Stereo Matching by Self-supervision of Multiscopic Vision

Weihao Yuan, Yazhan Zhang, Bingkun Wu, Siyu Zhu, Ping Tan, Michael Yu Wang, and Qifeng Chen

***Abstract*— Self-supervised learning for depth estimation possesses several advantages over supervised learning. The benefits of no need for ground-truth depth, online fine-tuning, and better generalization with unlimited data attract researchers to seek self-supervised solutions. In this work, we propose a new self-supervised framework for stereo matching utilizing multiple images captured at aligned camera positions. A cross photometric loss, an uncertainty-aware mutual-supervision loss, and a new smoothness loss are introduced to optimize the network in learning disparity maps end-to-end without ground-truth depth information. To train this framework, we build a new multiscopic dataset consisting of synthetic images rendered by 3D engines and real images captured by real cameras. After being trained with only the synthetic images, our network can perform well in unseen outdoor scenes. Our experiment shows that our model obtains better disparity maps than previous unsupervised methods on the KITTI dataset and is comparable to supervised methods when generalized to unseen data. Our source code and dataset are available at `https://sites.google.com/view/multiscopic`.**

I. Introduction

Accurate depth estimation is essential for numerous applications such as robotic manipulation, autonomous driving, and 3D reconstruction [1–4]. Various industrial solutions have employed stereo cameras for dense depth sensing. A vital benefit of a stereo camera is that the two cameras are axis-aligned so that the corresponding pixels between two images can lie on the same row. It appears that adding more axis-aligned cameras can help obtain high-quality depth maps. In this work, we study a novel problem for self-supervised stereo matching: given axis-aligned multiscopic images, can we train a stereo matching model for better depth sensing?

Recent deep-learning-based methods for depth estimation have dominated most benchmarks [5–8]. However, these learning-based methods do not perform stably on unseen scenarios since the methods rely heavily on the training data. The ground-truth depth is hard to collect in the real world so that the size of the training data with ground-truth depth is usually small [1]. In this case, the network trained with limited data is prone to overfitting the training data rather than understanding the underlying structure of the task. Thus, most deep-learning-based approaches do not have stable generalization ability, which restricts their applications in the industry.

To this end, researchers are exploring unsupervised learning for depth estimation [9, 10]. Compared to supervised approaches, unsupervised learning does not need the ground-truth depth, so it can be trained with a large amount of data. Also, since there is no ground-truth depth, the network needs to find the underlying pattern and geometry structure of images for stereo matching rather than simply overfitting the real depth value. Another advantage of unsupervised learning is that online fine-tuning becomes plausible [11–13]. As the ground truth is not needed for training a model, the model can be updated when processing the new inputs, which can lead to various practical applications.

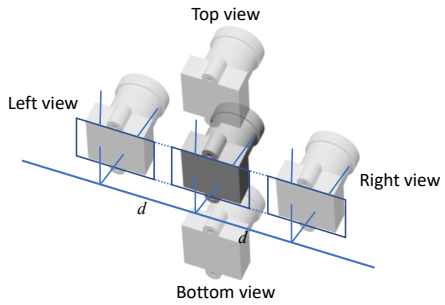


Fig. 1. Our multiscopic vision structure. Images are taken at parallel, co-planar, and same-parallax camera positions, such that the pixel disparities between adjacent images are the same.

The idea of exploiting more input data to assist the learning has been recently explored by some researchers [9, 10, 14]. For example, a model for unsupervised monocular depth estimation can be trained with stereo images [9]. Inspired by this idea, we wonder if the unsupervised stereo matching can be improved with three or more images during training.

In this paper, we explore self-supervised learning depth estimation utilizing multiple images taken at aligned camera positions as input during training. We assume that multiple images are captured at horizontally or vertically aligned positions. As the stereo structure brings some benefits to depth estimation, the multiscopic structure introduces more constraints that can supervise the learning [15]. These cameras are parallel, co-planar, and with the same baselines, as shown in Fig. 1, such that the pixel disparities between adjacent images are the same. In this case, the disparity maps from the center view to any surrounding view should be exactly the same. Furthermore, for the left and right views, the pixel shift exists only in the horizontal axis, and for the bottom and top view, the pixel shift exists only in the vertical axis.

By exploiting these constraints, we can train a network to predict a better disparity map end-to-end without ground-truth depth information, as shown in Fig. 2. Uncertainty-aware mutual supervision between different disparity maps

W. Yuan (qianmu.ywh@alibaba-inc.com), S. Zhu and P. Tan are with AI Lab, Alibaba Cloud, China. Y. Zhang, M. Wang and Q. Chen (cqf@ust.hk) are with the Dept. MAE and Dept. CSE, Hong Kong University of Science and Technology, Hong Kong SAR, China. B. Wu is with the Beihang University, China.

