SENSE: Self-Evolving Learning for Self-Supervised Monocular Depth Estimation

Guanbin Li, *Member, IEEE*, Ricong Huang, Haofeng Li, *Member, IEEE*, Zunzhi You, and Weikai Chen

***Abstract*— Self-supervised depth estimation methods can achieve competitive performance using only unlabeled monocular videos, but they suffer from the uncertainty of jointly learning depth and pose without any ground truths of both tasks. Supervised framework provides robust and superior performance but is limited by the scope of the labeled data. In this paper, we introduce SENSE, a novel learning paradigm for self-supervised monocular depth estimation that progressively evolves the prediction result using supervised learning, but without requiring labeled data. The key contribution of our approach stems from the novel use of the pseudo labels – the noisy depth estimation from the self-supervised methods. We surprisingly find that a fully supervised depth estimation network trained using the pseudo labels can produce even better results than its "ground truth". To push the envelope further, we then evolve the self-supervised backbone by replacing its depth estimation branch with that fully supervised network. Based on this idea, we devise a comprehensive training pipeline that alternatively enhances the two key branches (depth and pose estimation) of the self-supervised backbone network. Our proposed approach can effectively ease the difficulty of multi-task training in self-supervised depth estimation. Experimental results have shown that our proposed approach achieves state-of-the-art results on the KITTI dataset.**

***Index Terms*— Self-supervised, monocular depth estimation, pseudo labels, learning paradigm.**

I. Introduction

IMAGE-BASED depth estimation provides a low-cost manner of sensing the 3D surroundings by only using a commodity camera. It has gained an increasing attention due to its ability to inexpensively complement LiDAR sensors that are widely used in autonomous driving and robotics. Generating high-quality depth-from-color can also enable new applications in mobile devices, such as 3D photo editing and AR compositing. Despite the ill-posed nature of this problem, recent advances in deep neural networks have achieved promising results by learning from a large corpus of paired data. However, collecting per-pixel ground-truth depth data for supervised learning is a challenge, especially at a large scale. As an alternative, recent works on self-supervised learning [1], [2], [3] have shown the possibility of obtaining high-quality depth estimation by using only monocular videos as supervisory signals.

Besides using stereo as supervision, a lot of self-supervised monocular depth estimation methods resort to a multi-task framework that predicts the depth and pose simultaneously. In particular, they typically employ a view synthesis system that encourages the estimation of the scene geometry and camera pose to be consistent with the physical measurements. However, due to the lack of ground truths in both tasks, the training of the multi-task network is particularly challenging and is prone to suffer from inferior local minima, leading to noisy estimation or even artifacts. Moreover, since the depth regression is indirectly supervised by the image reconstruction loss, the supervision flow can be highly sparse especially in the textureless regions. Such uncertainty could cause vulnerability in the system where a small perturbation of the input may lead to dramatic variations in the output depth maps.

The key to resolving the above issues in self-supervised learning is to provide dense per-pixel supervision to remove the uncertainties. Nonetheless, this traces back to the aforementioned dilemma of data collection. To this end, we propose to attack this problem by introducing a novel learning paradigm, coded SENSE, which can proactively evolve the performance of self-supervised backbones with supervised learning. However, our method does not need ground-truth depths for training but only requires monocular videos instead. We achieve this goal by fully exploiting the pseudo labels generated by the self-supervised methods. Our approach is built upon the insight that deep neural networks have a memorization effect and tend to learn clean and simple patterns before overfitting noisy labels [4], [5]. Thus a well-trained deep convolutional network without over-fitting can even correct the artifacts existing in the noisy training samples, *i.e.* the pseudo labels [6], [7]. We verify this observation via extensive

Manuscript received 31 August 2022; revised 29 June 2023 and 8 November 2023; accepted 21 November 2023. Date of publication 25 December 2023; date of current version 29 December 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 62322608, Grant 62102267, and Grant 61976250; in part by the Shenzhen Science and Technology Program under Grant JCYJ20220530141211024; in part by the Open Project Program of State Key Laboratory of Virtual Reality Technology and Systems, Beihang University under Grant VRLAB2023A01; and in part by the Open Project Program of the Key Laboratory of Artificial Intelligence for Perception and Understanding, Liaoning Province (AIPU) under Grant 20230003. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Yulan Guo. *(Corresponding author: Haofeng Li.)*

Guanbin Li, Ricong Huang, and Zunzhi You are with the School of Computer Science and Engineering, Research Institute of Sun Yat-sen University in Shenzhen, Sun Yat-sen University, Guangzhou 510006, China (e-mail: liguanbin@mail.sysu.edu.cn; huangrc3@mail2.sysu.edu.cn; youzunzhi@gmail.com).

Haofeng Li is with the Shenzhen Research Institute of Big Data, The Chinese University of Hong Kong (Shenzhen), Shenzhen 518172, China (e-mail: lhaof@foxmail.com).

Weikai Chen is with Tencent America, Los Angeles, CA 90066 USA (e-mail: chenwk891@gmail.com).

Digital Object Identifier 10.1109/TIP.2023.3338053

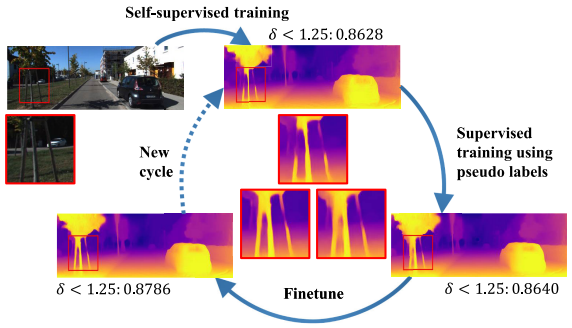


Fig. 1. The self-evolving process of SENSE for depth estimation. We propose a novel training paradigm that can effectively evolve the performance of a self-supervised network with supervised learning. After completed self-supervised training with videos, we leverage its output as pseudo labels for supervised training. This helps remove the artifacts in the pseudo labels and leads to a stronger depth estimation network which is used to replace the depth branch of the self-supervised backbone. We then finetune the entirety of the backbone network with a carefully designed alternate training.

experiments and demonstrate the effectiveness of this idea in Fig. 1 and Tab. IV respectively.

Motivated by the above observations, we propose a novel training framework that can progressively evolve the performance of the self-supervised networks. Following [3], we adopt the self-supervised backbone that estimates depth and pose simultaneously. We first train the backbone network in a self-supervised manner using the monocular videos. The obtained pseudo labels are then used to train a stronger depth network with full supervision. We achieve the first "evolvement" by replacing the depth branch of the backbone with the fully supervised depth network. Afterwards, we employ the self-supervised mechanism again to finetune the pose model while fixing the boosted depth branch. Lastly, we enhance the entirety of the backbone network by combining the self and full supervised paradigms, which accomplishes the second "evolvement". We repeat this process until the network fully converges. In Fig. 1, we show that the proposed self-evolving training strategy can effectively improve the depth estimation in terms of both qualitative and quantitative measurements.

In addition to the self-evolving learning strategy, we also propose a new backbone network, Hierarchical Depth Inference Network (HiDNet). As shown by works [8], [9], and [10], feature resolution is important for dense prediction tasks, like semantic segmentation and depth prediction. High-resolution features can maintain more details of the scene, which is critical for estimating more precise depth of those objects occupying only few pixels in the image. We also consider that when estimating the depth of objects from different distance, human vision could capture the details of different levels. For close objects, a viewer can perceive fine-grained details, but could be misled by more noises. For distant objects, the viewer can obtain smooth and stable estimated depths, but delicate patterns are missed. Inspired by the use of high-resolution feature and Level of Detail (LOD) [11], a computer graphic method that simplifies the mesh as objects become distant from the viewer, we propose a level-of-detail embedding module that yields robust representations by aggregating visual features from short to long distance. The proposed HiDNet is built upon the LOD embedding modules. Using the HiDNet alone without the self-evolving framework can already achieve better results than the state-of-the-art.

We summarize our contributions as follows:

- We introduce a novel self-evolving learning paradigm that advocates a novel use of pseudo labels and can progressively evolve the performance of self-supervised monocular depth estimation using supervised learning, but without the need of labeled data.
- We develop a new level-of-detail (LOD) embedding module to harvest robust depth features, and implement a hierarchical depth inference network based on the proposed LOD module.
- We validate that the proposed framework considerably surpasses the existing self-supervised counterparts in the challenging KITTI dataset.

