-domain stereo networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 484–500, 2018.

[64] Zhongkai Zhou, Xinnan Fan, Pengfei Shi, and Yuanxue Xin. R-msfm: Recurrent multi-scale feature modulation for monocular depth estimating. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12777–12786, 2021.

[65] Hang Zhou, David Greenwood, Sarah Taylor, and Han Gong. Constant velocity constraints for self-supervised monocular depth estimation. In *Proceedings of the 17th ACM SIGGRAPH European Conference on Visual Media Production*, pages 1–8, 2020.

[66] Adrian Johnston and Gustavo Carneiro. Self-supervised monocular trained depth estimation using self-attention and discrete disparity volume. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 4756–4765, 2020.

[67] Jinwoo Bae, Sungho Moon, and Sunghoon Im. Monoformer: Towards generalization of self-supervised monocular depth estimation with transformers. *arXiv preprint arXiv:2205.11083*, 2022.

[68] Armin Masoumian, Hatem A Rashwan, Saddam Abdulwahab, Julian Cristiano, and Domenec Puig. Gcndepth: Self-supervised monocular depth estimation based on graph convolutional network. *arXiv preprint arXiv:2112.06782*, 2021.

[69] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013.

[70] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016.

[71] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[72] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[73] Jiawang Bian, Zhichao Li, Naiyan Wang, Huangying Zhan, Chunhua Shen, Ming-Ming Cheng, and Ian Reid. Unsupervised scale-consistent depth and ego-motion learning from monocular video. *Advances in neural information processing systems*, 32, 2019.

[74] Ashutosh Saxena, Min Sun, and Andrew Y Ng. Make3d: Learning 3d scene structure from a single still image. *IEEE transactions on pattern analysis and machine intelligence*, 31(5):824–840, 2008.

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

Ran Li received the bachelor's degree in automation science from BUAA, in 2021, and is currently studying for a master degree with the Beihang University (BUAA), China. His research interests include computer vision and machine learning.

Shuwei Shao received the B.Eng. from Xidian University, Xi'an, Shanxi, China, in 2019. He is currently pursuing the Ph.D. degree in School of Automation Science and Electrical Engineering, Beihang University, Beijing, China. His current research interests include image registration, depth estimation.

Xingming Wu received his B.Eng. and M.Eng. degrees from Zhejiang University, in 1981 and 1985, respectively. He is currently an Professor in the School of Automation Science and Electronic Engineering at Beihang University, China. His main research directions are intelligent sensor systems, embedded systems, autonomous mobile robots, and image processing.

Zhong Liu received the B.Eng., M.Eng. and Ph.D. degrees from the Harbin Institute of Technology, Harbin,China, in 1991, 1994, and 1997, respectively. He has been with the School of Automation Science and Electronic Engineering, Beihang University, as an Associate Professor from 2000 and as a Professor since 2006. He has published over 50 technical papers in referred journals and conference proceedings and field more than 10 patents. His research interests include bionic CPG mechanism and control, computer vision, parallel mechanism and robotics.

Weihai Chen received the B.Eng. degree from Zhejiang University, China, in 1982, and the M.Eng. and Ph.D. degrees from Beihang University, China, in 1988 and 1996, respectively. He has been with the School of Automation Science and Electronic Engineering, Beihang University, as an Associate Professor from 1998 and as a Professor since 2007. He has published over 200 technical papers in referred journals and conference proceedings and field more than 20 patents. His research interests include bio-inspired robotics, computer vision, image processing, precision mechanism, automation, and control.