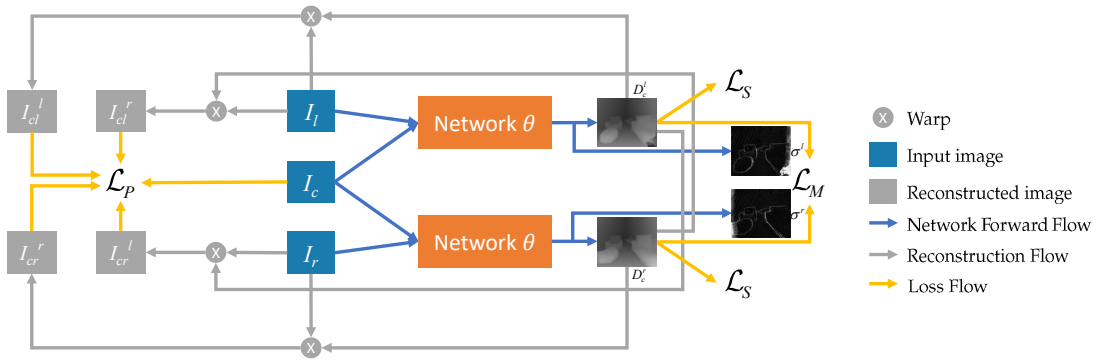


Fig. 2. Self-supervised learning framework for three-image multiscopic structure. Two networks sharing the same parameters compute the disparity D_c^l, D_c^r and uncertainty map σ^l, σ^r end-to-end, which are then used to cross warp the left and right images I_l, I_r back to the center view, generating four reconstructed images $I_{cl}^l, I_{cl}^r, I_{cr}^l, I_{cr}^r$. The cross photometric loss $\mathcal{L}_P$ is built between them and the real center image I_c, the uncertainty-aware mutual-supervision loss $\mathcal{L}_M$ is built between D_c^l and D_c^r conditioned on σ^l, σ^r, and the smoothness loss $\mathcal{L}_S$ is built for each disparity map.

serves as the self-supervising signal to optimize the network. Since the learning is self-supervised, it needs to find the underlying relationship of the multiscopic images to compute the disparity. Our experiment shows that the learned network can be generalized to unseen scenarios well, as there is no chance to overfit the ground-truth depth. After being trained with only synthetic images, our model performs well in real-world scenarios.

To make our unsupervised learning framework work and to promote more works in multiscopic vision, we propose a new dataset with 1300 scenes of multiscopic images as there is no available large dataset for multiscopic vision. There are a large number of synthetic images rendered by 3D engines [16, 17] and a small number of real images taken by our multiscopic cameras. For the synthetic images, ground-truth depths are also provided.

Our main contributions concerning the self-supervised learning depth estimation framework and the multiscopic dataset are summarized as follows:

1) We exploit multiple images in a multiscopic structure for training a network to predict the disparity maps without the need for ground-truth depth. Optimization is built from the mutual supervision between confident areas of disparities obtained from different center-surround image pairs.

2) We build a new dataset of multiscopic images of 1200 scenes rendered in 3D engines and 100 scenes captured in the real world with our multiscopic cameras. Trained on only synthetic images, our model can be generalized to perform well on real-world data.

3) The experiment shows that our model outperforms other unsupervised approaches for stereo matching by a large margin and is comparable to supervised methods when generalized to unseen data.

II. Related Work

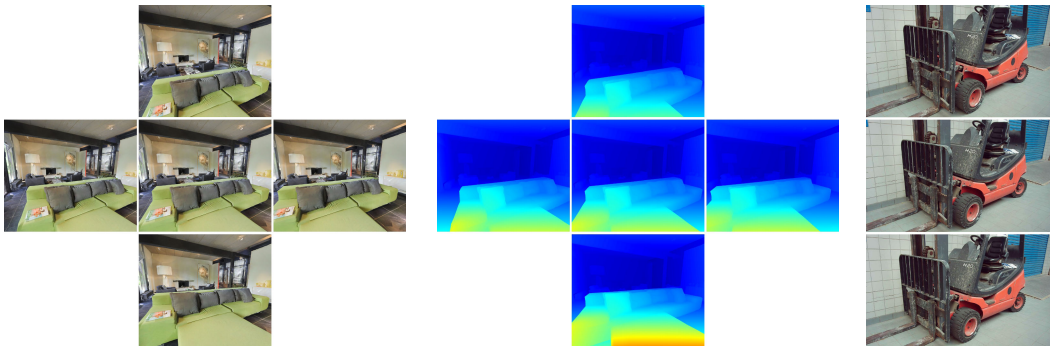
Multiscopic depth estimation. All previous depth estimation works using multiscopic images are based on traditional matching methods [18–21], since there is no available large dataset to train a network. Various methods take advantage of some constraints in multiscopic structure to improve the estimation results and only evaluate it on a small synthetic dataset, the Tsukuba images [22].

Deep depth estimation. Deep learning based stereo methods explore using neural networks to replace one [23] or all steps [5–7] in traditional stereo matching to compute the disparity map. Exploiting the network to compute the matching cost of two image patches was first proposed in MC-CNN [23]. To further take advantage of the neural networks, end-to-end approaches are proposed to remove all traditional matching steps by directly using the network [5–7]. Additionally, optimization methods in traditional stereo matching, such as semi-global matching, are also introduced to assist the deep methods in more meaningful ways [7]. Heretofore, end-to-end learning approaches have dominated most benchmarks and show significant advantages.

Unsupervised depth estimation. Despite the success of learning-based approaches in some specific datasets, they need plenty of ground-truth training data, which is expensive to acquire. To overcome this, more researchers are seeking assistance from unsupervised learning. Some unsupervised frameworks were first proposed for monocular depth estimation. The monocular depth network is supervised by the reconstruction loss between the stereo image pairs [24–26]. The reconstructed images are synthesized by warping one view to the other view using the disparity maps. This is enhanced by introducing the consistency of disparity maps between the left view and the right view [9].

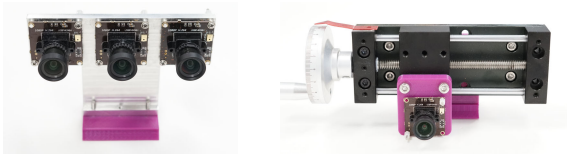
Instead of relying on the stereo images, some works combine the ego-motion and optical flow estimation to help the unsupervised depth learning from videos [27, 28]. The loss is built by warping adjacent views with the computed depth, pose, and optical flow. Also, some geometric consistency losses are introduced to improve the estimation [28].