II. Related Work

A. Self-Supervised Monocular Depth Estimation

Self-supervised monocular depth estimation methods [3], [12], [13], [14], [15] adopt the photometric differences between a reprojected source view and the target view as supervisory signals. Prior methods in this direction can be roughly divided into using adjacent monocular frames [3], [16] and using stereo pairs [17], [18]. We focus on the former one which is more flexible in practise. Since Zhou et al. [3] propose an effective self-supervised structure-from-motion (SfM) pipeline, most recent works aim at improving the self-supervised SfM pipeline by designing novel loss functions, including a 3D-based loss [16], a minimum reprojection loss and an auto-masking loss [1], and a feature-metric loss [19], or building new neural network modules, including a 3D packing module [2], a feature fusion Squeeze-and-Excitation module [9], a self-attention module [20], a transformer-based module [21], and orthogonal planes [22]. Moreover, Feng et al. [23] introduce semantic information to deal with object motion and occlusion. Following [3], these methods learn a single-view depth model and a camera pose estimator at one stage, while few of them consider a multi-stage training scheme. Poggi et al. [24] train the depth network in 2 stages based on the modelling of uncertainty. Ren et al. [25] propose to distill the learned knowledge from teacher networks to a student network through an ensemble architecture. Petrovai and Nedevschi [26] propose a two-stage training strategy to resolve inter-frame scale consistency, which is similar to our work. However, their applicable scenarios are limited. Different from existing methods, we propose a stage-wise learning framework that progressively trains the depth network and the pose network in a self-evolving way. Some works [27], [28] distill knowledge via left and right disparity maps. Peng et al. [29] propose a data grafting strategy, which requires stereo image pairs as input. To release this limitation, our method distills knowledge based on the SfM framework, and adopts a more flexible setting only using monocular images. As for depth network architectures, most of the existing methods [1], [2], [19] employ an encoder-decoder network that reduces and recovers the resolution of features

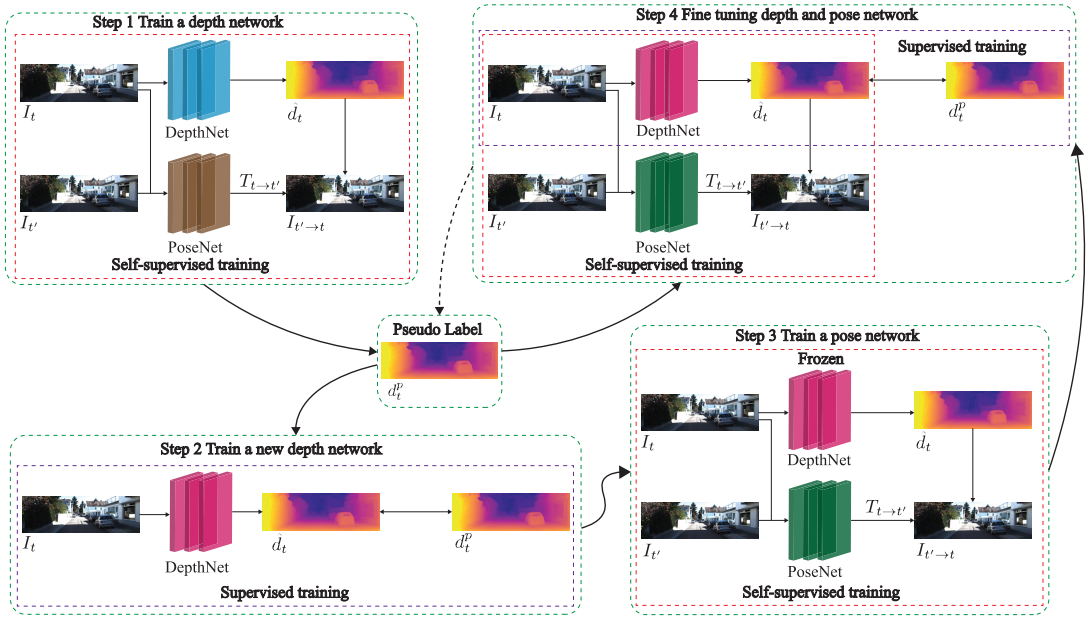


Fig. 2. Overview of the proposed Self-evolving Learning Framework. The framework consists of 4 steps. Notice that pseudo labels can be used for the training in Step 2 and Step 4. Step 4 could update the pseudo labels and form an iterative loop.

in one stream, while our proposed depth network learns high-resolution features with multi-resolution fusion.

B. Supervised Monocular Depth Estimation

Supervised monocular depth estimation methods usually train a fully convolutional network to regress pixel-wise depth values with LiDAR-captured ground truths. Eigen et al. [30] first tackle the supervised depth estimation from a single image by using a coarse global network and a local refinement network. Fu et al. [31] build a spacing-increasing strategy for depth discretization and cast depth perception as an ordinal regression task. Luo et al. [32] use a network to synthesize the right image from a left one, and a stereo matching network to predict depth with the image pair. The other recent advances are achieved with semantic information [33], [34], [35], [36], domain adaptation [37], [38], [39], geometric constraints [40] and new network architectures [41], [42]. In particular, Guo et al. [37] train a supervised stereo network with synthetic data and a monocular network by distilling the stereo network, which is most related to our work. They utilize the supervised stereo network to produce virtual disparity maps, while we synthesize pseudo labels with a self-supervised monocular network.

C. Self-Training for Visual Perception

Self-training is to infer pseudo labels on unlabeled data and then train a model with both real and pseudo labels [43]. It has been widely used in semi-supervised learning. Arpit et al. [4] point out that a deep neural network (DNN) without overfitting tends to learn the simple and general patterns shared by the training samples since the noises are harder for a DNN to fit. And some methods [44], [45] achieve great performance in image classification through learning with noisy labels. Even though real depth labels are not available, pseudo depth labels could derive from non-learning based algorithms [46], [47] or self-supervised learning based models [48]. Klodt and Vedaldi [46] apply a traditional structure-from-motion algorithm to produce proxy ground truths of depth and ego-motion, while we adopt a self-supervised learned depth network to yield pseudo labels and update them iteratively.

III. Method

The proposed self-evolving learning framework is a stage-wise method, as shown in Fig. 2. Sec. III-A describes a self-supervised training pipeline that is a prerequisite of the framework. Sec. III-B details each stage of the self-evolving framework. In Sec. III-D and Sec. III-E, we introduce a hierarchical depth inference network and the training loss.

A. Self-Supervised Training

The following discusses how we achieve depth perception with raw monocular videos. Let $\langle I_1, \ldots, I_N \rangle$ denote the frames of a video. Due to the lack of ground truth data, we leverage the view synthesis error to train a depth prediction model D and a camera pose estimator P. The goal of view synthesis is to reconstruct a target image I_t from its adjacent frame (referred to as source image) $I_{t'}$. The reconstructed view is denoted as $I_{t'\to t}$ which is calculated with the depth of target image I_t and the relative camera pose $\hat{T}_{t\to t'} = P(I_t, I_{t'})$, following [3]. We denote the depth of I_t as $\hat{D}_t = D(I_t)$. As p_t indicates the homogeneous coordinates of a pixel in the target image, $\hat{D}_t(p_t)$ returns the depth of p_t. Let $p_{t'}$ stand for p_t's projected coordinates in the source image $I_{t'}$. Hence, $p_{t'}$ is calculated as:

$$p_{t'} \sim K \hat{T}_{t\to t'} \hat{D}_t(p_t) K^{-1} p_t, \tag{1}$$

where K denotes the intrinsic matrix of the camera which is known in our pipeline. We calculate $p_{t'}$ for each p_t, and use $p_{t'}$ to sample $I_{t'\to t}$ from $I_{t'}$ using bilinear interpolation. Following [1] and [17], we combine structural similarity

(SSIM) [49] with L1 loss as a photometric loss to measure the view synthesis error as supervisory signals:

$$L_{ph}\left(I_t, I_{t'\to t}\right) = \frac{\alpha}{2}\left(1 - \mathrm{SSIM}\left(I_t, I_{t'\to t}\right)\right) + (1-\alpha)\left|I_t - I_{t'\to t}\right|, \tag{2}$$

where α is a weighting factor. We further apply a smoothness loss [17] to encourage the continuity of the predicted depth, as shown in: $L_s\left(\hat{d}_t^*\right) = |\partial_x \hat{d}_t^*| e^{-|\partial_x I_t|} + |\partial_y \hat{d}_t^*| e^{-|\partial_y I_t|}$, where ∂_x and ∂_y denote the gradients in horizontal and vertical direction respectively. $\hat{d}_t^* = \hat{d}_t / \overline{\hat{d}_t}$ denotes an inverse depth map normalized by its mean.

