Data-free Dense Depth Distillation

Junjie Hu, *Member, IEEE,* Chenyou Fan, Mete Ozay, Hualie Jiang, and Tin Lun Lam†, *Senior Member, IEEE*

***Abstract*—We study data-free knowledge distillation (KD) for monocular depth estimation (MDE), which learns a lightweight network for real-world depth perception by compressing from a trained expert model under the teacher-student framework while lacking training data in the target domain. Owing to the essential difference between dense regression and image recognition, previous methods of data-free KD are not applicable to MDE. To strengthen the applicability in the real world, in this paper, we seek to apply KD with out-of-distribution simulated images. The major challenges are i) lacking prior information about object distribution of the original training data; ii) the domain shift between the real world and the simulation. To cope with the first difficulty, we apply object-wise image mixing to generate new training samples for maximally covering distributed patterns of objects in the target domain. To tackle the second difficulty, we propose to utilize a transformation network that efficiently learns to fit the simulated data to the feature distribution of the teacher model. We evaluate the proposed approach for various depth estimation models and two different datasets. As a result, our method outperforms the baseline KD by a good margin and even achieves slightly better performance with as few as $1/6$ images, demonstrating a clear superiority.**

***Index Terms*—Monocular depth estimation, knowledge distillation, data-free KD, dense distillation**

I. Introduction

As a cost-effective alternative solution to depth sensors, monocular depth estimation (MDE) predicts scene depth from only RGB images and has wide applications in various tasks, such as scene understanding [26], autonomous driving [47], 3D reconstruction [14], and augmented reality [9]. In recent years, accuracy of MDE methods has been significantly boosted and dominated by deep learning based approaches [13], [20], [27], where the advances are attributed to modeling and estimating depth by complex nonlinear functions using large-scale deep convolutional neural networks (CNNs).

On the other hand, many practical applications, *e.g.*, robot navigation, demand a lightweight model due to the hardware limitations and requirement for computationally efficient inference. In these cases, we can either perform model compression on a well-trained large network [53] or apply supervised learning to directly train a compact network [36]. These solutions assume that the original training data of the target domain is known and can be freely accessed. However, since data privacy and security are invariably a severe concern in the real world, the training data is routinely unknown in practice, especially for industrial applications. A potential solution under this practical constraint is to distill preserved knowledge from a well-trained and publicly available model. The task is called data-free knowledge distillation (KD) [33] and has been shown effective for image recognition.

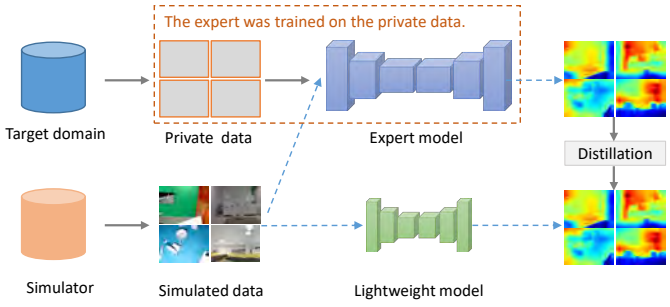


Fig. 1. A visualization of the proposed framework for model compression in monocular depth estimation tasks. We propose to use simulated images as an alternative solution to the challenges of applying knowledge distillation for monocular depth estimation when original training data is not available.

Most existing methods of data-free KD proposed to synthesize training images from random noise [55], [12]. Specifically, assuming that y is a target object attribute, it is an element that inherently exists in the last layer of a classifier and is easily pre-specified, such that we can enforce a classifier to produce the desired output by gradually optimizing its input data. We refer to this property as the inherent constraint of classification. Unfortunately, due to the essential difference between outputs of models obtained in depth estimation and object classification tasks, the inherent constraint does not hold for MDE, making most existing data-free approaches incompatible.

Given the above challenges, in this paper, we propose to leverage out-of-distribution (OOD) images as an alternative for applying KD. For MDE task, intuitively, we consider three critical elements for choosing the alternative set: i) scene similarity, ii) the number of images, and iii) domain gap. The effective yet not practical solution is to collect a dataset similar to the original training data. In reality, data collection is always costly and time-consuming. Besides, due to the lack of prior information about scene structures of the original training data, we have no sufficient clues to guide this data collection process. For these reasons, we prefer using synthetic images collected from simulators. In this way, we can handily obtain enough data to satisfy the requirement ii), and may accordingly sample useful scenarios to ensure i) to a certain extent. However, it will trigger iii) and bring us the significant

J. Hu is with the Shenzhen Institute of Artificial Intelligence and Robotics for Society, the Chinese University of Hong Kong, Shenzhen. E-mail: hujunjie@cuhk.edu.cn

C. Fan is with the School of Artificial Intelligence, South China Normal University, China. E-mail: fanchenyou@scnu.edu.cn.

M.Ozay is with the Samsung Research, UK. E-mail: meteozay@gmail.com.

H. Jiang and T. T. Lam are with the School of Science and Engineering, the Chinese University of Hong Kong, Shenzhen. E-mail: hualiejiang@link.cuhk.edu.cn, tllam@cuhk.edu.cn.

†Corresponding author: Tin Lun Lam

domain gap between simulated and the real world data.

We analyze the effect of these three factors on the accuracy of KD through empirical experiments. Unsurprisingly, high scene similarity, sufficient data, and a small domain gap contribute to better accuracy. Another valuable observation is that the teacher still estimates meaningful depth maps that correctly represent relative depths among objects even from simulated images. It reveals that CNNs may utilize some geometric cues [21], [19], [50], or can learn some domain-invariant features [5] for inferring depths, rather than the straightforward fitting. Therefore, it is still possible to perform KD even though the predicted depth maps are completely wrong in scales. This phenomenon encourages us to develop a data-free dense depth distillation framework with OOD simulated images.

The problem formulated in this paper is visualized in Fig. 1 where we aim to learn a lightweight model on the target domain by distilling from an expert teacher pre-trained with the private data, utilizing a set of simulated images. Following previous methods of data-free KD, we only have prior knowledge of model outputs. For image recognition, this prior information is the detailed target category. For MDE, we are only aware of the model's deployment environments, *e.g.*, indoor or outdoor scenarios. In general, the difficulties are two-fold. The first is the unknown distributed patterns of objects in the target domain. The second is the unavoidable domain discrepancy between the transfer and original training sets.

Our distillation framework is composed of two sub-branches. The first branch applies the plain KD using original simulated images to ensure a lower bound performance. The second branch generates additional training samples to tackle the above challenges with two technical proposals. Specifically, to handle the first challenge, we generate additional training images to cover the distributed patterns of objects in the target domain by applying random object-wise mixing between two simulated images. The object-wise mixing is achieved by utilizing semantic maps provided by most modern simulators. To tackle the second challenge, we propose to regularize the OOD simulated images to fit the target domain. However, as the original data is unavailable, such transformation is intractable. Inspired by DeepInversion [55], we formulate it as an image-to-feature adaption problem by leveraging the running statics in batch normalization layers. To solve the issue of slow optimization, we propose a transformation network that formulates the batch-wise optimization into a learning problem. Fig. 2 shows the diagram of these technical components where we learn the transformation network and the target student network at the same time.

To the best of our knowledge, we are the first to distill knowledge for MDE in data-free scenarios. We extensively evaluate the proposed method for different depth estimation models and multiple datasets, including NYU-v2 and ScanNet. In all datasets, our approach demonstrates the best performance. It outperforms the baseline KD by a good margin and shows slightly better performance with as few as $1/6$ of the image dataset.

In summary, our contributions include:

1) We are the first to study data-free KD for monocular depth estimation. We tackle this unexplored problem with the proposal of using OOD simulated images in a novel data-free dense depth distillation framework.
2) We perform preliminary studies to understand the essential requirements for selecting the OOD data by analyzing how a depth estimator reacts to different types of OOD datasets.
3) We apply object-wise image mixing to generate new training images to cover the objects' distributed patterns in the target domain.
4) We propose to learn a transformation network to efficiently regularize the simulated images to fit the feature distribution of the teacher model.
5) Our method obtains consistent performance improvements for various MDE models and different datasets.