For stereo depth estimation, there are some losses similar to monocular approaches and more constraints that can further supervise the learning. Zhou et al. [27] evaluate the matching result and pick the confident patch pairs as the new training data, in which way the network is iteratively optimized. Godard et al. [9] realize the stereo matching using the same reconstruction loss and left-right consistency



(a) Color images (b) Ground-truth disparity maps (c) Real images

Fig. 3. Example scenes of multiscopic images in our dataset. The synthetic color images (a) and corresponding perfect disparity maps (b) displayed in Jet colormap are presented. (c) is one set of images obtained by the multiscopic system in the real world.



(a) Multi-eye (b) Moving-eye

Fig. 4. Multiscopic systems designed to capture real multiscopic images. Multi-eye design (a) is using multiple identical cameras while moving-eye design (b) is using one camera sliding on a precision linear rail.

loss as the monocular depth task. This is further improved by introducing occlusion-aware loss [29] or heuristic loss [13]. Then more works try to explore supervision from the consistency in stereo videos [10, 14, 30, 31], by utilizing the consistency in stereo depth, optical flow, and ego-motion. Some other works also introduce parallax-attention [32] or seek the assistance from other networks [33].

So far, a limited number of works have been dedicated to unsupervised deep depth estimation in the multiscopic structure that provides more constraints to assist unsupervised learning [15, 34]. Similar to supervising the monocular depth estimation with stereo images, we show that we can supervise the stereo matching with multiscopic images to obtain a better disparity map by exploiting the inner shape of the multiscopic structure. To the best of our knowledge, our work is the first deep stereo depth estimation method utilizing multiscopic images.

III. Multiscopic Dataset

In this section, we introduce the details of our multiscopic vision system and describe how we build our multiscopic dataset in virtual environments and the real world.

In a multiscopic vision structure, multiple images are captured at axis-aligned camera positions, such that these images are taken with parallel optical axes on the same image plane, and with the same parallaxes, as shown in Fig. 1. Multiscopic vision structure brings clear benefits to depth estimation [35]. The disparity of a pixel from the center view to any surrounding view should be the same, which is a strong constraint to optimize the result. Additionally, the occluded regions in one view can be perceived from other views, in which case there are no occlusion problems for the entire multiscopic system [20].

Some interesting research can be conducted in this setup. For instance, multiscopic images can be utilized for better image enhancement since there are more constraints to be enforced, such as super-resolution with multiple images [36]. Also, novel view synthesis [37] can be done in a multiscopic structure, such as predicting the right view provided with only the left and center views.

Since there is no available large dataset for multiscopic vision, we build a new dataset composed of 1200 scenes of synthetic multiscopic data rendered by 3D engines and 100 scenes of real images taken with real-world cameras.

A. Synthetic Data

The synthetic images are rendered using the 3D render engine Habitat-sim [38] and SceneNet RGB-D [16], with 3D models from Matterport3D [17], Gibson [39], and ShapeNet [40]. Habitat-sim is used to render Matterport3D as well as Gibson, and SceneNet RGB-D is used to render ShapeNet. The models from Matterport3D and Gibson are collected from real indoor spaces by reconstruction from data acquired using real-world RGB cameras, depth cameras, and 3D scanning sensors. Most models within them are ordinary houses with daily-life objects like furniture, decorations, and instruments. The models from ShapeNet are some virtual indoor space models in which we can randomly place all kinds of 3D object models, like toys, plants, cars, or even airplanes. So when we render the scenes using models from ShapeNet, we adjust the lighting condition randomly while we keep the light the same as the real world when we render the models from Matterport3D and Gibson.

There are more than 20K raw scenes of images. We remove the similar scenes and keep 1200 scenes in total that are unlike each other. This makes these 1200 scenes representative of visual variation. For each scene in our dataset, there are five images taken with multiscopic vision structure, the center, left, right, bottom, and top view, as shown in Fig. 3. More scenes can be seen in the supplementary materials. The resolution of all images is 1280×1080, and every color image has its corresponding ground-truth disparity map. As a multiscopic structure, the baselines between the

four surrounding images and the center image are the same and vary from 0.05 to 0.2 meters. The value range of the disparity maps is from 0 to 255.

B. Real-world Data

Real-world multiscopic data are taken with one or multiple RGB cameras. Multiscopic images require aligned camera locations, which is not easy to control. Therefore, to capture images in this structure, we design two architecture: The first architecture is composed of three identical cameras with Sony IMX322 inside, whose resolution is 1920×1080. These three cameras are aligned on a horizontal line, as displayed in Fig. 4(a). They are parallel, co-planar, and with the same baseline of 4 centimeters. The second design is a moving-eye system. Although we utilize the same cameras, a multi-eye structure still suffers from the problem that the cameras' parameters are not exactly the same. To overcome this, we place one camera on a precision linear sliding rail and slide it horizontally on its image plane, as displayed in Fig. 4(b). In this system, three images are taken at three locations to obtain multiscopic images. The baselines of adjacent locations vary from 3 to 4 centimeters.

Images obtained by a moving-eye system have strictly identical parameters, but sliding a camera needs some time so that it cannot take images of dynamic scenes. Thus, we combine these two systems to build a real-world image dataset. There are 100 scenes of real-world images in our dataset. For every scene, there are the left, center, and right images, as shown in Fig. 3(c). More scenes are shown in the supplementary materials. Some images are cropped to reduce the vignette influence and remove messy environments, after which the resolutions vary from 1400×1000 to 1920×1080. For the real data, there is no ground-truth depth information.

IV. Self-supervised Multiscopic Matching

A. Learning Framework

In rectified stereo matching, pixels shift along the horizontal axis such that the searching for pixels correspondence is simplified. The depth map can be constructed from a computed disparity map that establishes pixel correspondences between the left image I_l and the right image I_r. With a disparity map, the right image can be warped back to the left image to produce a synthetic left image I'_l. Ideally, if the disparity map D_l is correctly estimated, the synthetic left image I'_l should be the same as I_l except for occluded regions. Then the discrepancy between them can be used as a photometric loss in training a self-supervised model [9].