Considering that the presence of occlusions could lead to a large photometric loss that fails to measure the quality of depth. Inspired by [1] and [3], we adopt two adjacent frames of the target image as source images. For each pixel, only the smaller error in the two source images is chosen as the final loss to help reduce the affect of occlusion. Since the self-supervised monocular training assumes that a camera moves in a static scene, an auto-mask [1] is utilized to filter out stationary pixels. Thus, the masked photometric loss is computed as:

$$L_p = M_{auto} \odot \min_{t' \in \{t-1, t+1\}} L_{ph}\left(I_t, I_{t'\to t}\right), \tag{3}$$

where $\odot$ is element-wise multiplication, M_{auto} is a mask that sets the loss of stationary pixels as zeros. M_{auto} is defined as:

$$M_{auto} = \left[\min_{t'} L_{ph}\left(I_t, I_{t'\to t}\right) < \min_{t'} L_{ph}\left(I_t, I_{t'}\right)\right], \tag{4}$$

where $[*]$ returns 1 if the input is true. Otherwise, it returns 0. The overall loss (see Sec. III-E) is a combination of L_p, L_s, and a pseudo-label loss L_{pse} (see Sec. III-B). By minimizing the loss, the gradients are passed through $\hat{T}_{t\to t'}$ and $\hat{D}_t\left(p_t\right)$ to train the pose network P and the depth model D.

B. Self-Evolving Learning

Existing self-supervised monocular methods usually train the depth network and the ego-motion estimator simultaneously. Instead, we propose to learn the depth and the pose estimation tasks progressively in a self-evolving manner, for better training results. As shown in Fig. 2, the proposed self-evolving learning framework consists of four steps. First, we employ the self-supervised pipeline described in Sec. III-A to train a depth network D' and a pose network P' at once. For each video frame in the train set, the depth network D' infers a disparity map $\hat{d}'$. Pseudo labels $\{d^p\}$ (p for "pseudo") are obtained by enhancing the disparity maps $\hat{d}'$ with a post-refinement [17].

In the second step, we train another depth network D'' from scratch with the proxy ground truth $\{d^p\}$. Since a pseudo disparity map is noisy, only the disparity of some positions are beneficial for training. To locate these useful pseudo labels, we use a Gaussian Mixture Model to compute a confidence map P (see Sec. III-C) for each pseudo label map. The proposed pseudo-label loss is defined as:

$$L_{pse} = P \odot L_{Berhu},$$

$$L_{Berhu}\left(\hat{d}'', d^p\right) = \begin{cases} |\hat{d}'' - d^p|, & |\hat{d}'' - d^p| \le c \\ \dfrac{\left(\hat{d}'' - d^p\right)^2 + c^2}{2c}, & |\hat{d}'' - d^p| > c, \end{cases}$$
$$c = 0.2 \cdot \max_x \left(|\hat{d}''_x - d^p_x|\right) \tag{5}$$

where L_{Berhu} denotes a Berhu loss [36], [50], [51], [52]. $\hat{d}''$ is a disparity map output by D''. d^p is the corresponding pseudo ground truth. x is the coordinate of an arbitrary pixel in $\hat{d}''$. Incorrect depth in pseudo labels can be viewed as the noise of the pseudo-label, which is usually harder for the model to fit than the normal pixels that meet the contextual depth estimation inference rules. From the perspective of model generalization, we believe that a model well-trained with a large number of noisy pseudo-labels without overfitting usually learns the universal inference of depth estimation, and thus has the potential of label denoising. Besides, the pseudo labels can become more accurate after the post-processing [17] and more robust with the confidence map in Eq. (5). Thus, it is likely that after the supervised training with pseudo labels, the depth network D'' could obtain better results than D'.

In the third step, we resort to the self-supervised pipeline again, and train another pose network P'' from scratch by fixing the parameters of the depth network D''. Since D'' performs better depth perception than D', then P'' is likely to infer more accurate ego-motion than P' according to Eq. (1). In the last step, we combine the self-supervised view synthesis loss with the supervised Berhu loss based on the pseudo labels $\{d^p\}$. Such a joint loss is used to simultaneously finetune the depth network D'' and the pose network P''. The updated variants of depth networks and pose networks are denoted as D^* and P^* respectively. After updating pseudo ground truths with the depth network D^*, the last step is connected to the second step and forms a closed loop. The iterative process continues until the depth prediction of D^* converges.

C. Pseudo Label Refinement

It is uneasy to obtain accurate depths for some image regions, such as object boundaries. Therefore, the pseudo depth labels on these regions could be noisy. To resolve this problem, we follow [53] and adopt a two-component Gaussian Mixture Model (GMM) to calculate the confidence of each position in the pseudo label maps. For each pixel position, the GMM measures the possibility that the pseudo label of the position is clean. The GMM is fitted to the $L_{ph,i}$ of each image i in the training set using the Expectation-Maximization algorithm. The pseudo label depth d_i^p is used when calculating $L_{ph,i}$. Then the probability of each position being a clean label is collected to form a confidence map. For each position (i, j) in a pseudo label map, the confidence $P_{i,j}$ is the posterior probability $p(g|L_{ph,i})$, where g is the Gaussian component with smaller mean. The photometric loss L_p and Berhu loss L_{Berhu} are weighted by multiplying with the confidence map so that the network concentrates on the reliable positions.

D. Hierarchical Depth Inference Network

Most of the depth estimation networks [3], [17], [54] adopt a U-shape architecture with an encoder-decoder network.

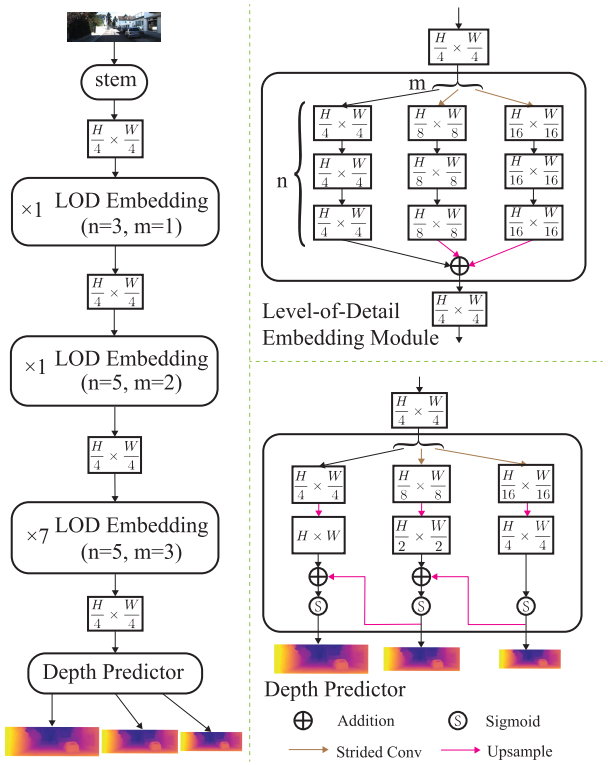


Fig. 3. Architecture of the proposed hierarchical depth inference network and the proposed level-of-detail embedding module.

The encoder gradually increases the channel number of image features to capture higher-level semantics at the cost of lower resolution. The decoder then recovers the feature resolution by fusing low and high-level features. However, the precise spatial information decreases in the encoder with the lower resolution of features and is hard to recover in the decoder. In fact, Ranftl et al. [10] also point out that feature resolution and granularity are critical for depth prediction.

To solve the issue of the U-shape encoder-decoder, HR-Net [8] is proposed to achieve high-resolution visual recognition. In HR-Net, four parallel streams learn and maintain image features of four different resolutions, which help describe the objects of different scales for semantic understanding tasks. However, in the task of depth estimation, we argue that, compared to the high-level semantic of multi-scale objects, stable low-level features with details are more important in determining the final depth quality. Thus, to achieve high-resolution depth estimation, we propose to maintain only one main feature stream of high-resolution, which is different from HR-Net that learns four parallel feature streams in the meanwhile. To enhance and stabilize the high-resolution feature map, we apply a series of multi-resolution fusion modules proposed in the following to suppress the feature noises and to aggregate the details observed from far to near distance.