The rest of the paper is organized as follows. In Section. II, we introduce the related backgrounds and techniques, including monocular depth estimation, knowledge distillation, and image mixing. In Section. III, we first give formal analyzes regarding the difficulties of applying data-free KD for MDE and present our method in detail. Section. IV shows detailed experimental settings and results to verify the effectiveness of our method. Section. IV concludes the paper.

II. Related Work

A. Monocular Depth Estimation

Monocular depth estimation (MDE) aims to predict scene depths from only a single image. Deep learning-based approaches have dominated recent progress [25], [34], [13], [23], [27] in which the advanced performances are attributed to modeling and estimating depth using large and complex CNNs with data-driven learning.

On the other hand, deploying MDE algorithms into real-world applications often faces practical challenges, such as limited hardware resources and inefficient computation. Therefore, an emerging requirement of MDE is to develop lightweight models to meet the above demands. This problem has been specifically considered in previous studies [36], [40], [53], [18] where several different lightweight networks have been designed.

However, lightweight networks inevitably degrade their MDE performance due to the trade-off between the model complexity and accuracy. Hence, it remains an open question: how to reduce the model complexity while maintaining high accuracy. One potential solution to this problem is knowledge distillation (KD) which transfers the knowledge from a cumbersome teacher network to a compact student network with descent accuracy improvement. However, KD requires the original training dataset for implementation. Currently, there are no existing solutions in data-free scenarios for MDE.

B. Knowledge Distillation

Knowledge distillation [16] was initially introduced in image recognition where either the soft label or the one-hot label predicted by the teacher is used to supervise the student learning. Existing methods can be generally categorized into two classes considering whether they can access the original

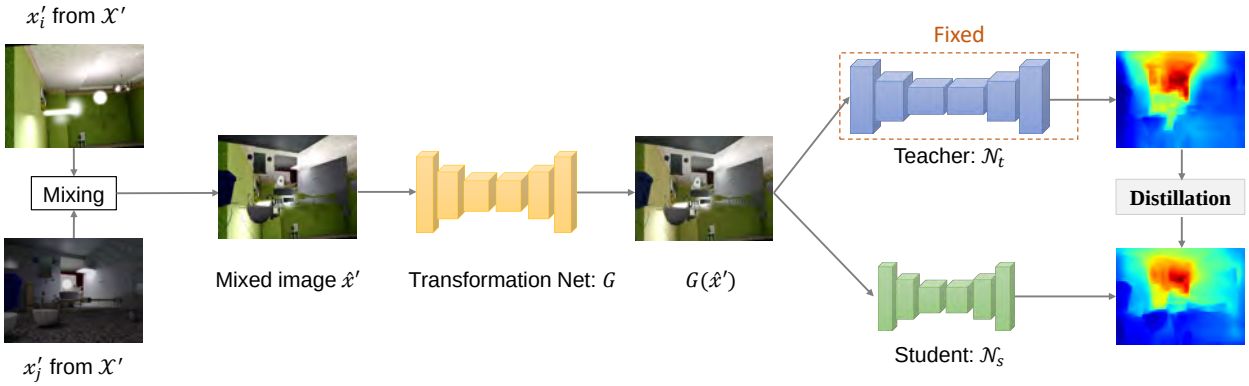


Fig. 2. A flowchart of the proposed approach for distilling a trained model in real world with simulated images in a forward computation. We firstly mix two images x_i' and x_j' sampled from the simulated dataset $\mathcal{X}'$ to generate a new sample $\hat{x}'$, and use a transformation network to fit $\hat{x}'$ to the feature distribution of the trained teacher. Then, the distillation is applied from the teacher $\mathcal{N}_t$ to the target student $\mathcal{N}_s$ with the new input $G(\hat{x}')$.

training set: 1) the standard data-aware KD, and 2) data-free KD.

For data-aware KD, its effectiveness has also been demonstrated on various vision tasks, such as image recognition [16], semantic segmentation [32], object detection [2], and depth estimation [43], [52], *etc*. In addition to the conventional setup, researchers have proposed to improve KD via distilling intermediate features [22], [32], distilling from multiple teachers [48], [31], employing an additional assistant network [38], and adversarial distillation [6], [45].

For data-free KD, researchers resorted to synthesize the training set from random noises [33], [55], [12], [11], [56] or employ other large scale data from different domains [3], [54], [10], [39]. However, existing methods are most effective for classification tasks due to the natural property of deep classification models and cannot be applied to MDE. We will elaborate these observations in Sec. III-A. In this paper, we propose the first method of data-free distillation for MDE. Our method leverages data from simulated environments to distill a model trained on a real-world dataset.

We especially clarify the differences between data-free KD and domain adaptation (DA), *e.g*., sim-to-real adaptation, since they may cause some misunderstandings. Data-free KD essentially differs from DA in two aspects. First, DA transfers a model from the original domain to a different domain, while data-free KD aims at preserving the model accuracy in the original domain. Second, both RGB images in the original and the new domains are usually considered prior information and can be freely accessed in DA. In contrast, training images in the original domain are unknown for data-free KD.

C. Image Mixing for Data Augmentation

Image Mixing is a common technique used for applying data augmentation in semi-supervised learning. One can generate new training pairs in data-scarce scenarios by linearly blending two images and their respective labels (or pseudo-labels). Then, those mixed images and labels are utilized for model training. Classical methods of image mixing include MixUp [59], MixMatch [1] for pixel-wise blending, and CutMix [58], ClassMix [41] for mask-based mixing, *i.e*., exchanging parts of patches or objects between two images. Among them, the former three methods are applicable to classification, and the ClassMix is tailored to image segmentation.

In our method, we apply object-wise mixing to cover the distributed objects in the original training dataset. Unlike the above methods, we only mix RGB images because mixing depth maps will destroy the geometric relations among objects and yield wrong target labels. In a nutshell, instead of generating training pairs of the mixed images and the mixed labels, we use pairs of the mixed images and predictions from them as additional training samples.

III. Method

A. Preliminary

1) Knowledge Distillation: Suppose that $\mathcal{N}_t$ is a model trained using data from the target domain $\mathcal{D} = \{\mathcal{X}, \mathcal{Y}\}$ where $\mathcal{X}$ and $\mathcal{Y}$ denote input data (i.e. image) and label space, respectively. For any $x \in \mathcal{X}$, its corresponding label is estimated by $y = \mathcal{N}_t(x)$.

KD aims at learning a smaller network $\mathcal{N}_s$ with the supervision from $\mathcal{N}_t$. Usually, $\mathcal{N}_t$ is called the teacher network and $\mathcal{N}_s$ is called the student network, respectively. Then, the learning is formulated as:

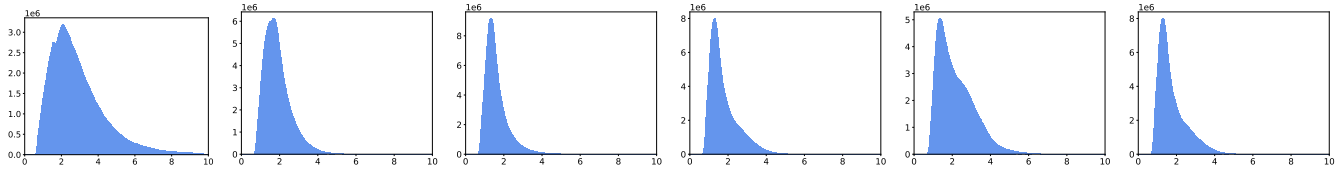
$$\min_{\mathcal{N}_s} \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} \lambda \mathcal{H}\left(\mathcal{N}_t(x), \mathcal{N}_s(x)\right) + (1-\lambda)\mathcal{H}\left(y, \mathcal{N}_s(x)\right) \quad (1)$$

where $\mathcal{H}$ is a loss function, $\lambda > 0$ is a weighting coefficient and usually is a relatively large number, *e.g*., 0.9, for giving more weights to the teacher predictions than ground truths. In practice, the second term of Eq. (1) is sometimes discarded. In these cases, Eq. (1) is simplified for $\lambda = 1$ by

$$\min_{\mathcal{N}_s} \sum_{x \in \mathcal{X}} \mathcal{H}\left(\mathcal{N}_t(x), \mathcal{N}_s(x)\right). \quad (2)$$