In multiscopic vision, a similar photometric loss can be enforced between any two axis-aligned images. Since the baselines between adjacent camera positions are the same, more constraints could be established in a constrained way. As shown in Fig. 2, we deal with a multiscopic system with three images without loss of generality. Benefiting from the multiscopic architecture, the disparities between the center image and any surrounding image should be the same, with the center image as the reference. Therefore, one disparity map can be used to warp all surrounding images back to the center image to work as a cross photometric loss. The disparity D_c^r between the center image I_c and the right image I_r can be used to not only warp I_r to produce a synthetic center image I_{cr}^r, but also warp I_l to produce I_{cl}^r, as shown in Fig. 5. On the one hand, this cross warping consistency can strengthen the photometric supervision. On the other hand, it can address the occlusion problem which troubles stereo vision, e.g., the invisible pixel in the right view would cause a large loss between I_{cl}^r and I_c.

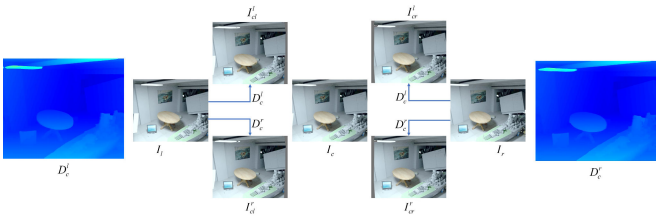


Fig. 5. Cross warping between I_l, I_c, I_r using disparity maps D_c^l, D_c^r. The cross photometric loss is then established between the warping results $I_{cl}^l, I_{cl}^r, I_{cr}^r, I_{cr}^l$ and the center image I_c.

In a multiscopic structure, another constraint is that the disparity maps between the center image and all surrounding images should be the same. The discrepancy in disparity values indicates inaccurate disparity estimation. Thus the difference between multiple disparities could work as mutual supervision, as shown in Fig. 2. This also helps handle the occlusion problem, which is usually problematic in stereo matching. The occluded regions of the center image in one surrounding image can be seen in other images. For example, the occluded areas between the left image and the center image are hard to be estimated correctly in D_c^l, but these parts are estimated well in D_c^r. In this case, we estimate an uncertainty map for each disparity map. In the uncertain areas like occluded pixels in D_c^l, we take the confident pixels in D_c^r as supervision labels and optimize the estimation network for D_c^l.

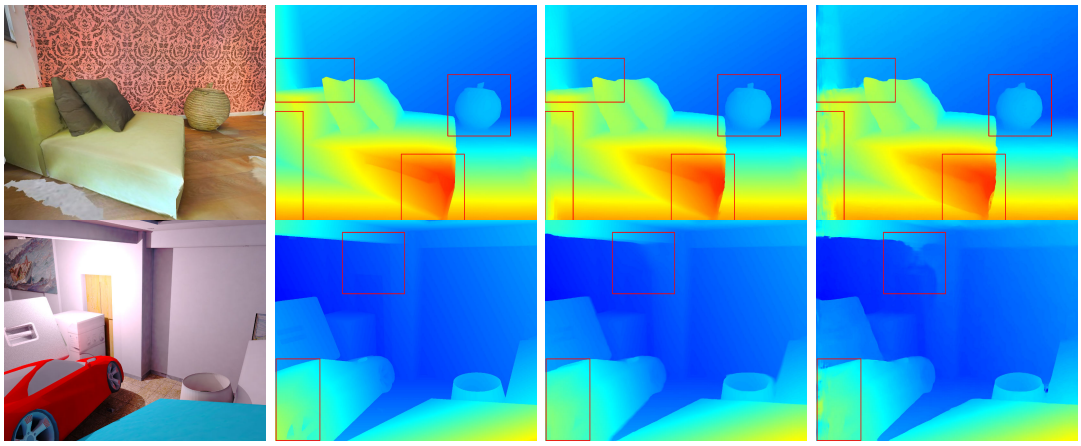
B. Network Structure

While the design of the network structure is not our focus in this work, we use PSMNet [6] as our network backbone, which is an end-to-end disparity regression model. We trim the structure by reducing some layers for efficiency and add an uncertainty head. Details of the network are provided in the supplementary materials. The generated cost volume is used to regress a disparity map by the disparity head and is used to regresses an uncertainty map by the uncertainty head.

In our three-image multiscopic system, there are two matchings. One is between the center image and the left image, and the other is between the center image and the right image. We use two matching networks sharing the same parameters to calculate the disparity maps D_c^l and D_c^r.

C. Loss Function

In our multiscopic self-supervised framework, we utilize the cross photometric loss $\mathcal{L}_P$ and the uncertainty-aware mutual-supervision loss $\mathcal{L}_\sigma, \mathcal{L}_M$ to guide the learning process. Additionally, a regularization term $\mathcal{L}_S$ is also intro-



Center view | Ground truth | Our model | PSMNet

Fig. 6. Example outputs on synthetic test images. The center-view color images, the ground-truth disparity maps, the output disparity maps of our self-supervised model, and the output of supervised method PSMNet are shown.