In the self-supervised monocular depth estimation task, a photometric loss is commonly used [54], [55]. However, the loss may lead to noisy signals, due to the lack of ground truth. To overcome the difficulty, we resort to the idea of Level of Detail (LOD) [11]. LOD refers to the complexity of a 3D

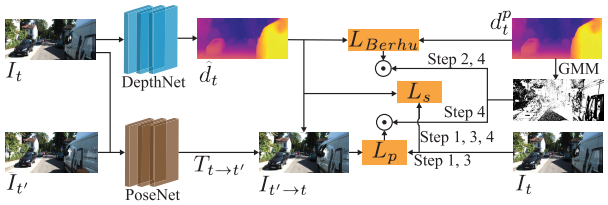


Fig. 4. The training loss functions. L_p, L_s, L_{Berhu} denote the masked photometric loss, the smoothness loss and the Berhu loss.

model representation in computer graphics. When rendering an object distant from the camera, the vertex number of mesh can be reduced by the LOD technique. The rationale is that the fine details of distant objects are usually missed by the viewer and the perceived size of the objects scales inversely with the distance [56]. When human subjects estimate the depth of long-range objects, they could neglect noisy signals and perceive smooth depths. For short-range objects, human vision can capture fine-grained textures as well as the patterns of high uncertainty. Thus we propose a level-of-detail embedding module that first simulates visual features at different distance by applying multiple neural network branches of different resolutions to an input high-resolution feature map, and then merges these multi-resolution outputs into an enhanced depth representation as the module output.

1) Level-of-Detail Embedding Module: In each module, a feature with resolution of $\frac{H}{4} \times \frac{W}{4}$ is the input and the output is of the same resolution as its input. In the main body, the feature will be transformed into m different branches which simulate the information observed from m different distances. In each branch, the resolution of the feature halves gradually. The lower-resolution features are obtained by one or more stride-2 3×3 convolutions and their channels are doubling gradually. Then those features with different resolutions will have n times of convolution units to enhance themselves. After that, the lower-resolution features will be upsampled to the size of $\frac{H}{4} \times \frac{W}{4}$. All those features will be fused by element-wise addition and then output.

2) Depth Predictor: In the depth predictor, the feature with resolution of $\frac{H}{4} \times \frac{W}{4}$ will be transformed into 3 different branches. In each branch, the resolution of feature halves gradually and then will be upsampled. For the output of each branch $\{O_{t,n}, n = 1, 2, 3\}$, the current estimation $\hat{d}_{t,n}$ is obtained by using the last estimation $\hat{d}_{t,n-1}$: $\hat{d}_{t,n} = \text{Sigmoid}\left(O_{t,n} + \hat{d}_{t,n-1}\right)$, $\hat{d}_{t,0} = 0$ especially.

3) Overall Architecture: We build a hierarchical depth inference network (HiDNet) based on the LOD embedding module, as shown in Fig. 3. The details of HiDNet are shown in TABLE I. The input images are sent into a stem, which consists of two 3×3 convolutions of stride 2. Then we obtain a feature of the resolution $\frac{H}{4} \times \frac{W}{4}$. The first stage contains 1 LOD embedding module that has 1 branch and 3 convolution units. The second stage has 1 LOD embedding module with 2 branches and 5 convolution units. The last stage has 7 LOD embedding modules with the same structure. Each module has 3 branches and 5 convolution units. At the end, multi-scale depth estimation results are output by a depth predictor.

TABLE I

DETAILS OF HIDNET ARCHITECTURE. **K** DENOTES THE KERNEL SIZE,**S** DENOTES THE STRIDE AND **INPUT** DENOTES THE INPUT OF EACH LAYERWHERE ↑ DENOTES AN UPSAMPLING OPERATION USING BILINEAR INTERPOLATIONAND ↓ DENOTES ONE OR MORE CONVOLUTIONS WITH STRIDE 2.WHEN THE INPUT HAS MORE THAN ONE FEATURES, ALL FEATURES ARE FUSED BY ELEMENTWISE ADDITION.B, R AND S IN PARENTHESES DENOTE BATCH NORMALIZATION, RELU AND SIGMOID, RESPECTIVELY.NUMBERS IN PARENTHESES DENOTE THE REPEATED TIMES OF THE BLOCK. [] INDICATES THE INPUT MAY NOT EXIST

Depth Network

	layer description	k	s	input	output tensor dim
	RGB image				$3 \times H \times W$
stem	conv1 (BR)	3	2	RGB image	$64 \times H/2 \times W/2$
	conv2 (BR)	3	2	conv1	$64 \times H/4 \times W/4$
stage1 ($\times 1$)	Bottleneck			conv2	$256 \times H/4 \times W/4$
stage2 ($\times 1$)	fuse1 (BR)	3	1	Bottleneck	$48 \times H/4 \times W/4$
	fuse2 (BR)	3	2	Bottleneck	$96 \times H/8 \times W/8$
	BasicBlock1			fuse1	$48 \times H/4 \times W/4$
	BasicBlock2			fuse2	$96 \times H/8 \times W/8$
	fuse1			BasicBlock1, ↑BasicBlock2	$48 \times H/4 \times W/4$
stage3 ($\times 7$)	fuse2 (BR)	3	2	fuse1	$96 \times H/8 \times W/8$
	fuse3 (BR)	3	2	fuse1	$192 \times H/16 \times W/16$
	BasicBlock1			fuse1	$48 \times H/4 \times W/4$
	BasicBlock2			fuse2	$96 \times H/8 \times W/8$
	BasicBlock3			fuse3	$192 \times H/16 \times W/16$
	fuse1			BasicBlock1, ↑BasicBlock2, ↑BasicBlock3	$48 \times H/4 \times W/4$
depth predictor	InvDepth3			↑fuse3	$1 \times H/4 \times W/4$
	InvDepth2			↑fuse2, ↑InvDepth3	$1 \times H/2 \times W/2$
	InvDepth1			↑fuse1, ↑InvDepth2	$1 \times H \times W$

Bottleneck

	layer description	k	s	input	output tensor dim
($\times 1$)	x				$64 \times H/4 \times W/4$
	conv1 (BR)	1	1	x	$64 \times H/4 \times W/4$
	conv2 (BR)	3	1	conv1	$64 \times H/4 \times W/4$
	conv3 (B)	1	1	conv2	$256 \times H/4 \times W/4$
	conv4 (B)	1	1	x	$256 \times H/4 \times W/4$
	x (R)			conv3, conv4	$256 \times H/4 \times W/4$
($\times 3$)	conv1 (BR)	1	1	x	$64 \times H/4 \times W/4$
	conv2 (BR)	3	1	conv1	$64 \times H/4 \times W/4$
	conv3 (B)	1	1	conv2	$256 \times H/4 \times W/4$
	x (R)			conv3, x	$256 \times H/4 \times W/4$

BasicBlock

	layer description	k	s	input	output tensor dim
	x				$C \times H' \times W'$
($\times 4$)	conv1 (BR)	3	1	x	$C \times H' \times W'$
	conv2 (B)	3	1	conv1	$C \times H' \times W'$
	x (R)			conv2, x	$C \times H' \times W'$

InvDepth

	layer description	k	s	input	output tensor dim
	x, [InvDepth]				$C \times H' \times W'$
($\times 1$)	conv1 (BR)	1	1	x	$C \times H' \times W'$
	x (S)	1	1	conv1, [InvDepth]	$1 \times H' \times W'$

Different from HR-Net [8] using 4 parallel feature streams, HiDNet maintains only 1 main feature stream without mining too high-level semantic information. In HR-Net, the four streams of different resolutions are fused with each other to update their feature maps. HiDNet updates the high-resolution feature map of the main stream with LOD Embedding modules, which employ two to three network branches to compute the multi-resolution representations and to merge these representations into one enhanced feature map. HiDNet is related to but does has difference from HR-Net and it is a compact model tailored for high-resolution depth estimation.