2) Data-free Knowledge Distillation: As shown above, the standard KD requires knowing the original training data sampled from $\mathcal{X}$. Contrarily, data-free KD attempts to learn the student without being aware of $\mathcal{X}$. It is formulated by

$$\min_{\mathcal{N}_s} \sum_{x' \in \mathcal{X}'} \mathcal{H}\left(\mathcal{N}_t(x'), \mathcal{N}_s(x')\right) \quad (3)$$



(a) NYU-v2 (b) ScanNet (c) ImageNet (d) Random noises (e) KITTI (f) SceneNet

Fig. 3. Histograms of depths predicted by the teacher from (a) the NYU-v2 [46], (b) the ScanNet [7], (c) ImageNet, (d) random noises, (e) the KITTI [49], and (f) the SceneNet [37]. Note that the teacher was trained on the NYU-v2.

where $\mathcal{X}'$ is a proxy to $\mathcal{X}$ and can be either i) a set of images synthesized from $\mathcal{N}_t$, or ii) other alternative OOD datasets. Then, Eq. (3) can be solved by searching for the optimal $\mathcal{X}'$.

For image recognition, the success is attributed to the natural property that provides an inherent constraint for identifying $\mathcal{X}'$. As y denotes an object category, it is corresponded to an index of the SoftMax outputs from the last fully convolutional layer and thus provides prior information about the desired model output. Then, $\mathcal{X}'$ is constructed by

$$\arg\min_{x'} \sum_{x' \in \mathcal{X}'} \mathcal{H}(\mathcal{N}_t(x'), y) + \mathcal{R}(x') \tag{4}$$

where $\mathcal{R}$ denotes regularization terms.

Proposition 1. *The first term of Eq.* (4) *is an inherently strong constraint of image recognition that enforces the output consistency such that* $x' = x$.

Proof. Suppose that $y = \mathcal{N}_t(x)$. If $y = \mathcal{N}_t(x')$, then we have $\mathcal{N}_t(x') = \mathcal{N}_t(x)$, equivalently, $x' = x$. □

We can specify any category corresponding to an actual label of $\mathcal{Y}$ and generate sufficient images from random noises. Besides, in some works, this inherent constraint is used to transform the OOD data to the target distribution [10] or identify the most relevant data with low entropy from a large-scale dataset to the distribution of the target domain for efficient KD [3]. Finally, we have $\mathcal{P}_{\mathcal{X}'} = \mathcal{P}_{\mathcal{X}}$ to ensure KD, where $\mathcal{P}_{\mathcal{X}'}$ and $\mathcal{P}_{\mathcal{X}}$ denotes the distribution of $\mathcal{X}'$ and $\mathcal{X}$, respectively.

Unfortunately, such an inherent constraint does not hold for depth estimation. In the case of MDE, the output is a high-resolution two-dimensional map with correlated objects, not a score for a category. Therefore, we can hardly pre-specify a target depth map and learn to generate its corresponding input image as previous approaches in data-free scenarios.

B. Depth Distillation with OOD data

Given the above difficulties, data-free KD for MDE seems intractable, since no correct depth maps are available. A plausible way is to use some OOD data if we can decipher the essential requirements for $\mathcal{X}'$. Here, we consider that three factors are essential for selecting $\mathcal{X}'$: i) scene structure similarity to $\mathcal{X}$, ii) data-scale for performing KD, and iii) domain gap between $\mathcal{X}'$ and $\mathcal{X}$.

We conducted preliminary experiments to analyze how a network reacts to different types of OOD data. Specifically, we let a model trained on the NYU-v2 [46] dataset as the teacher and apply KD with several different OOD datasets with the same number of randomly sampled images.

TABLE I
RESULTS OF THE STUDENT MODEL EMPLOYED ON THE NYU-V2 TEST SET. THE STUDENT MODEL IS TRAINED VIA KNOWLEDGE DISTILLATION WITH DIFFERENT OOD DATA. EXCEPT FOR (G), ALL DATASETS HAVE APPROXIMATELY 50K IMAGES.

	Dataset ($\mathcal{X}'$)	Properties of $\mathcal{X}'$			δ_1
(a)	NYU-v2 [46]	indoor scene	real world	50K	0.808
(b)	ScanNet [7]	indoor scene	real world	50K	0.787
(c)	ImageNet [8]	single object	real-wrold	50K	0.685
(d)	Random noises	-	-	50K	0.194
(e)	KITTI [49]	outdoor scene	real world	50K	0.705
(f)	SceneNet [37]	indoor scene	simulation	50K	0.712
(g)	SceneNet [37]	indoor scene	simulation	300K	**0.742**

Fig. 3 shows the histogram of depths predicted by the teacher for different datasets, where Fig. 3 (a) denotes the histogram of the target domain. First, it is observed that the teacher yields similar depth histograms for those datasets as all exhibit a long-tail distribution. Second, although the teacher tended to produce smaller depths for OOD data, the outputted depths are still constrained in the target distribution. Based on these observations, we can say that the teacher is able to inherently produce depths in the target depth distribution even for data sampled from different domains. However, this constraint is not sufficient to ensure KD, as shown in Table I where random Gaussian noises led to the lowest performance even though they yield similar depth histograms.

Then, we analyze the effect of the above three factors by comparing target scenarios, data scale, and data domain in Table I. Except for (g), all datasets have 50,000 (50K) images and $(b) = (f) > (e) > (c) > (d)$, in terms of scene similarity to the original data, *i.e.*, (a). The δ_1 accuracy is consistent with the scene similarity in input space. In addition, (g) is an augmented version of (f) obtained by increasing the data scale. Not surprisingly, high scene structure similarity and small domain gap in images, and training models using large-scale datasets are beneficial for performance boost.

However, it is challenging to satisfy all these three conditions simultaneously. Considering the difficulties of data collection in the real world applications, we prefer to apply data-free KD for MDE with simulated images.

Despite a significant domain gap, we observe only an 11.9% accuracy drop from (a) to (f). A reasonable interpretation is that some monocular cues, such as lines, and object boundaries, are essential to activate the related neurons and form a

meaningful internal representation inside the teacher model. It can be validated in Fig. 4 that the depth maps inferred from the simulated images are perceptually correct despite being wrong in absolute scales.

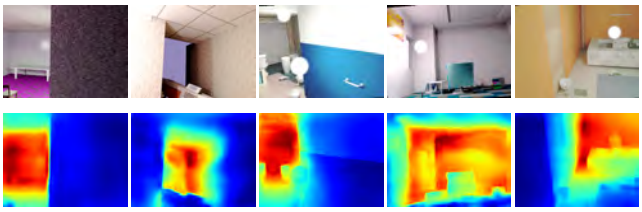


Fig. 4. Visualization of the simulated images and the depth maps estimated by the teacher.