Method	Training set	Gound truth	EPE	Bad0.5	Bad1	Bad2
Godard et al. [9]	Synthetic	✗	2.16	35.22%	21.67%	10.83%
Ours	Synthetic	✗	1.19	**16.54%**	**9.46%**	**6.07%**
PSMNet [6]	Synthetic	✓	**1.07**	23.01%	11.69%	6.36%
MC-CNN [23]	KITTI	✓	3.55	42.88%	27.18%	20.05%
PSMNet [6]	Cityscapes + KITTI	✓	2.94	75.53%	50.12%	18.09%

TABLE I. Results on the synthetic test set.

duced to improve the smoothness of the output disparity. Our overall loss is the sum of these terms with weighting factors $\lambda_1, \lambda_2, \lambda_3, \lambda_4$:

$$\mathcal{L} = \lambda_1 \mathcal{L}_P + \lambda_2 \mathcal{L}_\sigma + \lambda_3 \mathcal{L}_M + \lambda_4 \mathcal{L}_S. \tag{1}$$

1) Cross Photometric Loss: $\mathcal{L}_P$ is used to evaluate how similar the reconstructed images $I^l_{cl}, I^r_{cl}, I^r_{cr}, I^l_{cr}$ are to the real center image I_c, where

$$\begin{aligned} I^l_{cl}(u,v) &= I_l(u + D^l_c(u,v), v), \\ I^r_{cl}(u,v) &= I_l(u + D^r_c(u,v), v). \end{aligned} \tag{2}$$

I^r_{cr}, I^l_{cr} are calculated in a similar way. The warping process is illustrated in Fig. 5.

We use structural similarity (SSIM) [41] loss to calculate the image discrepancy as

$$\mathcal{L}_P(I^l_{cl}, I_c) = \frac{1 - SSIM(I^l_{cl}, I_c)}{2}, \tag{3}$$

where the SSIM window size is set to 5. The other three parts are calculated in a similar way, and the final cross photometric loss is computed by taking the average:

$$\mathcal{L}_P = \frac{\mathcal{L}_P(I^l_{cl}) + \mathcal{L}_P(I^r_{cl}) + \mathcal{L}_P(I^l_{cr}) + \mathcal{L}_P(I^r_{cr})}{4}. \tag{4}$$

2) Uncertainty-aware Mutual-supervision Loss: $\mathcal{L}_M$ utilizes the output of one network to supervise the other network. The discrepancy between the disparity output D^l_c generated by the center-left network and D^r_c generated by the center-right network means erroneous matching. Thus we learn an uncertainty map for each disparity output. In uncertain areas, we supervise the uncertain output with the other output, and in confident areas, we calculate the consistency loss between two outputs.

Following the heteroscedastic aleatoric uncertainty setup in [42], we model the reconstruction task with uncertain L1 loss as

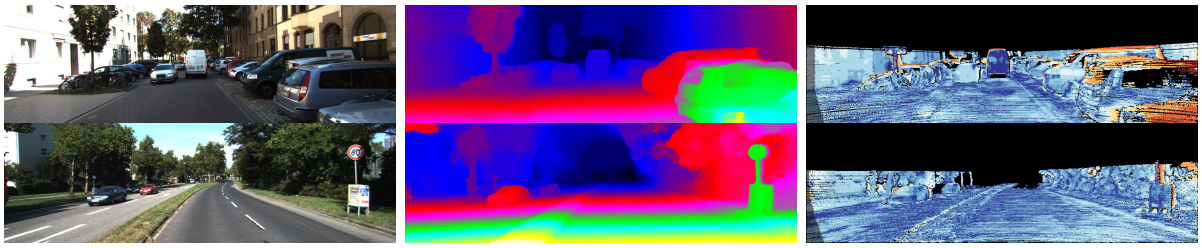
$$\mathcal{L}_\sigma(I^l_{cl}) = \sum_{u,v} \frac{\sqrt{2}}{\sigma^l(u,v)} |I^l_{cl}(u,v) - I_c(u,v)| + \log \sigma^l(u,v), \tag{5}$$

where σ^l is the estimated uncertainty map generated by center-left image pair. Then $\mathcal{L}_\sigma$ is computed by taking the average of $\mathcal{L}_\sigma(I^l_{cl}), \mathcal{L}_\sigma(I^r_{cl}), \mathcal{L}_\sigma(I^l_{cr}), \mathcal{L}_\sigma(I^r_{cr})$.

With the uncertainty map σ_l, σ_r, we can do the mutual-supervising as

$$\mathcal{L}_M = \begin{cases} ||D^l_c - D^r_c||, & \text{if } \sigma_l, \sigma_r < T_\sigma \\ ||D^l_c - \text{label}(D^r_c)||, & \text{if } \sigma_l \geq T_\sigma, \sigma_r < T_\sigma \\ ||D^r_c - \text{label}(D^l_c)||, & \text{if } \sigma_r \geq T_\sigma, \sigma_l < T_\sigma. \end{cases} \tag{6}$$

where $T_\sigma = e$ is a threshold over which the disparity estimation is regarded as uncertain, and the label operation converts the confident estimation to supervision signal that is used to optimize the estimation of the other network. In regions where both estimations are uncertain, we do not calculate the loss. This loss could be especially useful in occluded regions. When the matching between the center-left images is wrong due to occlusion, it can be corrected by the label from the center-right matching.



Left image Disparity map Error map

Fig. 7. Qualitative results on KITTI 2015 test set.

Method	#Infer input	Train set D1-all	Test set D1-all
Zhou et al. 2017 [27]	2	9.91%	–
Godard et al. 2017 [9]	2	9.194%	–
Li et al. 2018 [43]	2	–	8.98%
Guo et al. 2018 [26]	2	7.06%	–
UnOS 2019 [10]	2	7.073%	–
UnOS 2019 [10]	4	5.943%	6.67%
Wang et al. 2020 [32]	2	–	7.23%
Flow2Stereo 2020 [14]	2	6.13%	6.61%
Ours	2	**3.82%**	**4.43%**
PSMNet	2	1.83%	2.32%
PSMNet*	2	5.90%	–

TABLE II. Quantitative evaluation on the KITTI 2015 dataset. D1-all is the percentage of stereo disparity outliers in all regions of the first frame. End-point error smaller than 3 pixels or 5% is considered correct.