E. Training Loss

This section is to summarize the loss functions utilized in the four steps of our proposed framework, as shown in Fig. 4. In the first step, we combine a mask photometric loss and a smoothness loss as: $L_{pc} = L_p + \lambda_s L_s$, where λ_s denotes

TABLE II

QUANTITATIVE COMPARISON OF PERFORMANCE ON THE KITTI DATASET. IN THE *Train* COLUMN: M REFERS TO TRAINING BY MONOCULAR IMAGE SEQUENCE SUPERVISION; S REFERS TO TRAINING BY STEREO IMAGES; † REFERS TO USING CITYSCAPES OR SEMANTIC SEGMENTATION INFORMATION ALONG WITH KITTI FRAMES IN TRAINING; ‡ REFERS TO USING MULTIPLE FRAMES AT TEST TIME; ⋆ REFERS TO USING POST-PROCESSING [17];FOR EACH RESOLUTION IN THE M SETTING, THE BEST RESULTS ARE PRESENTED IN **BOLD**, WITH SECOND BEST RESULTS UNDERLINED. FOR METRICS FOLLOWED BY ↓, LOWER IS BETTER, AND FOR METRICS FOLLOWED BY ↑, HIGHER IS BETTER

Method	Train	Resolution	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSE log ↓	$\delta < 1.25$ ↑	$\delta < 1.25^2$ ↑	$\delta < 1.25^3$ ↑
Klodt *et al.* [46]	M	-	0.166	1.490	5.998	-	0.778	0.919	0.966
SGDepth [57]	M	640×192	0.117	0.907	4.844	0.196	0.875	0.958	0.980
Monodepth2 [1]	M	640×192	0.115	0.903	4.863	0.193	0.877	0.959	0.981
PackNet-SfM [2]	M	640×192	0.111	0.785	4.601	0.189	0.878	0.960	0.982
HR-Depth [9]	M	640×192	0.109	0.792	4.632	0.185	0.884	0.962	0.983
Wang *et al.* [58]	M	640×192	0.109	0.779	4.641	0.186	0.883	0.962	0.982
Johnston *et al.* [20]	M	640×192	0.106	0.861	4.699	0.185	0.889	0.962	0.982
SD-SSMDE [26]	M	640×192	0.106	0.751	4.485	0.180	0.885	0.964	**0.984**
HiDNet (Ours)	M	640×192	0.108	0.733	4.454	0.184	0.887	0.962	0.983
SENSE (Ours)	M	640×192	**0.104**	**0.693**	**4.294**	**0.177**	**0.894**	**0.965**	**0.984**
Wang *et al.* [59]	M‡	640×192	0.106	0.799	4.662	0.187	0.889	0.961	0.982
Monodepth2 [1]	MS	640×192	0.106	0.818	4.750	0.196	0.874	0.957	0.979
FSRE-Depth [60]	M†	640×192	0.102	0.675	4.393	0.178	0.893	0.966	0.984
ManyDepth [61]	M‡	640×192	0.098	0.770	4.459	0.176	0.900	0.965	0.983
EPCDepth [29]	S⋆	640×192	0.099	0.754	4.490	0.183	0.888	0.963	0.982
DynamicDepth [23]	M†‡	640×192	0.096	0.720	4.458	0.175	0.897	0.964	0.984
Monodepth2 [1]	M	1024×320	0.115	0.882	4.701	0.190	0.879	0.961	0.982
SGDepth [57]	M	1280×384	0.113	0.880	4.695	0.192	0.884	0.961	0.981
R-MSFM6 [62]	M	1024×320	0.108	0.748	4.470	0.185	0.889	0.963	0.982
PackNet-SfM [2]	M	1280×384	0.107	0.802	4.538	0.186	0.889	0.962	0.981
FeatDepth [19]	M	1024×320	0.104	0.729	4.481	0.179	0.893	0.965	0.984
HR-Depth [9]	M	1280×384	0.104	0.727	4.410	0.179	0.894	0.966	0.984
SD-SSMDE [26]	M	1024×320	0.101	0.700	4.332	0.174	0.895	0.966	**0.985**
HiDNet (Ours)	M	1024×320	0.104	0.676	4.250	0.177	0.896	0.967	0.984
SENSE (Ours)	M	1024×320	**0.099**	**0.617**	**4.079**	**0.172**	**0.902**	**0.968**	**0.985**
Monodepth2 [1]	MS	1024×320	0.106	0.806	4.630	0.193	0.876	0.958	0.980
Wang *et al.* [59]	M‡	1024×320	0.106	0.773	4.491	0.185	0.890	0.962	0.982
FeatDepth [19]	MS	1024×320	0.099	0.697	4.427	0.184	0.889	0.963	0.982
FSRE-Depth [60]	M†	1024×320	0.102	0.687	4.366	0.178	0.895	0.967	0.984
ManyDepth [61]	M‡	1024×320	0.091	0.694	4.245	0.171	0.911	0.968	0.983
EPCDepth [29]	S⋆	1024×320	0.093	0.671	4.297	0.178	0.899	0.965	0.983
PlaneDepth [22]	S	1280×384	0.085	0.563	4.023	0.171	0.910	0.968	0.984

the weight of the smoothness loss, to train the depth network branch D' and the pose network branch P'. In the second step, we train a depth network D'' from scratch with the enhanced pseudo labels $\{d^p\}$ using the pseudo-label loss L_{pse}. In the third step, we train a pose network P'' with the loss L_{pc} by fixing the depth network D''. Finally, we finetune the depth and the pose network branches together using the loss: $L_{final} = P \odot L_p + \lambda_s L_s + \lambda_{pse} L_{pse}$.

IV. EXPERIMENTS

Dataset: We evaluate the effectiveness of our method on the KITTI [63] dataset. We use the data split in Eigen et al. [30] with the removal of static frames in Zhou et al. [3] for a fair comparison. We use 39,810/ 4,424/ 697 images for training/ validation/ evaluation. The ground truth depth maps for evaluation are captured by a calibrated LiDAR sensor.

Training Details: Our models are built on PyTorch [64] with 2 NVIDIA GeForce RTX 3090 GPUs. We use the Adam optimizer [65] with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and a batch size of 16. The depth network is the proposed HiDNet (see Sec. III-D). The pose network adopts a backbone of ResNet18 [66] with the fixed input resolution of 640 × 192. Both networks are initialized with ImageNet-pretrained [67] weights. The SSIM weight α, smoothness loss weight λ_s and pseudo-label loss weight λ_{pse} are set to 0.85, 0.001, 0.1, respectively.

The proposed self-evolving paradigm contains 4 steps. In the first step, we train the depth and the pose networks for 10 epochs with L_{pc} and a learning rate (LR) of 2×10^{-4}, and then decrease the LR to 2×10^{-5} when finetuning the networks with L_{pc} for another 10 epochs. The depth network in the second step is trained from scratch for the first 10 epochs with an LR of 2×10^{-4}, and then 10 epochs with an LR of 2×10^{-5}. In the third step, while the depth network is fixed, the pose network is trained from scratch for the first 5 epochs and then 5 epochs, with LR 2×10^{-4} and 2×10^{-5}, respectively. Finally, in the last step, the depth and the pose networks are jointly finetuned for 10 epochs with an LR of 2×10^{-5}. For an input/output resolution of 640 × 192, it takes 8 hours for

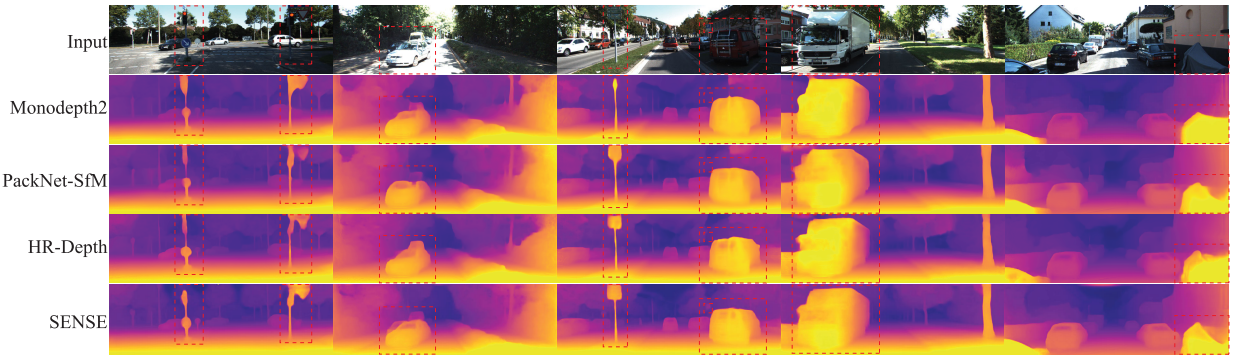


Fig. 5. Qualitative comparison with other methods on the KITTI dataset. Comparing with other self-supervised methods, our approach predicts depth maps with finer details and sharper boundaries of objects. It also successfully estimates the depth of distant objects which other methods fail to recognize.