C. Learning to Regularize Feature Distribution

We have shown that the trained model would map the out-of-distribution data to the depth histogram distribution in the target space. Admittedly, Eq. (3) is equivalent to Eq. (2) if $\mathcal{P}_{\mathcal{X}'} = \mathcal{P}_{\mathcal{X}}$. However, due to the domain gap, there is a significant discrepancy between $\mathcal{X}$ and $\mathcal{X}'$. Thus, we wish to mitigate this domain gap and accordingly improve KD. Since the original trained data is unavailable, we leverage the running average statistics captured in batch normalization (BN) as DeepInversion [55] to regularize x'. Specifically, assuming that feature statistics follow the Gaussian distribution and can be defined by mean μ and variance σ^2, then, x' is optimized through the following loss

$$\ell_{BN} = \sum_{l\in[L]} \|u_l(x') - \overline{u}_l\|_2 + \sum_{l\in[L]} \|\sigma_l^2(x') - \overline{\sigma}_l^2\|_2 \quad (5)$$

where $u_l(x')$ and $\sigma_l^2(x')$ are the batch-wise mean and variance of feature maps of the l-th convolutional layer of $\mathcal{N}_t$, respectively. $\overline{u}_l$ and $\overline{\sigma}_l^2$ are the running mean and variance of the l-th BN layer of $\mathcal{N}_t$, respectively. Eq.(5) allows regularizing x' to approach the feature distribution of the teacher model. However, this optimization requires thousands of iterations[1] for a single batch and is highly time-consuming. To tackle this problem, we use an additional network G for data transformation. Then, Eq.(5) can be rewritten as

$$\ell_{BN} = \sum_{l\in[L]} \|u_l(G(x')) - \overline{u}_l\|_2 + \sum_{l\in[L]} \|\sigma_l^2(G(x')) - \overline{\sigma}_l^2\|_2 \quad (6)$$

It is essential to ensure the fidelity of the original scenes to avoid arbitrarily meaningless transformation. Thus, we adopt an image reconstruction loss. Finally, the transformation of x' is formulated by

$$\min_G \sum_{x'\in\mathcal{X}'} (\alpha\ell_{BN} + \beta\ell_{rec}) \quad (7)$$

where $\ell_{rec} = \|x' - G(x')\|_1$ is the reconstruction error that penalizes the ℓ_1 norm of image difference, and α and β are weighting coefficients.

[1] 3000 iterations in DeepInversion.

Algorithm 1 Data-free Depth Distillation

Input: $\mathcal{X}'$: OOD images collected from simulator; $\mathcal{N}_t$: the teacher model trained on target domain; α, β: weighting coefficients used for defining loss in training G;
Hyper-parameters: Adam optimizer, initial learning rate: 0.0001, weight decay: $1e^{-4}$, training iterations: *iterations*.
Output: $\mathcal{N}_s$: the student model; G: the transformation model;

Freeze $\mathcal{N}_t$;
Initialize $\mathcal{N}_s$ and G;
for j = 1 to $iterations$ **do**
  Set gradients of $\mathcal{N}_s$ and G to 0;
  Select a batch x' from $\mathcal{X}'$;
  Let $x'_i = x'$ and $x'_j = random\_shuffle(x')$;
  Generate mixed images $\hat{x}'$ by Eq. (8);
  ▷ % Updating the student network %
  Calculate $\mathcal{N}_t(x')$, $\mathcal{N}_s(x')$, $\mathcal{N}_t(G(\hat{x}'))$, $\mathcal{N}_s(G(\hat{x}'))$;
  Calculate the depth loss by Eq. 10;
  Update $\mathcal{N}_s$;
  ▷ % Updating the transformation network %
  Calculate the loss ℓ_{BN} by Eq. 9;
  Update G;
end for

D. Distillation from Mixed Images

Since we have no clues about training data, including objects, textures, scene structures, *etc*., we naturally consider applying data augmentation to maximally cover the distributed patterns of objects in the target domain. We randomly change half of objects between two simulated images to obtain a new image with the help of semantic maps collected from the simulator. More formally, for two images x'_i and x'_j where $x'_i \in \mathcal{X}'$, $x'_j \in \mathcal{X}'$, we generate a new mixed image $\hat{x}'$ by

$$\hat{x}' = m \odot x'_i + (1 - m) \odot x'_j \quad (8)$$

where m is a binary mask obtained from the semantic map of x'_i, and randomly selects half of the classes from x'_i.

This object-wise mixing operation will lead to significant artifacts around object boundaries. In order to remove those noises, G is applied to the augmented images instead, then, Eq. (7) is rewritten as

$$\min_G \sum_{\hat{x}'\in\hat{\mathcal{X}}'} (\alpha\ell_{BN} + \beta\ell_{rec}) \quad (9)$$

where $\hat{\mathcal{X}}'$ denotes the augmented set.

E. Data-free Student Learning

We formally describe the distillation framework to enable data-free student learning. The learning objective consists of two loss terms. The first loss term adopts the plain distillation with the initial simulated images to ensure a lower bound performance. The second loss term penalizes depth differences between the teacher and the student models using images obtained from the transformation network.

TABLE II
QUANTITATIVE RESULTS ON THE NYU-V2 DATASET.

Teacher (Backbone) → Student (Backbone) Parameter Reduction		ResNet-34 [18] → ResNet-34 None		ResNet-34 [18] → MobileNet-v2 21.9 M → 1.7 M		ResNet-50 [25] → ResNet-18 63.6 M → 13.7 M		ResNet-50 [20] → ResNet-18 67.6 M → 14.9 M		SeNet-154 [4] → ResNet-34 258.4 M → 38.7 M	
Method	Data	REL ↓	δ_1 ↑	REL ↓	δ_1 ↑	REL ↓	δ_1 ↑	REL	δ_1 ↑	REL ↓	δ_1 ↑
Teacher	NYU-v2	0.133	0.829	0.133	0.829	0.134	0.824	0.126	0.843	0.111	0.878
Student		0.133	0.829	0.145	0.802	0.145	0.805	0.137	0.826	0.125	0.843
Random noises	None	0.426	0.193	0.431	0.194	0.517	0.102	0.511	0.112	0.514	0.107
DFAD [12]		0.285	0.402	0.306	0.329	0.300	0.382	0.341	0.338	0.347	0.278
KD-OOD [16]	SceneNet $\mathcal{X}_1'$	0.164	0.753	0.175	0.712	0.188	0.660	0.175	0.710	0.174	0.695
Ours		**0.155**	**0.774**	**0.168**	**0.742**	**0.173**	**0.701**	**0.167**	**0.722**	**0.156**	**0.759**
KD-OOD [16]	SceneNet $\mathcal{X}_2'$	0.158	0.761	0.165	0.742	0.180	0.676	0.172	0.713	0.161	0.738
Ours		**0.151**	**0.789**	**0.157**	**0.778**	**0.165**	**0.726**	**0.157**	**0.760**	**0.151**	**0.776**

TABLE III
DETAILS OF THE RGBD DATASETS USED IN THE EXPERIMENTS.

	Dataset	Training scenarios / images	Test scenarios / images
Target domain	NYU-v2	249 / 50688	215 / 654
	ScanNet	1513 / 50473	100 / 17607
Simulated data	SceneNet $\mathcal{X}_1'$	1000 / 50K	-
	SceneNet $\mathcal{X}_2'$	1000 / 300K	-

The optimization objective of depth distillation from the teacher model to the student model is defined by

$$\min_{\mathcal{N}_s} \sum_{x' \in \mathcal{X}'} \mathcal{L}\left(\mathcal{N}_t(x')), \mathcal{N}_s(x')\right) + \sum_{\hat{x}' \in \hat{\mathcal{X}}'} \mathcal{L}\left(\mathcal{N}_t(G(\hat{x}')), \mathcal{N}_s(G(\hat{x}'))\right) \tag{10}$$

where $\mathcal{L}$ is a loss function used for measuring the depth errors. We employ the loss function proposed in [20] that penalizes losses of depth, gradient, and normal. The details of our method are given in Algorithm 1 where $random\_shuffle$ denotes the operation of randomizing images.

IV. EXPERIMENTAL RESULTS

A. Experimental Settings

1) Implementation Details: Our learning framework includes three networks; (1) the teacher network $\mathcal{N}_t$ trained on the target domain and is fixed during training the student model; (2) the student network $\mathcal{N}_s$, which we aim to train; and (3) the transformation network G which will be also optimized during training. We train $\mathcal{N}_s$ and G for 20 epochs using the Adam optimizer with an initial learning rate of 0.0001, and reduce it to 50% for every 5 epochs. The hyper-parameters α and β controlling the data transformation are set to 0.001 for all experiments throughout the paper. We trained models with batch size of 8 in all the experiments and developed the code-base using PyTorch [42] .