3) Smoothness Loss: $\mathcal{L}_S$ encourages the adjacent pixels to have similar disparities, especially for those with a similar color. The punishment is enforced for discontinuous disparities and determined by the disparity variation and color difference, as

$$\begin{aligned}\mathcal{L}_S = f_x(D_c^l)e^{-\gamma||\partial_x I_c||} + f_y(D_c^l)e^{-\gamma||\partial_y I_c||} + \\ f_x(D_c^r)e^{-\gamma||\partial_x I_c||} + f_y(D_c^r)e^{-\gamma||\partial_y I_c||},\end{aligned} \tag{7}$$

where γ is a scale factor related to the image intensity normalization and is set to 10 in our work. $f_{x/y}$ is composed of a discontinuous loss K plus the disparity differential as

$$f_x(D) = \begin{cases} K + ||\partial_x D||, & \text{if } ||\partial_x D|| > 0.5 \\ ||\partial_x D||, & \text{otherwise.} \end{cases} \tag{8}$$

The disparity difference larger than 0.5 is considered discontinuity and is punished by an additional $K = 10$.

V. Experiments

In this section, we first present the details of our experimental setup. Afterward, we evaluate our model on the synthetic test set, KITTI 2015 dataset, and real-world images. Also, ablation studies are performed to provide insight into the effect of each module in our framework.

A. Implementation Details

The self-supervised learning framework is implemented in Pytorch and trained on two Nvidia GTX 1080 Ti GPUs. The batch size is set to 4 and the weights $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ are set to $1, 0.01, 0.03, 0.03$. Images are randomly cropped to 512×256 for every epoch during training to reduce the size and work as data augmentation. The network is optimized end-to-end with Adam ($\beta_1 = 0.9$, $\beta_1 = 0.999$) and the learning rate is set to 0.0001.

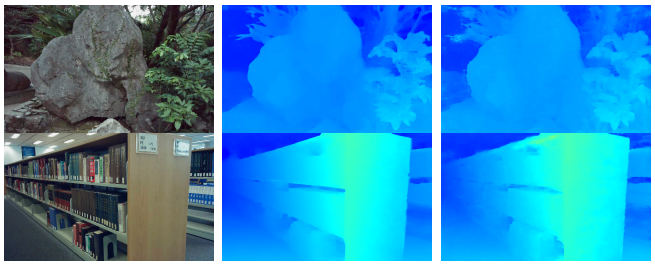
Our network is only trained using 1,000 scenes of synthetic data. The performance trained with additional real multiscopic images is similar since our synthetic images are quite photo-realistic. During training, three multiscopic images are used as the input of our learning framework because the performance of using five images is similar, and our real-world system captures three images. For evaluation, we only use two stereo images for depth estimation. Examples of the output D_c^r between the right view and the center view of test images are shown in Fig. 6. The output disparity maps are very close to the ground truth disparity maps. For the computation efficiency, since the backbone of our network is based on PSMNet [6], the network forward time is similar at around 0.5 s for one KITTI image pair. By choosing the backbone, the balance between accuracy and efficiency can be achieved.

B. Evaluation

1) Synthetic data: We build a synthetic test set consisting of 200 scenes of multiscopic images. After the training, we evaluate our model on these images to examine its performance on synthetic images. The center view and the right view are used as the input to the network. Four metrics are applied, and the results are summarized in Table I. The EPE is the average endpoint error, Bad0.5, Bad1, and Bad2 is the percentage of pixels whose error is higher than 0.5, 1, and 2, respectively.

We also compare our method with some unsupervised and supervised methods when trained on our synthetic dataset, as presented in Table I. The results show that our method is better than the previous unsupervised method [9] and comparable to the supervised method PSMNet. Qualitative results are displayed in Fig. 6. Then, we test the performance of the original supervised models of the MC-CNN and PSMNet trained on the KITTI dataset. The errors of their outputs are high, which illustrates that the generalization ability of these supervised methods is not satisfactory.

2) KITTI 2015: To compare our method with other unsupervised or self-supervised approaches, we evaluate the trained model on the KITTI 2015 dataset [1], which includes 200 training stereo images pairs and 200 test pairs. The ground truth for the test set is withheld for evaluation. From the quantitative results shown in Table II, our method



Color image Our model PSMNet

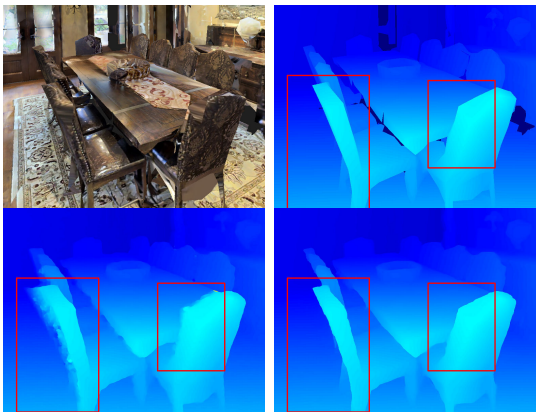
Fig. 8. Test on real images in our dataset. The output of our model and the supervised method PSMNet are presented.

outperforms all previous unsupervised learning methods by a large margin even without fine-tuning on the KITTI data. In other words, our model is not trained on outdoor images but can perform well on outdoor images in the KITTI dataset. We believe our model could achieve a lower error if trained on KITTI-style multiscopic images. The D1-all error of Godard [9] trained with our synthetic images is 11.36% while its error reduces to 9.194% after the fine-tuning on KITTI images. Also, the second-best performing method [10] utilizes two consecutive stereo pairs while we only use one pair.

Some qualitative results are displayed in Fig. 7 for visualization. The results demonstrate the generalization ability of our method, which benefits from self-supervised learning. In self-supervised learning, since there is no ground truth depth information, there is no chance to overfit the depth value. Instead, the network needs to find the inner pattern and geometry shape underneath.