TABLE III

QUANTITATIVE COMPARISON OF PERFORMANCE ON KITTI IMPROVED GROUND TRUTH FROM [68]

Method	Train	Resolution	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSE log ↓	$\delta < 1.25$ ↑	$\delta < 1.25^2$ ↑	$\delta < 1.25^3$ ↑
Monodepth2 [1]	M	640×192	0.090	0.545	3.942	0.137	0.914	0.983	0.995
Johnston *et al.* [20]	M	640×192	0.081	0.484	3.716	0.126	0.927	0.985	0.996
PackNet-SfM [2]	M	640×192	0.078	0.420	3.485	0.121	0.931	0.986	0.996
HiDNet (Ours)	M	640×192	0.075	0.367	3.323	0.115	0.938	0.989	0.997
SENSE (Ours)	M	640×192	**0.071**	**0.339**	**3.175**	**0.109**	**0.945**	**0.990**	**0.998**
PackNet-SfM [2]	M†	1280×384	0.071	0.359	3.153	0.109	0.944	0.990	0.997
HiDNet (Ours)	M	1024×320	0.070	0.313	3.071	0.107	0.947	0.991	**0.998**
SENSE (Ours)	M	1024×320	**0.067**	**0.281**	**2.923**	**0.102**	**0.951**	**0.992**	**0.998**

training in the first step, 5 hours in the second step, 2 hours in the third step, and 4 hours in the last step.

A. Depth Estimation Performance

For evaluation, we follow the commonly used metrics described in [30], and cap the maximum predictions of all network to 80 meters [17]. We conduct experiments with two input/output resolutions, i.e. 640 × 192 and 1024 × 320, to better compare with other methods. Methods that use the resolution of 1280 × 384 are also included and compared with our results of larger resolution for a more comprehensive comparison. We use only a monocular video for training. Such limited supervision requires a scale factor $\hat{s}$ to match the median of the predicted depth maps with ground truth [3], i.e. $\hat{s} = median(D_{gt})/median(D_{pred})$.

The results are summarized in TABLE II. We first train our model using the proposed backbone of HiDNet (cf. Sec. III-D) without self-evolving learning or pseudo label refinement, which already produces a strong baseline that is competitive or even better than the current state-of-the-art methods. Moreover, our full approach, SENSE, achieves better performance and outperforms all existing self-supervised models. Our method shows improvements in all metrics, especially for Sq Rel and RMSE. According to their definitions, Sq Rel is more sensitive to errors in the short range, while RMSE penalizes large depth errors which occur in distant regions more often. Therefore, our approach not only overcomes the challenges from textureless regions or near objects that produce sparse supervisory signals, but also learns high-resolution representations that capture precise information in long-range regions. SENSE also outperforms or produces comparable results to methods that rely on a stronger supervision in training, including stereo image pairs and semantic segmentation label, or multiple frames at test time.

We study the performance of our model using an improved set of ground truth depth maps for the KITTI dataset from [68], which produces 652 high-quality depth maps by using 5 consecutive LiDAR frames and stereo pairs to handle moving objects and occlusions. We compare the results with other methods in TABLE III where our proposed SENSE surpasses previous methods in all measures. Fig. 5 shows the performance comparison qualitatively. In general, SENSE is able to capture finer details and structures of objects including trucks, trees, and signs in contrast to existing methods. Note that in the 4th column, while other models struggle to recognize the traffic sign, SENSE successfully estimates its depth.

B. Ablation Studies

In TABLE IV, we perform an ablative analysis to validate the effectiveness of each proposed algorithmic component.

1) HiDNet: First, we show the effectiveness of our proposed LOD embedding module by comparing the performance of HiDNet with a baseline. In the baseline, the LOD modules are replaced with single-level embedding modules, i.e., $m = 1$ for all modules and n remains the same. In the first and second rows, our proposed HiDNet outperforms the baseline in all metrics, indicating the hierarchical architectures can perform more accurately by inferring depth from different levels of details. Second, we show that our proposed HiDNet performs better than using HRNet as depth network in most metrics, especially in the metrics Sq Rel and RMSE. The results indicate that our design of one high resolution branch is better than HRNet to resist the noisy influence from the

TABLE IV

ABLATION STUDY ON THE KITTI BENCHMARK FOR 1024 × 320 RESOLUTION. PP MEANS THE RESULTS HAVE BEEN POST PROCESSED USING [17]

	Method	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSE log ↓	$\delta < 1.25$ ↑	$\delta < 1.25^2$ ↑	$\delta < 1.25^3$ ↑
1	HRNet [8]	0.104	0.687	4.317	0.177	0.894	0.966	0.984
2	baseline	0.115	0.735	4.444	0.187	0.877	0.961	0.983
3	HiDNet	0.104	0.676	4.250	0.177	0.896	0.967	0.984
4	HiDNet w/ pp	0.103	0.652	4.191	0.176	0.897	0.967	**0.985**
5	SENSE Step 2 w/o GMM	0.102	0.644	4.145	0.174	0.898	0.967	**0.985**
6	SENSE Step 2 w/ GMM	0.100	0.620	4.115	0.173	0.900	**0.968**	**0.985**
7	SENSE	**0.099**	0.617	4.079	0.172	**0.902**	**0.968**	**0.985**
8	SENSE w/ another round	**0.099**	**0.612**	**4.069**	**0.171**	**0.902**	**0.968**	**0.985**

TABLE V

THE COMPLEXITY OF HIDNET. THE FLOPS ARE CALCULATED BASED ON ASINGLE IMAGE WITH A RESOLUTION OF 640 × 192

Model	RMSE	Params	Ratio-to-HiDNet	FLOPs	Ratio-to-HiDNet
HiDNet	4.250	31.64M	1×	36.26G	1×
PackNet-SfM	4.538	128.29M	4.1×	205.37G	5.7×
FeatDepth	4.481	35.21M	1.1×	31.95G	0.9×

TABLE VI

THE INFLUENCE OF λ_{pse}. INPUT RESOLUTION IS 1024 × 320

λ_{pse}	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSE log ↓	$\delta < 1.25$ ↑	$\delta < 1.25^2$ ↑	$\delta < 1.25^3$ ↑
0	**0.098**	0.638	4.112	0.172	**0.903**	**0.968**	**0.985**
1	0.100	0.622	4.091	0.172	0.901	**0.968**	**0.985**
0.1	0.099	**0.617**	**4.079**	0.172	0.902	**0.968**	**0.985**
0.01	**0.098**	0.631	4.091	**0.171**	**0.903**	**0.968**	**0.985**
0.001	**0.098**	0.639	4.110	0.172	**0.903**	**0.968**	**0.985**

photometric loss. What's more, we analyse the complexity of HiDNet, which is shown in TABLE V. Compared with PackNet-SfM [2], HiDNet has less parameters and GFLOPs, which is more applicable when the computation resources are limited. Besides, HiDNet has comparable parameters and GFLOPs with FeatDepth [19], but shows superior performance.

2) Pseudo Labels as Supervision: As described in Sec. III-B, the generated pseudo labels from the self-supervised network are post processed using [17] to provide better supervision. The row of "SENSE Step 2 w/o GMM" refers to the depth network trained by the post-processed pseudo-labels but without further steps. We show that its performance surpasses its "ground truth" (4th row), which validates our insight that CNNs properly trained for monocular depth estimation are able to correct the noise in training samples.

3) Refinement With GMM: The performance with and without GMM pseudo label refinement (cf. Sec. III-C) is compared in the 5th and 6th row. A great performance boost is obtained by using GMM, indicating that more confident predictions provide better supervision and our method based on GMM is an effective way to find them.

4) Self-Evolving Learning: Finally, we carry out the whole cycle of self-evolving learning, resulting in the full model of SENSE. By jointly finetuning the depth and the pose networks with self-supervised and supervised training, we further improve the depth prediction (7th row *v.s.* 6th row). To fully explore the potential of self-evolving learning, we conduct experiments that train the model for another round of total 4 steps. We observe that the performance can be slightly improved, which further validates the effectiveness of the self-evolving training.

5) Generalization of Our Approach: To demonstrate the generalization of our proposed self-evolving learning framework, which can be applied to any self-supervised monocular depth estimation model. In TABLE IX, we perform an ablation study on previous methods, i.e. Monodepth2 [1] with the backbone of ResNet-18 and ResNet-50 [66], and HR-Depth [9]. We produce pseudo labels using the original models whose performance is shown in the first, third, and fifth rows. Then we apply our proposed SENSE framework to these models using the pseudo labels, and report their results. The results demonstrate that our method is general enough to boost existing self-supervised networks.