2) Datasets:

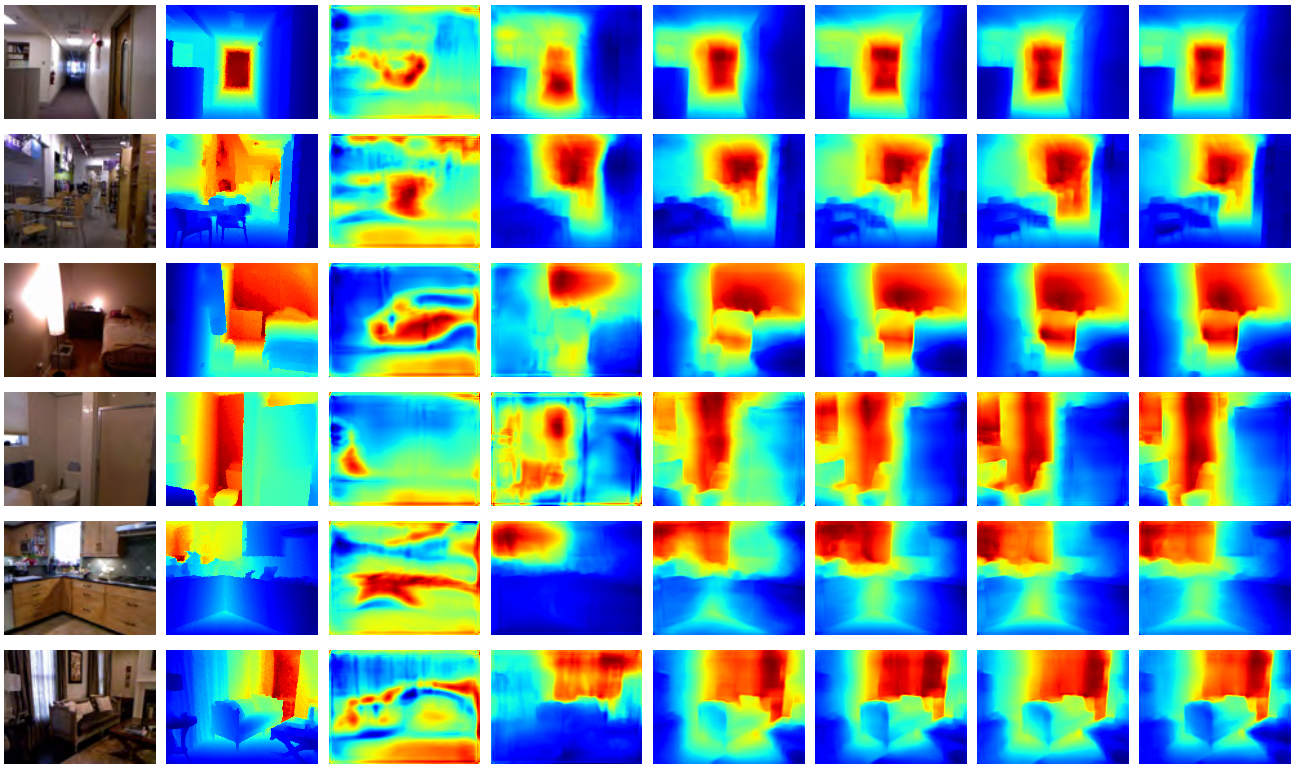
a) NYU-v2 [46]: The NYU-v2 dataset is the benchmark most commonly used for depth estimation. It is captured by Microsoft Kinect with an original resolution of 640×480, and contains 464 indoor scenes. Among them, 249 scenes are chosen for training, and 215 scenes are used for testing. We use the pre-processed data by Hu et al. [20], [19] with approximately 50,000 unique pairs of an image and a depth map with the resolution of 640×480. Following most previous studies, we resize the images to 320×240 pixels and then crop their central parts of 304×228 pixels as inputs. For testing, we use the official small subset of 654 RGBD pairs.

b) ScanNet [7]: ScanNet is a large scale RGBD dataset that contains 2.5 million RGBD images. We randomly and uniformly select a subset of approximately 50,000 samples from the training splits of 1513 scenes for training, and evaluate the models on the test set of another 100 scenes with 17K RGB pairs. We apply the same image pre-processing methods, that is, image resizing and cropping as utilized on the NYU-v2 dataset.

c) SceneNet [37]: SceneNet is a large scale synthesized dataset which contains 5 Million RGBD indoor images from over 15,000 synthetic trajectories. Each trajectory has 300 rendered frames. The original image resolution is 320×240. Thus, we only apply the center crop to yield an image resolution of 304×228.

We sample two subsets from 1000 indoor scenes of the official validation set. The two subsets have 50,000 and 300,000 images, respectively, and are denoted by $\mathcal{X}_1'$ and $\mathcal{X}_2'$ in the following texts. The detailed information of the datasets used in the experiments is given in Table III.

3) Networks: We choose multiple combinations of the teacher and student models to evaluate our models and methods extensively. For the first combination, we let the teacher and student models be the same network proposed in [18] built on ResNet-34 [15] to investigate the performance without model compression. For the second combination, we use the above ResNet-34 based network as the teacher model, and the MobileNet-v2 [44] based network as the student model in [18]. For the next two combinations, the teacher models are implemented using a ResNet-50 [15] based encoder-decoder network [25] and multi-branch depth estimation network [20], respectively. Networks of the student models are modified



(a) RGB images (b) Ground truth. (c) Random noise. (d) DFAD (e) KD-OOD with $\mathcal{X}_1'$. (f) Our results with $\mathcal{X}_1'$. (g) KD-OOD with $\mathcal{X}_2'$. (h) Our results with $\mathcal{X}_2'$.

Fig. 5. Qualitative comparison of depth maps predicted by different methods on the NYU-v2 test set.

from the teacher networks by replacing ResNet-50 with ResNet-18. For the last combination, network of the teacher model is a SeNet-154 [17] based residual pyramid network [4]. Similarly, the student model is derived from the teacher model by replacing the backbone with a smaller ResNet-34.

To implement the network of the transformation model, we use the dilated convolution [57] based encoder-decoder network modified from the saliency prediction network [21], [19] by adding symmetric skip connections between the encoder and the decoder.

4) Baselines: As discussed in Sec. III-A, most of the previous data-free KD methods cannot be applied to depth regression tasks. Thus, we choose DFAD [12] as a baseline, since this method does not apply the inherent constraint for synthesizing images. Overall, we consider the following methods as baselines for comparison.

Teacher: The teacher model trained on the target dataset.

Student: The student model trained on the target dataset.

KD-OOD: For the sake of comparison, we take KD [16] using the OOD simulated data as the strong baseline of our method. It is the first loss term used in our method.

Random noise: The student model is learned via KD with random Gaussian noise. It is also a baseline commonly used for image recognition.

DFAD: The student model is learned with data-free adversarial distillation [12] that synthesizes images from random noise with adversarial training.

B. Quantitative Comparisons

1) NYU-v2 Dataset: We first thoroughly evaluate the proposed method on the NYU-v2 dataset. We measure depth maps using the mean relative error (REL) and the δ_1 accuracy. Table II shows the quantitative results of different methods for various teacher-student combinations where the performance of the student (trained in supervised learning) exhibits an upper bound that we aim to reach. As seen, distillation with random noise yields the lowest performance, although they are shown to be effective for some toy datasets, *e.g.*, MNIST [30] and CIFAR-10 [28], for image recognition. Moreover, DFAD has also failed on the task.