We also compare our results with the supervised method PSMNet when trained on our synthetic dataset without fine-tuning on the KITTI images, denoted as "PSMNet*". When the model is trained with KITTI data, the D1 error on KITTI 2015 can reach 1.83%. However, the error is 5.90% when trained on our synthetic training set. This means if not trained on the same dataset, the performance of this supervised method does not have advantages over our self-supervised approach. Generalization between different datasets suffers from a large performance decrease, which is the drawback of supervised learning methods.

3) Real-world Data and Online Fine-tuning: To further inspect the generalization ability of our method, we test the network on the images captured by the real cameras in our dataset. The center-view color image and the output disparity map between the center view and the right view are shown in Fig. 8. From the qualitative results, we can see the network still performs stably and obtains better disparity maps than the supervised method PSMNet after being trained on our synthetic dataset. This indicates that after the training, our self-supervised model can perform more stably in practical applications. Another advantage of the self-supervised method is that online fine-tuning becomes feasible. Therefore, we fine-tune the network using the newly captured images when being employed in the real world. The results show the estimated disparity maps are further improved, which are presented in supplementary materials. This



Stereo Multiscopic

Fig. 9. Comparison between stereo mode and our multiscopic mode. The first row displays the center view image ground-truth disparity. The second row displays the disparity map obtained by self-supervised learning using 2 images and that using 3 images. Occluded areas are much better with our multiscopic approach.

Image	$\mathcal{L}_P$	$\mathcal{L}_M$	D1-all	D1-occ
2	w/o cross	✗	7.43%	91.66%
3	w/o cross	✗	7.40%	91.54%
3	Cross	✗	5.05%	38.71%
3	Cross	w/o uncertainty	4.32%	15.29%
3	Cross	uncertainty-aware	3.82%	10.87%

TABLE III. Ablation study on KITTI 2015. "w/o cross" means only using $\mathcal{L}_P(I^l_{cl}, I_c)$, $\mathcal{L}_P(I^r_{cr}, I_c)$. "w/o uncertainty" means only using the first equation of Equ. 6 for $\mathcal{L}_M$. D1-occ is the D1 error of occluded pixels.

demonstrates the practical usefulness of our self-supervised multiscopic framework.

C. Ablation Study

To inspect the effect of our framework, we evaluate each module of our model on the KITTI 2015 training set, as displayed in Table III. For the stereo self-supervised mode, we use only two images where the photometric loss is built by warping between the center image and the right image. Then we add modules proposed in our framework one by one. The result shows that a better disparity map can be obtained with our cross-photometric loss as well as the uncertainty-aware mutual-supervision loss, especially in occluded areas. A qualitative comparison is shown in Fig. 9, from where we can see the occluded parts in stereo mode cannot be reconstructed correctly but are estimated much better in multiscopic mode. This is as expected. The loss $\mathcal{L}_P$ and $\mathcal{L}_M$ in multiscopic mode can both assist the estimation in occluded areas since they link the left and the right view, e.g., the invisible pixel in the left view would cause a large loss in $\mathcal{L}_M$ and $\mathcal{L}_P(I^l_{cr}, I_c)$.

VI. Conclusion

In this paper, we exploit the image shape of multiscopic images to learn an end-to-end network to estimate the disparity map without ground-truth depth information. In a multiscopic structure, the images are captured with parallel,

co-planar, and same-parallax camera angles, such that the pixel disparities are the same between the center image and any surrounding image. Self-supervision is obtained by taking advantage of this constrain. The cross photometric loss, uncertainty-aware mutual-supervision loss, and smoothness loss are introduced to guide self-supervised learning. As there is no public large multiscopic dataset, we build a new dataset composed of synthetic images rendered by 3D engines and real-world images captured by the multiscopic systems designed by ourselves. After being trained with only these indoor synthetic images, our network can obtain better disparity maps in the unseen outdoor KITTI dataset compared to previous unsupervised methods.

REFERENCES

[1] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *CVPR*, 2012.

[2] Y. Yao, Z. Luo, S. Li, T. Fang, and L. Quan, "Mvsnet: Depth inference for unstructured multi-view stereo," in *ECCV*, 2018.

[3] W. Yuan, K. Hang, H. Song, D. Kragic, M. Y. Wang, and J. A. Stork, "Reinforcement learning in topology-based representation for human body movement with whole arm manipulation," in *ICRA*, 2019.

[4] W. Yuan, K. Hang, D. Kragic, M. Y. Wang, and J. A. Stork, "End-to-end nonprehensile rearrangement with deep reinforcement learning and simulation-to-reality transfer," *RAS*, vol. 119, pp. 119–134, 2019.

[5] A. Kendall, H. Martirosyan, S. Dasgupta, P. Henry, R. Kennedy, A. Bachrach, and A. Bry, "End-to-end learning of geometry and context for deep stereo regression," in *ICCV*, 2017.

[6] J.-R. Chang and Y.-S. Chen, "Pyramid stereo matching network," in *CVPR*, 2018.

[7] F. Zhang, V. Prisacariu, R. Yang, and P. H. Torr, "Ga-net: Guided aggregation net for end-to-end stereo matching," in *CVPR*, 2019.

[8] J. Xie, C. Lei, Z. Li, L. E. Li, and Q. Chen, "Video depth estimation by fusing flow-to-depth proposals," in *IROS*, 2020.

[9] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *CVPR*, 2017.

[10] Y. Wang, P. Wang, Z. Yang, C. Luo, Y. Yang, and W. Xu, "Unos: Unified unsupervised optical-flow and stereo-depth estimation by watching videos," in *CVPR*, 2019.

[11] A. Tonioni, O. Rahnama, T. Joy, L. D. Stefano, T. Ajanthan, and P. H. Torr, "Learning to adapt for stereo," in *CVPR*, 2019.