6) Hyperparameters: We investigate the influence of weight factor λ_{pse}. When $\lambda_{pse} = 0$, the pseudo labels are not used to guide the training. As shown in TABLE VI, pseudo labels can lead to significant improvements for the metrics Sq Rel and RMSE while influencing the metrics Abs Rel and $\delta < 1.25$ slightly. $\lambda_{pse} = 0.1$ achieves a better trade off in most of the metrics. Additionally, we study the effect of different combinations of m in LOD embedding module. We design 4 different depth network architectures. The baseline model contains $m = 1$ branch in all 3 stages. HiDNet_1_2_2 contains $m = 1$ branch in Stage 1, $m = 2$ branches in Stage 2 and $m = 2$ branches in Stage 3. HiDNet_1_2_4 contains $m = 1$ branch in Stage 1, $m = 2$ branches in Stage 2 and $m = 4$ branches in Stage 3. In TABLE VII, the results show that out proposed HiDNet with 3-branch LOD embedding

TABLE VII
THE INFLUENCE OF DIFFERENT BRANCH NUMBER m IN A LOD EMBEDDING MODULE

Model	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSE log ↓	$\delta < 1.25$ ↑	$\delta < 1.25^2$ ↑	$\delta < 1.25^3$ ↑
baseline	0.115	0.735	4.444	0.187	0.877	0.961	0.983
HiDNet_1_2_2	0.108	0.690	4.308	0.181	0.889	0.964	**0.984**
HiDNet	**0.104**	**0.676**	**4.250**	**0.177**	**0.896**	**0.967**	**0.984**
HiDNet_1_2_4	0.108	0.753	4.407	0.182	0.891	0.964	0.983

TABLE VIII
THE INFLUENCE OF VARYING QUALITY OF PSEUDO LABELS

Epoch	Step	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSE log ↓	$\delta < 1.25$ ↑	$\delta < 1.25^2$ ↑	$\delta < 1.25^3$ ↑
5	1	0.135	0.822	4.636	0.200	0.842	0.956	0.983
	2	0.125	0.757	4.444	0.190	0.861	0.960	0.984
10	1	0.114	0.785	4.425	0.187	0.879	0.963	0.983
	2	0.109	0.715	4.249	0.180	0.887	0.966	0.984
20	1	0.104	0.676	4.250	0.177	0.896	0.967	0.984
	2	0.100	0.620	4.115	0.173	0.900	0.968	0.985

TABLE IX
APPLYING OUR SELF-EVOLVING LEARNING METHOD TO MONODEPTH2 [1] WITH THE BACKBONE OF RESNET-18 AND RESNET-50 (REFERRED BY "MD2 R18" AND "MD2 R50", RESPECTIVELY)AND HR-DEPTH [9]. THE "+" REFERS TO THE MODEL COMBINED WITH THE SENSE FRAMEWORK

Method	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$
MD2 R18	0.115	0.903	4.863	0.193	0.877
MD2 R18+	**0.114**	**0.886**	**4.701**	**0.188**	**0.880**
MD2 R50	0.110	0.831	4.642	0.187	0.883
MD2 R50+	**0.108**	**0.803**	**4.510**	**0.182**	**0.890**
HR-Depth	0.109	0.792	4.632	0.185	**0.884**
HR-Depth+	**0.108**	**0.758**	**4.567**	**0.183**	**0.884**

module in Stage 3 outperforms the other two settings in our experiment.

7) Varying Quality of Pseudo Labels: We investigate the relationship between the quality level of pseudo labels and the performance of the model and the results are shown in TABLE VIII. The pseudo labels are obtained using the depth network trained in Step 1 in different epochs. Then the pseudo labels are used to train a new depth network from scratch in Step 2. TABLE VIII shows that the performance of depth estimation is influenced by the quality of pseudo labels. With higher-quality pseudo labels, depth estimation can have better performance. And our proposed method is shown to improve the performance of depth estimation efficiently even when the quality of pseudo labels is low.

V. DISCUSSION AND CONCLUSION

We have proposed SENSE, a novel learning paradigm for self-supervised depth estimation with monocular videos. The SENSE framework bridges the gap between self and full supervised learning, and progressively enhances the depth and pose network along with the pseudo labels in an self-evolving manner. Moreover, we develop HiDNet, a depth estimation backbone, when deployed alone without the proposed self-evolving learning, can already produce comparable or even better results than the state-of-the-art. We believe HiDNet may become the new baseline for monocular depth estimation and benefit the community.

REFERENCES

[1] C. Godard, O. M. Aodha, M. Firman, and G. Brostow, "Digging into self-supervised monocular depth estimation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 3828–3838.

[2] V. Guizilini, R. Ambrus, S. Pillai, A. Raventos, and A. Gaidon, "3D packing for self-supervised monocular depth estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2485–2494.

[3] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1851–1858.

[4] D. Arpit et al., "A closer look at memorization in deep networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 233–242.

[5] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning requires rethinking generalization," 2016, *arXiv:1611.03530*.

[6] G. Li, Y. Xie, and L. Lin, "Weakly supervised salient object detection using image labels," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, no. 1, 2018, pp. 7024–7031.

[7] G. Pang, C. Yan, C. Shen, A. van den Hengel, and X. Bai, "Self-trained deep ordinal regression for end-to-end video anomaly detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 12173–12182.

[8] J. Wang et al., "Deep high-resolution representation learning for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3349–3364, Oct. 2021.

[9] X. Lyu et al., "HR-Depth: High resolution self-supervised monocular depth estimation," 2020, *arXiv:2012.07356*.

[10] R. Ranftl, A. Bochkovskiy, and V. Koltun, "Vision transformers for dense prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 12179–12188.

[11] D. Luebke, M. Reddy, J. D. Cohen, A. Varshney, B. Watson, and R. Huebner, *Level of Detail for 3D Graphics*. Burlington, MA, USA: Morgan Kaufmann, 2003.

[12] V. Casser, S. Pirk, R. Mahjourian, and A. Angelova, "Depth prediction without the sensors: Leveraging structure for unsupervised learning from monocular videos," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, no. 1, 2019, pp. 8001–8008.

[13] A. Ranjan et al., "Competitive collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 12240–12249.

[14] C. Wang, J. M. Buenaposada, R. Zhu, and S. Lucey, "Learning depth from monocular videos using direct methods," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2022–2030.

[15] Z. Yang, P. Wang, Y. Wang, W. Xu, and R. Nevatia, "LEGO: Learning edge with geometry all at once by watching videos," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 225–234.

[16] R. Mahjourian, M. Wicke, and A. Angelova, "Unsupervised learning of depth and ego-motion from monocular video using 3D geometric constraints," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5667–5675.

[17] C. Godard, O. M. Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 270–279.

[18] R. Garg, V. K. Bg, G. Carneiro, and I. Reid, "Unsupervised CNN for single view depth estimation: Geometry to the rescue," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2016, pp. 740–756.

[19] C. Shu, K. Yu, Z. Duan, and K. Yang, "Feature-metric loss for self-supervised learning of depth and egomotion," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2020, pp. 572–588.

[20] A. Johnston and G. Carneiro, "Self-supervised monocular trained depth estimation using self-attention and discrete disparity volume," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 4756–4765.

[21] D. Han, J. Shin, N. Kim, S. Hwang, and Y. Choi, "TransDSSL: Transformer based depth estimation via self-supervised learning," *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 10969–10976, Oct. 2022.

[22] R. Wang, Z. Yu, and S. Gao, "PlaneDepth: Self-supervised depth estimation via orthogonal planes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 21425–21434.

[23] Z. Feng, L. Yang, L. Jing, H. Wang, Y. Tian, and B. Li, "Disentangling object motion and occlusion for unsupervised multi-frame monocular depth," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2022, pp. 228–244.

[24] M. Poggi, F. Aleotti, F. Tosi, and S. Mattoccia, "On the uncertainty of self-supervised monocular depth estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 3227–3237.

[25] W. Ren, L. Wang, Y. Piao, M. Zhang, H. Lu, and T. Liu, "Adaptive co-teaching for unsupervised monocular depth estimation," in *Proc. 17th Eur. Conf. Comput. Vis.*, Tel Aviv, Israel. Cham, Switzerland: Springer, 2022, pp. 89–105.

[26] A. Petrovai and S. Nedevschi, "Exploiting pseudo labels in a self-supervised learning framework for improved monocular depth estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 1578–1588.

[27] X. Tang and L. Chen, "An unsupervised monocular image depth prediction algorithm based on multiple loss deep learning," *IEEE Access*, vol. 7, pp. 162405–162414, 2019.

[28] A. Pilzer, S. Lathuilière, N. Sebe, and E. Ricci, "Refine and distill: Exploiting cycle-inconsistency and knowledge distillation for unsupervised monocular depth estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9768–9777.