Compared to the above methods, KD-OOD demonstrates much better results, showing the advance of our route that utilizes OOD simulated images. In the case of using the smaller set $\mathcal{X}_1'$, it provides 28.1% mean increase in REL and 14.0% decrease in δ_1. Most importantly, the proposed method outperforms all baselines and attains consistent performance improvement for all different teacher-student combinations. It yielded 19.7% and 9.9% performance degradation in REL and δ_1. Compared to KD-OOD, it achieves 8.4% and 4.1% mean improvement in REL and δ_1, respectively.

We then analyze the effect of utilizing the larger set $\mathcal{X}_2'$. As a result, we found a performance boost for both KD-OOD and our method in all experiments when using a more large-scale set. Our method consistently outperforms KD-OOD by 8.0% and 4.9% in REL and δ_1, respectively. Besides, our method using $\mathcal{X}_1'$ even outperforms KD-OOD using $\mathcal{X}_2'$, that is to say,

TABLE IV
THE RESULTS PROVIDED BY THE MODELS ON THE SCANNET DATASET.

Method	Data	REL ↓	δ_1 ↑
Teacher Model [18]	ScanNet	0.150	0.790
Student Model [18]		0.165	0.764
Random noise	None	0.539	0.079
DFAD [12]		0.335	0.368
KD-OOD [16]	SceneNet $\mathcal{X}_1'$	0.224	0.541
Ours		**0.196**	**0.646**
KD-OOD [16]	SceneNet $\mathcal{X}_2'$	0.200	0.618
Ours		**0.185**	**0.693**

we contribute to compressing the data-scale to $1/6$.

Another observation is that the first two teacher-student model combinations outperform the latter three. The results agree well with previous studies [51] which verified that the performance of the student model degrades when the gap of model capacity between them is significant. This problem can be well handled by using an additional assistant model [38], distilling intermediate features [32], multiple teacher models [48], and ensemble of distributions [35]. Since it is a common challenge, we leave it as future work.

Fig. 5 visualizes a qualitative comparison of different methods. It is seen that random noises produce meaningless predictions, and DFAD estimates coarse depth maps. A closer observation of maps predicted by KD, OOD and our method shows that our proposed method can estimate more accurate depth in local regions. Overall, the quantitative and the qualitative results verified the effectiveness of our approach.

2) ScanNet Dataset: To fully evaluate our method, we also test methods using the ScanNet dataset. We use the teacher and student models proposed in [18]. The results are given in Table IV. The final results are highly consistent with those obtained using NYU-v2. Both random noises and DFAD show extremely low accuracy. The proposed method outperforms KD-OOD even using the smaller set. We obtained 13.7% and 9.1% improvement in δ_1 and 17.0%, and 9.8% improvement in REL for $\mathcal{X}_1'$ and $\mathcal{X}_2'$, respectively. The performance improvement obtained on ScanNet is more significant than the improvement obtained using NYU-v2.

C. Analyses of the Transformation Network

Fig. 6 shows some examples of the input and output images of the transformation network as well as their corresponding predictions. In the figure, x_i' and x_j' denote two images randomly selected from the simulated set, and $\hat{x}'$ is the image generated by applying object-wise mixing between x_i' and x_j'. $G(\hat{x}')$ denotes the transformed image, *i.e.*, the output of G. By visually comparing $\hat{x}'$ and $G(\hat{x}')$, we observe that G tends to reduce artifacts around object boundaries such that G can produce more realistic images. It can be validated by $|\hat{x}'-G(\hat{x}')|$ (Fig. 6. (g)) where differences at object boundaries are highlighted. Furthermore, Fig. 6. (d) and (f) shows the predicted depth maps for $\hat{x}'$ and $G(\hat{x}')$, respectively. They demonstrate a clear difference, as observed in Fig. 6. (h). We quantify these differences by evaluating the whole set $\mathcal{X}_1'$. As a result, the ℓ_1-norm of the image and depth difference is 0.156 and 0.227, respectively.

TABLE V
RESULTS FOR ABLATION STUDIES.

	REL	δ_1
Original	**0.168**	0.742
Without using l_{rec}	0.172	0.735
Without using G	0.175	0.722
Without using image mixing	0.171	0.724
With $G(x')$	**0.168**	**0.748**

D. Ablation Studies

We conduct several ablation studies to analyze our approach and provide additional results on the NYU-v2 dataset. Table V gives the results. Specifically, we perform several experiments as follows:

Without using l_{rec}: In our original method, we impose the reconstruction consistency between $\hat{x}'$ and $G(\hat{x}')$ to suppress undesirable noises while training the transformation model. We relax this constraint and observe that the REL and δ_1 dropped to 0.172 and 0.735, respectively.

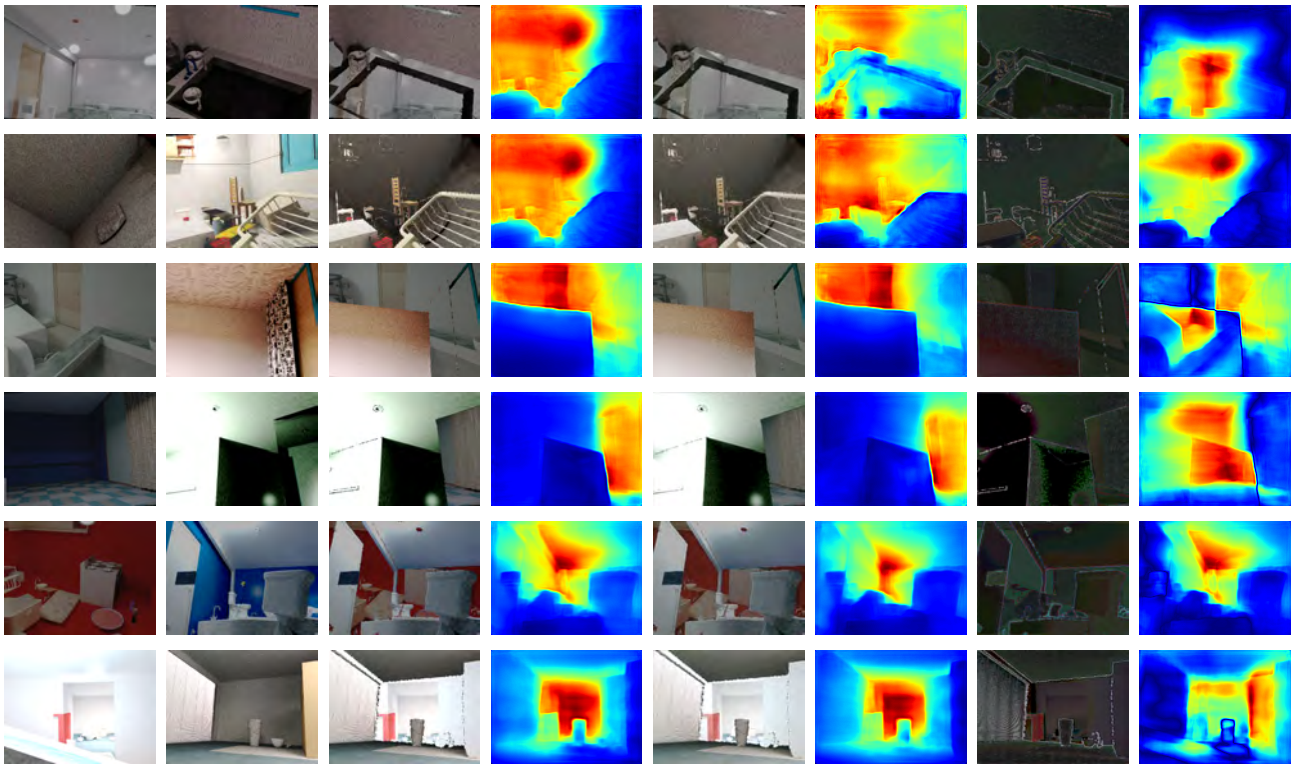
Without using G: We also test the performance while removing the transformation network in the pipeline. We directly perform distillation using x' and mixed images $\hat{x}'$. As a result, the REL and δ_1 dropped to 0.175 and 0.722, respectively.