[12] A. Tonioni, F. Tosi, M. Poggi, S. Mattoccia, and L. D. Stefano, "Real-time self-adaptive deep stereo," in *CVPR*, 2019.

[13] Y. Zhong, Y. Dai, and H. Li, "Self-supervised learning for stereo matching with self-improving ability," *arXiv:1709.00930*, 2017.

[14] P. Liu, I. King, M. R. Lyu, and J. Xu, "Flow2stereo: Effective self-supervised learning of optical flow and stereo matching," in *CVPR*, 2020.

[15] W. Yuan, R. Fan, M. Y. Wang, and Q. Chen, "Mfusenet: Robust depth estimation with learned multiscopic fusion," *RA-L*, vol. 5, no. 2, pp. 3113–3120, 2020.

[16] J. McCormac, A. Handa, S. Leutenegger, and A. J. Davison, "Scenenet rgb-d: Can 5m synthetic images beat generic imagenet pre-training on indoor segmentation?" in *ICCV*, 2017.

[17] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niessner, M. Savva, S. Song, A. Zeng, and Y. Zhang, "Matterport3d: Learning from rgb-d data in indoor environments," *3DV*, 2017.

[18] V. Kolmogorov and R. Zabih, "Multi-camera scene reconstruction via graph cuts," in *ECCV*, 2002.

[19] M. Maitre, Y. Shinagawa, and M. N. Do, "Symmetric multi-view stereo reconstruction from planar camera arrays," in *CVPR*, 2008.

[20] Y. Wei and L. Quan, "Asymmetrical occlusion handling using graph cut for multi-view stereo," in *CVPR*, 2005.

[21] Z. Lee and T. Q. Nguyen, "Multi-array camera disparity enhancement," *TOM*, vol. 16, no. 8, pp. 2168–2177, 2014.

[22] Y. Nakamura, T. Matsuura, K. Satoh, and Y. Ohta, "Occlusion detectable stereo-occlusion patterns in camera matrix," in *CVPR*, 1996.

[23] J. Zbontar and Y. LeCun, "Stereo matching by training a convolutional neural network to compare image patches." *JMLR*, vol. 17, no. 1-32, p. 2, 2016.

[24] R. Garg, V. K. BG, G. Carneiro, and I. Reid, "Unsupervised cnn for single view depth estimation: Geometry to the rescue," in *ECCV*, 2016.

[25] J. Xie, R. Girshick, and A. Farhadi, "Deep3d: Fully automatic 2d-to-3d video conversion with deep convolutional neural networks," in *ECCV*, 2016.

[26] X. Guo, H. Li, S. Yi, J. Ren, and X. Wang, "Learning monocular depth by distilling cross-domain stereo networks," in *ECCV*, 2018.

[27] C. Zhou, H. Zhang, X. Shen, and J. Jia, "Unsupervised learning of stereo matching," in *ICCV*, 2017.

[28] Z. Yin and J. Shi, "Geonet: Unsupervised learning of dense depth, optical flow and camera pose," in *CVPR*, 2018.

[29] N. Luo, C. Yang, W. Sun, and B. Song, "Unsupervised stereo matching with occlusion-aware loss," in *Pacific Rim ICAI*, 2018.

[30] Y. Zhong, H. Li, and Y. Dai, "Open-world stereo video matching with deep rnn," in *ECCV*, 2018.

[31] T. Kim, K. Ryu, K. Song, and K.-J. Yoon, "Loop-net: Joint unsupervised disparity and optical flow estimation of stereo videos with spatiotemporal loop consistency," *RA-L*, vol. 5, no. 4, pp. 5597–5604, 2020.

[32] L. Wang, Y. Guo, Y. Wang, Z. Liang, Z. Lin, J. Yang, and W. An, "Parallax attention for unsupervised stereo correspondence learning," *TPAMI*, 2020.

[33] F. Aleotti, F. Tosi, L. Zhang, M. Poggi, and S. Mattoccia, "Reversing the cycle: self-supervised deep stereo through enhanced monocular distillation," *arXiv:2008.07130*, 2020.

[34] K. Zhang, J. Xie, N. Snavely, and Q. Chen, "Depth sensing beyond lidar range," in *CVPR*, 2020.

[35] W. Yuan, R. Fan, M. Y. Wang, and Q. Chen, "Active perception with a monocular camera for multiscopic vision," *arXiv:2001.08212*, 2020.

[36] X. Li, Y. Hu, X. Gao, D. Tao, and B. Ning, "A multi-frame image super-resolution method," *Signal Processing*, vol. 90, no. 2, pp. 405–414, 2010.

[37] T. Zhou, R. Tucker, J. Flynn, G. Fyffe, and N. Snavely, "Stereo magnification: learning view synthesis using multiplane images," *TOG*, vol. 37, no. 4, pp. 65:1–65:12, 2018.

[38] M. Savva *et al.*, "Habitat: A platform for embodied ai research," in *ICCV*, 2019.

[39] F. Xia, A. R. Zamir, Z.-Y. He, A. Sax, J. Malik, and S. Savarese, "Gibson env: real-world perception for embodied agents," in *CVPR*, 2018.

[40] A. X. Chang *et al.*, "Shapenet: An information-rich 3d model repository," *arXiv:1512.03012*, 2015.

[41] Z. Wang *et al.*, "Image quality assessment: from error visibility to structural similarity," *TIP*, vol. 13, no. 4, pp. 600–612, 2004.

[42] A. Kendall and Y. Gal, "What uncertainties do we need in bayesian deep learning for computer vision?" in *NIPS*, 2017.

[43] A. Li and Z. Yuan, "Occlusion aware stereo matching via cooperative unsupervised learning," in *ACCV*, 2018.