[29] R. Peng, R. Wang, Y. Lai, L. Tang, and Y. Cai, "Excavating the potential capacity of self-supervised monocular depth estimation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 15560–15569.

[30] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 1–9.

[31] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao, "Deep ordinal regression network for monocular depth estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2002–2011.

[32] Y. Luo et al., "Single view stereo matching," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 155–163.

[33] J. Jiao, Y. Cao, Y. Song, and R. Lau, "Look deeper into depth: Monocular depth estimation with semantic booster and attention-driven loss," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 53–69.

[34] F. Saeedan and S. Roth, "Boosting monocular depth with panoptic segmentation maps," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 3853–3862.

[35] L. Wang, J. Zhang, O. Wang, Z. Lin, and H. Lu, "SDC-Depth: Semantic divide-and-conquer network for monocular depth estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 541–550.

[36] Z. Zhang, Z. Cui, C. Xu, Z. Jie, X. Li, and J. Yang, "Joint task-recursive learning for semantic segmentation and depth estimation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 235–251.

[37] X. Guo, H. Li, S. Yi, J. Ren, and X. Wang, "Learning monocular depth by distilling cross-domain stereo networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 484–500.

[38] J. N. Kundu, P. K. Uppala, A. Pahuja, and R. V. Babu, "AdaDepth: Unsupervised content congruent adaptation for depth estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2656–2665.

[39] S. Zhao, H. Fu, M. Gong, and D. Tao, "Geometry-aware symmetric domain adaptation for monocular depth estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9788–9798.

[40] W. Yin, Y. Liu, C. Shen, and Y. Yan, "Enforcing geometric constraints of virtual normal for depth prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 5684–5693.

[41] X. Cheng, P. Wang, and R. Yang, "Depth estimation via affinity learned with convolutional spatial propagation network," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 103–119.

[42] D. Wofk, F. Ma, T.-J. Yang, S. Karaman, and V. Sze, "FastDepth: Fast monocular depth estimation on embedded systems," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 6101–6108.

[43] Q. Xie, M.-T. Luong, E. Hovy, and Q. V. Le, "Self-training with noisy student improves ImageNet classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10687–10698.

[44] D. Tanaka, D. Ikami, T. Yamasaki, and K. Aizawa, "Joint optimization framework for learning with noisy labels," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5552–5560.

[45] K. Yi and J. Wu, "Probabilistic end-to-end noise correction for learning with noisy labels," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7017–7025.

[46] M. Klodt and A. Vedaldi, "Supervising the new with the old: Learning SFM from SFM," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 698–713.

[47] F. Tosi, F. Aleotti, M. Poggi, and S. Mattoccia, "Learning monocular depth estimation infusing traditional stereo knowledge," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9799–9809.

[48] Z. Ren et al., "STFlow: Self-taught optical flow estimation using pseudo labels," *IEEE Trans. Image Process.*, vol. 29, pp. 9113–9124, 2020.

[49] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[50] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper depth prediction with fully convolutional residual networks," in *Proc. 4th Int. Conf. 3D Vis. (3DV)*, Oct. 2016, pp. 239–248.

[51] Y. Kuznietsov, J. Stückler, and B. Leibe, "Semi-supervised deep learning for monocular depth map prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6647–6655.

[52] L. Zwald and S. Lambert-Lacroix, "The BerHu penalty and the grouped effect," 2012, *arXiv:1207.6868*.

[53] J. Li, R. Socher, and S. C. Hoi, "DivideMix: Learning with noisy labels as semi-supervised learning," in *Proc. Int. Conf. Learn. Represent.*, 2020, pp. 1–14.

[54] R. Li, X. He, Y. Zhu, X. Li, J. Sun, and Y. Zhang, "Enhancing self-supervised monocular depth estimation via incorporating robust constraints," in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 3108–3117.

[55] J. Watson, M. Firman, G. Brostow, and D. Turmukhambetov, "Self-supervised monocular depth hints," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 2162–2171.

[56] L. Ladický, J. Shi, and M. Pollefeys, "Pulling things out of perspective," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 89–96.

[57] M. Klingner, J.-A. Termöhlen, J. Mikolajczyk, and T. Fingscheidt, "Self-supervised monocular depth estimation: Solving the dynamic object problem by semantic guidance," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2020, pp. 582–600.

[58] L. Wang, Y. Wang, L. Wang, Y. Zhan, Y. Wang, and H. Lu, "Can scale-consistent monocular depth be learned in a self-supervised scale-invariant manner?" in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 12727–12736.

[59] J. Wang, G. Zhang, Z. Wu, X. Li, and L. Liu, "Self-supervised joint learning framework of depth estimation via implicit cues," 2020, *arXiv:2006.09876*.

[60] H. Jung, E. Park, and S. Yoo, "Fine-grained semantics-aware representation enhancement for self-supervised monocular depth estimation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 12642–12652.

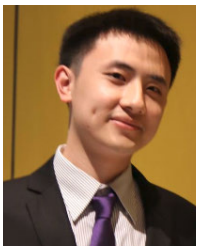
[61] J. Watson, O. M. Aodha, V. Prisacariu, G. Brostow, and M. Firman, "The temporal opportunist: Self-supervised multi-frame monocular depth," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 1164–1174.
[62] Z. Zhou, X. Fan, P. Shi, and Y. Xin, "R-MSFM: Recurrent multi-scale feature modulation for monocular depth estimating," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 12777–12786.
[63] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1231–1237, Sep. 2013.
[64] A. Paszke et al., "Automatic differentiation in PyTorch," in *Proc. 31st Conf. Neural Inf. Process. Syst.*, 2017, pp. 1–4.
[65] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[66] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[67] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
[68] J. Uhrig, N. Schneider, L. Schneider, U. Franke, T. Brox, and A. Geiger, "Sparsity invariant CNNs," in *Proc. Int. Conf. 3D Vis. (3DV)*, Oct. 2017, pp. 11–20.

Guanbin Li (Member, IEEE) received the Ph.D. degree from The University of Hong Kong in 2016. He is currently an Associate Professor with the School of Data and Computer Science, Sun Yat-sen University. He has authored and coauthored more than 100 papers in top-tier academic journals and conferences. His current research interests include computer vision, image processing, and deep learning. He was a recipient of the ICCV 2019 Best Paper Nomination Award. He serves as an Area Chair for the conference of VISAPP. He has been serving as a reviewer for numerous academic journals and conferences, such as IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, *International Journal of Computer Vision*, IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON MULTIMEDIA, IEEE TRANSACTIONS ON CYBERNETICS, CVPR, ICCV, ECCV, and NeurIPS.

Ricong Huang received the B.Eng. degree from Sun Yat-sen University in 2021, where he is currently pursuing the master's degree with the School of Data and Computer Science. His research interests include computer vision, image processing, and deep learning.

Haofeng Li (Member, IEEE) received the B.Sc. degree from Sun Yat-sen University in June 2015 and the Ph.D. degree from the Department of Computer Science, The University of Hong Kong, in March 2020. He is currently a Research Scientist with the Shenzhen Research Institute of Big Data, The University of Hong Kong (Shenzhen). He has published more than ten papers in the top conference/journals of artificial intelligence and computer vision, including ICCV, AAAI, ACM-MM, ISBI, IEEE TRANSACTIONS ON IMAGE PROCESSING, MedIA, *ACM Transactions on Multimedia Computing, Communications, and Applications*, and IEEE TRANSACTIONS ON CYBERNETICS. His research interests include computer vision, medical image analysis, brain MRI analysis, and adversarial machine learning. He is a reviewer of IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON IMAGE PROCESSING, *Pattern Recognition*, and *Neurocomputing*.

Zunzhi You received the B.Eng. degree from Sun Yat-sen University in 2021. His research interests include computer vision and machine learning.



Weikai Chen received the Ph.D. degree from The University of Hong Kong in 2017. He is currently a Lead Research Scientist with Tencent America. Previously, he was a Postdoctoral Researcher and then a Research Associate with the Vision and Graphics Laboratory (VGL), USC ICT. He has published more than 30 papers in top-tier academic journals and conferences, including IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, SIGGRAPH/SIGGRAPH Asia, CVPR, ICCV, ECCV, and NeurIPS. His research interests include interplay among computer graphics, computer vision, and deep learning. His work on differentiable rendering, SoftRas (ICCV'19), is adopted by Pytorch3D as one of the core algorithms. He was a recipient of the CVPR 2019 Best Paper Finalist Award.