Without using image mixing: We evaluate the effect without utilizing the object-wise image mixing. We feed the images x' to G and apply distillation with both x' and $G(x')$. We find that the REL and δ_1 dropped to 0.171 and 0.724, respectively.

With $G(x')$: Our method performs KD with the initial data x' and transformed data $G(\hat{x}')$. We also conduct an experiment to investigate the performance of applying G to x' by performing KD with both $G(x')$ and $G(\hat{x}')$. We gain a slight performance boost as δ_1 is improved from 0.742 to 0.748.

E. Invalidating KD by Adversarial Perturbation

We argue that the scene structure of the alternative OOD data is critical for successfully applying KD. To verify this, we conduct additional experiments adding adversarial perturbations to simulated images to undermine data distribution. Note that even those adversarial perturbations are imperceptible to human vision; they will generate non-robust features [24] and lead a depth estimator to a malfunction. We generate a set of adversarial images and predict depth maps from them by applying adversarial attacks. We then perform KD with those newly generated RGB and depth pairs. Following [19], we adopt IFGSM attack [29] to the ResNet-34 based teacher model [18] with different perturbation bounds ϵ. The results provided by the student model are given in Fig. 7 where $\epsilon = 0$ denotes the result of no attack. It is seen that the accuracy of depth distillation will gradually deteriorate as ϵ increases. It indicates that KD is vulnerable to adversarial attacks.



(a) x'_i (b) x'_j (c) $\hat{x}'$ (d) $\mathcal{N}_t(\hat{x}')$ (e) $G(\hat{x}')$ (f) $\mathcal{N}_t(G(\hat{x}'))$ (g) $|\hat{x}' - G(\hat{x}')|$ (h) $|\mathcal{N}_t(\hat{x}') - \mathcal{N}_t(G(\hat{x}'))|$

Fig. 6. Visual comparisons of images and depth maps where (a) and (b) are original images from the simulated set, (c) and (d) are mixed images and estimated depth maps, (e) and (f) denote transformed images of (c) and estimated depth maps, (g) and (f) denote image discrepancy and depth discrepancy, respectively.

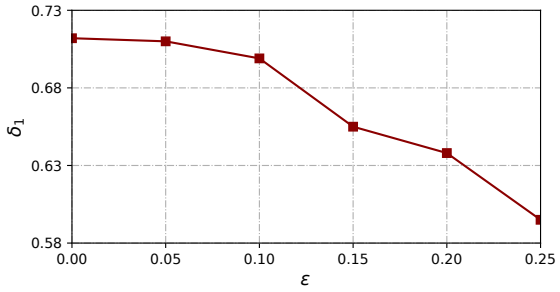


Fig. 7. Results of applying IFGSM attack to KD for monocular depth estimation.

V. Summary and Conclusion

We have studied knowledge distillation for monocular depth estimation in data-free scenarios. By analyzing the challenges of the task, we showed that a promising approach to address the challenges is to utilize out-of-distribution images as an alternative solution. We then empirically verified that i) high scene similarity, ii) large-scale dataset, and iii) small domain gap contribute to the performance boost of depth distillation through experiments with different OOD data. Given the difficulty of data collection in practice, we proposed to utilize simulated images to strengthen the applicability of KD.

In this paper, for the first time, we presented a novel framework to perform data-free knowledge distillation for monocular depth estimation. We noted that the major challenges are a lack of prior information on the scene structure and a significant domain shift between the simulated and target distribution. To remedy the first difficulty, we proposed to apply object-wise image mixing to cover the unknown distributed patterns in the target domain. To handle the second challenge, we proposed to leverage a transformation network that efficiently learns to adjust image distributions.

As a practical solution to the task, we have evaluated the effectiveness of the proposed approach for various depth estimation models and two real-world benchmark datasets. We hope our method can further inspire future explorations, shedding some light on this unexplored problem.

References

[1] D. Berthelot, N. Carlini, I. J. Goodfellow, N. Papernot, A. Oliver, and C. Raffel, "Mixmatch: A holistic approach to semi-supervised learning," in *Advances in Neural Information Processing Systems*, 2019, pp. 5050–5060.

[2] G. Chen, W. Choi, X. Yu, T. Han, and M. Chandraker, "Learning efficient object detection models with knowledge distillation," in *Advances in neural information processing systems*, 2017, pp. 742–751.

[3] H. Chen, T. Guo, C. Xu, W. Li, C. Xu, C. Xu, and Y. Wang, "Learning student networks in the wild," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 6424–6433.

[4] X. Chen, X. Chen, and Z.-J. Zha, "Structure-aware residual pyramid network for monocular depth estimation," in *International Joint Conferences on Artificial Intelligence*, 2019, pp. 694–700.

[5] X. Chen, Y. Wang, X. Chen, and W. Zeng, "S2r-depthnet: Learning a generalizable depth-specific structural representation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 3033–3042.

[6] I. Chung, S. Park, J. Kim, and N. Kwak, "Feature-map-level online adversarial knowledge distillation," in *International Conference on Machine Learning*, 2020, pp. 2006–2015.

[7] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 5828–5839.

[8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 248–255.

[9] R. Du, E. Turner, M. Dzitsiuk, L. Prasso, I. Duarte, J. Dourgarian, J. Afonso, J. Pascoal, J. Gladstone, N. Cruces, S. Izadi, A. Kowdle, K. Tsotsos, and D. Kim, "DepthLab: Real-Time 3D Interaction With Depth Maps for Mobile Augmented Reality," in *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*, 2020, pp. 829–843.

[10] G. Fang, Y. Bao, J. Song, X. Wang, D. Xie, C. Shen, and M. Song, "Mosaicking to distill: Knowledge distillation from out-of-domain data," in *Advances in Neural Information Processing Systems*, 2021, pp. 11 920–11 932.

[11] G. Fang, K. Mo, X. Wang, J. Song, S. Bei, H. Zhang, and M. Song, "Up to 100x faster data-free knowledge distillation," *ArXiv*, vol. abs/2112.06253, 2021.

[12] G. Fang, J. Song, C. Shen, X. Wang, D. Chen, D. Chen, and M. Song, "Data-free adversarial distillation," *ArXiv*, vol. abs/1912.11006, 2019.

[13] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao, "Deep ordinal regression network for monocular depth estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 2002–2011.

[14] X. Guo, J. Hu, J. Chen, F. Deng, and T. L. Lam, "Semantic histogram based graph matching for real-time multi-robot global localization in large scale environment," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 8349–8356, 2021.

[15] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[16] G. E. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *ArXiv*, vol. abs/1503.02531, 2015.

[17] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 7132–7141.

[18] J. Hu, C. Fan, H. Jiang, X. Guo, Y. Gao, X. Lu, and T. L. Lam, "Boosting light-weight depth estimation via knowledge distillation," *arXiv preprint arXiv:2105.06143*, 2021.

[19] J. Hu and T. Okatani, "Analysis of deep networks for monocular depth estimation through adversarial attacks with proposal of a defense method," *arXiv preprint arXiv:1911.08790*, 2019.

[20] J. Hu, M. Ozay, Y. Zhang, and T. Okatani, "Revisiting single image depth estimation: Toward higher resolution maps with accurate object boundaries," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2019, pp. 1043–1051.

[21] J. Hu, Y. Zhang, and T. Okatani, "Visualization of convolutional neural networks for monocular depth estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 3869–3878.

[22] Z. Huang and N. Wang, "Like what you like: Knowledge distill via neuron selectivity transfer," *ArXiv*, vol. abs/1707.01219, 2017.

[23] L. Huynh, P. Nguyen-Ha, J. Matas, E. Rahtu, and J. Heikkilä, "Guiding monocular depth estimation using depth-attention volume," in *European Conference on Computer Vision (ECCV)*, 2020, pp. 581–597.

[24] A. Ilyas, S. Santurkar, D. Tsipras, L. Engstrom, B. Tran, and A. Madry, "Adversarial examples are not bugs, they are features," in *Advances in Neural Information Processing Systems*, 2019, pp. 125–136.

[25] L. Iro, R. Christian, B. Vasileios, T. Federico, and N. Nassir, "Deeper depth prediction with fully convolutional residual networks," in *International Conference on 3D Vision (3DV)*, 2016, pp. 239–248.

[26] M. Jaritz, R. D. Charette, E. Wirbel, X. Perrotton, and F. Nashashibi, "Sparse and dense data with cnns: Depth completion and semantic segmentation," in *International Conference on 3D Vision (3DV)*, 2018, pp. 52–60.

[27] H. Jiang, L. Ding, J. Hu, and R. Huang, "Plnet: Plane and line priors for unsupervised indoor depth estimation," in *International Conference on 3D Vision (3DV)*, 2021, pp. 741–750.

[28] A. Krizhevsky, "Learning multiple layers of features from tiny images," 2009.

[29] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *International Conference on Learning Representations (ICLR)*, 2017.

[30] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, pp. 2278–2324, 1998.

[31] I.-J. Liu, J. Peng, and A. G. Schwing, "Knowledge flow: Improve upon your teachers," *ArXiv*, vol. abs/1904.05878, 2019.

[32] Y. Liu, C. Shun, J. Wang, and C. Shen, "Structured knowledge distillation for semantic segmentation." in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 2604–2613.

[33] R. G. Lopes, S. Fenu, and T. Starner, "Data-free knowledge distillation for deep neural networks," *ArXiv*, vol. abs/1710.07535, 2017.

[34] F. Ma and S. Karaman, "Sparse-to-dense: Depth prediction from sparse depth samples and a single image," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 1–8.

[35] A. Malinin, B. Mlodozeniec, and M. J. F. Gales, "Ensemble distribution distillation," *ArXiv*, vol. abs/1905.00076, 2020.

[36] M. Mancini, G. Costante, P. Valigi, and T. A. Ciarfuglia, "Fast robust monocular depth estimation for obstacle detection with fully convolutional networks," in *IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 4296–4303.

[37] J. McCormac, A. Handa, S. Leutenegger, and A. J. Davison, "Scenenet rgb-d: 5m photorealistic images of synthetic indoor trajectories with ground truth," *arXiv preprint arXiv:1612.05079*, 2016.

[38] S. I. Mirzadeh, M. Farajtabar, A. Li, N. Levine, A. Matsukawa, and H. Ghasemzadeh, "Improved knowledge distillation via teacher assistant," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, 2020, pp. 5191–5198.

[39] G. K. Nayak, K. R. Mopuri, and A. Chakraborty, "Effectiveness of arbitrary transfer sets for data-free knowledge distillation," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2021, pp. 1430–1438.

[40] V. Nekrasov, T. Dharmasiri, A. Spek, T. Drummond, C. Shen, and I. Reid, "Real-time joint semantic segmentation and depth estimation using asymmetric annotations," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019, pp. 7101–7107.

[41] V. Olsson, W. Tranheden, J. Pinto, and L. Svensson, "Classmix: Segmentation-based data augmentation for semi-supervised learning," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2021, pp. 1368–1377.

[42] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., 2019, pp. 8024–8035.

[43] A. Pilzer, S. Lathuilière, N. Sebe, and E. Ricci, "Refine and distill: Exploiting cycle-inconsistency and knowledge distillation for unsupervised monocular depth estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 9760–9769.

[44] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 4510–4520.

[45] Z. Shen, Z. He, and X. Xue, "Meal: Multi-model ensemble via adversarial learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 4886–4893.

[46] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *European Conference on Computer Vision (ECCV)*, vol. 7576, 2012, pp. 746–760.

[47] Z. Song, J. Lu, Y. Yao, and J. Zhang, "Self-supervised depth completion from direct visual-lidar odometry in autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, 2021.

[48] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Advances in Neural Information Processing Systems*, 2017, pp. 1195–1204.

[49] J. Uhrig, N. Schneider, L. Schneider, U. Franke, T. Brox, and A. Geiger, "Sparsity invariant cnns," in *International Conference on 3D Vision (3DV)*, 2017, pp. 11–20.

[50] T. van Dijk and G. C. de Croon, "How do neural networks see depth in single images?" in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 2183–2191.

[51] L. Wang and K.-J. Yoon, "Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 6, pp. 3048–3068, 2021.

[52] Y. Wang, X. Li, M. Shi, K. Xian, and Z. Cao, "Knowledge distillation for fast and accurate monocular depth estimation on mobile devices," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2021, pp. 2457–2465.

[53] D. Wofk, F. Ma, T.-J. Yang, S. Karaman, and V. Sze, "Fastdepth: Fast monocular depth estimation on embedded systems," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019, pp. 6101–6108.

[54] Y. Xu, Y. Wang, H. Chen, K. Han, C. Xu, D. Tao, and C. Xu, "Positive-unlabeled compression on the cloud," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019, pp. 2561–2570.

[55] H. Yin, P. Molchanov, J. M. Alvarez, Z. Li, A. Mallya, D. Hoiem, N. K. Jha, and J. Kautz, "Dreaming to distill: Data-free knowledge transfer via deepinversion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 8715–8724.

[56] J. Yoo, M. Cho, T. Kim, and U. Kang, "Knowledge extraction with no observable data," in *Advances in Neural Information Processing Systems*, 2019, pp. 2701–2710.

[57] F. Yu, V. Koltun, and T. Funkhouser, "Dilated residual networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 472–480.

[58] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. J. Yoo, "Cutmix: Regularization strategy to train strong classifiers with localizable features," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 6022–6031.

[59] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *International Conference on Learning Representations (ICLR)*, 2018.

Junjie Hu (Member, IEEE) received the M.S. and Ph.D. degrees from the Graduate School of Information Science, Tohoku University, Sendai, Japan, in 2017 and 2020, respectively. He is currently a Research Scientist with the Shenzhen Institute of Artificial Intelligence and Robotics for Society. His research interests include machine learning, computer vision, and robotics.

Chenyou Fan serves as Associate Profession of the School of Artificial Intelligence, South China Normal University, China. He received the B.S. degree in computer science from the Nanjing University, China, in 2011, and the M.S. and Ph.D. degrees from Indiana University, USA, in 2014 and 2019, respectively. His research interests include machine learning and computer vision. He served in the program committee of CVPR, NeurIPS, ACM MM and top AI journals for more than 20 times.

Mete Ozay (M'09) received the B.Sc., M.Sc., Ph.D. degrees in mathematical physics, information systems, and computer engineering & science from METU, Turkey. He has been a visiting Ph.D. and fellow in the Princeton University, USA, a research fellow in the University of Birmingham, UK, and an Assistant Professor in the Tohoku University, Japan. His current research interests include pure and applied mathematics, theoretical computer science & neuroscience.

Hualie Jiang Hualie Jiang received the M.Eng. degree in Pattern Recognition and Intelligent Systems from Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China, in 2017 and the Ph.D. degree at The Chinese University of Hong Kong, Shenzhen, China, in 2021. He is interested in 3D computer vision, including passive monocular depth estimation and active structured-light sensing.

Tin Lun Lam (Senior Member, IEEE) received the B.Eng. (First Class Hons.) and Ph.D. degrees in robotics and automation from the Chinese University of Hong Kong, Hong Kong, in 2006 and 2010, respectively. He is currently an Assistant Professor with the Chinese University of Hong Kong, Shenzhen, China, and the Director of Center for the Intelligent Robots, Shenzhen Institute of Artificial Intelligence and Robotics for Society. He has authored or coauthored two monographs and more than 50 research papers in top-tier international journals and conference proceedings in robotics and AI. His research interests include multirobot systems, field robotics, and collaborative robotics.