SynDeMo: Synergistic Deep Feature Alignment for Joint Learning of Depth and Ego-Motion

Behzad Bozorgtabar[1] Mohammad Saeed Rad[1] Dwarikanath Mahapatra
Jean-Philippe Thiran[1]
[1]École Polytechnique Fédérale de Lausanne (EPFL)
behzad.bozorgtabar@epfl.ch saeed.rad@epfl.ch dmahapatra@gmail.com jean-philippe.thiran@epfl.ch

Abstract

Despite well-established baselines, learning of scene depth and ego-motion from monocular video remains an ongoing challenge, specifically when handling scaling ambiguity issues and depth inconsistencies in image sequences. Much prior work uses either a supervised mode of learning or stereo images. The former is limited by the amount of labeled data, as it requires expensive sensors, while the latter is not always readily available as monocular sequences. In this work, we demonstrate the benefit of using geometric information from synthetic images, coupled with scene depth information, to recover the scale in depth and ego-motion estimation from monocular videos. We developed our framework using synthetic image-depth pairs and unlabeled real monocular images. We had three training objectives: first, to use deep feature alignment to reduce the domain gap between synthetic and monocular images to yield more accurate depth estimation when presented with only real monocular images at test time. Second, we learn scene specific representation by exploiting self-supervision coming from multi-view synthetic images without the need for depth labels. Third, our method uses single-view depth and pose networks, which are capable of jointly training and supervising one another mutually, yielding consistent depth and ego-motion estimates. Extensive experiments demonstrate that our depth and ego-motion models surpass the state-of-the-art, unsupervised methods and compare favorably to early supervised deep models for geometric understanding. We validate the effectiveness of our training objectives against standard benchmarks thorough an ablation study.

1. Introduction

Depth sensing and ego-motion estimation from image and videos play an important role in 3D scene geometry understanding and are widely applicable to many real-world domains, such as navigation systems [9, 7], video analysis [39] and autonomous driving platforms [4]. Although deep models using a supervised mode of learning [38, 29, 8, 30] have been widely used for understanding a scene geometry, it is typically difficult and expensive to obtain geometry-related labels in practice. Due to this limitation, unsupervised methods [13, 49, 16, 25] have been focused on estimating scene depth and camera motion, which have been formulated as self-supervised learning using photometric warp error. However, learning the entangled information of the depth and ego-motion for a monocular video suffers from the per frame scale confusion. This limitation is compounded by the fact that these methods do not generalize well to new datasets and scenes. Some recent methods [46, 32, 49, 27] have incorporated stereo image pairs in the training stage to address this issue. Nonetheless, stereo im-

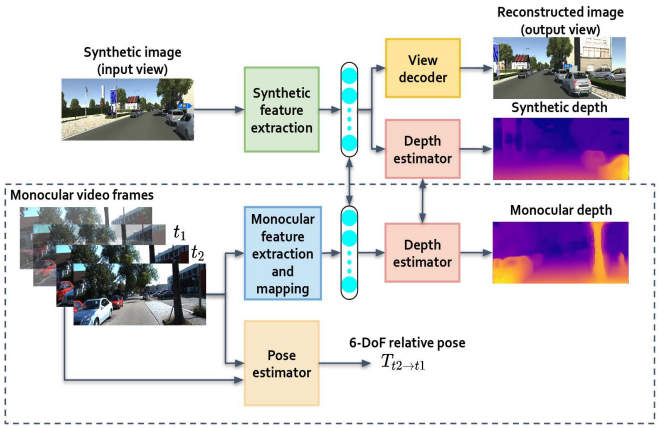

Figure 1. System overview of the proposed SynDeMo. It consists of two data streams, the upper stream takes multi-view synthetic image-depth pairs as input, while the lower stream takes unlabeled monocular frames as input during training. At test time, we use only the lower stream within the dashed line. SynDeMo can simultaneously estimate ego-motion and depth using only monocular video by resolving the scale ambiguity.

ages are not as readily available as monocular video, which hinders their wider application.

Learning from synthetic data can mitigate the issue. In particular, recent advances in computer graphics [11] have made it possible to generate a large set of synthetic 3D scenes, from which we can render multiple synthetic images from different viewpoints and their corresponding depth maps. However, learning from synthetic images can be problematic due to the distribution discrepancy between them and real monocular images. In this paper, we propose a solution to exploit geometry information from synthetic images, when trained jointly with unlabeled monocular images (see Fig. 1).

Contributions. Our contributions are as follows:

- We aim to exploit the heterogeneous synthetic images with a large diversity of urban scenes and unlabeled real monocular images to improve our depth estimator when presented with only real monocular images at test time. To do so, we use a synergistic deep feature alignment to minimize the distance between the inner feature representations for both real and synthetic images;
- We train a view decoder for the synthetic images to synthesize an image seen from one viewpoint from an image rendered from a different view. This helps to capture 3D scene structure in the latent space. When this latent representation space is aligned with latent space of the real images, it can help to improve predictions for real monocular images by retaining the scene geometry present in the synthetic images;
- The more accurate depth map estimation from our SynDeMo helps to constrain the output of the pose estimator through the temporal self-supervised loss, recovering the scale and consistent geometry predictions.

2. Related Work

We consider prior work within two distinct domains: deep models for scene geometry understanding (Sec. 2.1) and cross-domain adaptation methods (Sec. 2.2).

2.1. Deep Models for Geometry Understanding

The deep learning methods using a supervised mode of learning for scene geometry [5, 10, 29, 44, 8, 30, 45], have demonstrated great performance against the traditional Structure from Motion (SfM) approaches. However, these supervised methods have been trained on large-scale labeled datasets, which limits their applicability in a real scenario.

To address this problem, several unsupervised learning based methods [52, 48, 3, 33, 13, 51, 16] have been proposed. Initial methods [13, 16] focus only on depth estimation using the left-right photometric constraint between a pair of stereo images. Built upon this idea, some other methods [51, 33] use the geometric constraints between a pair of temporal images in monocular video to jointly learn depth and ego-motion. However, these approaches suffer from scale ambiguity. More recently, some methods [49, 27] have made use of stereo images, formulating both spatial and temporal geometric constraints in their learning framework, to solve the ambiguity. Nonetheless, stereo images are not as widely available as monocular video.

2.2. Cross-Domain Adaptation

Our proposed SynDeMo is related to cross-domain learning methods. Most of these frameworks [37, 50, 12, 36, 40] aim to minimize the distance between the feature or image distributions of different domains in order to mitigate the performance drop caused by the domain gap. Using knowledge distillation, Guo et al. [19] presented a deep stereo matching network as a proxy to learn depth from synthetic data. Zheng et al. [50] proposed T^2Net for single-image depth estimation. It uses an image translation network to improve the realism of synthetic images, followed by a depth estimator. Atapour et al. [1] proposed a learning approach based on style transfer and adversarial training to adapt a depth estimation model trained on synthetic images into real images. However, their image translation network introduces undesirable distortions, which degrades the performance of successive depth estimator and generalizes poorly to unseen data. To overcome this issue, we propose a learning framework to align the feature distributions of real and synthetic images without affecting source images. Moreover, we learn a geometry-aware representation from multi-view synthetic images to reduce annotated synthetic images.

3. Method

In this section, we first present our baseline, and then proceed with the loss terms used in SynDeMo.

3.1. Algorithm Baseline

The backbone of our method for learning a scene's geometry is based on a temporal self-supervision signal, which comes from the task of image reconstruction for two nearby temporal views. Our network in Fig. 2 consists of two data streams, one for real unlabeled video, while the other is for synthetic images. As in [51], the real data stream includes two sub-networks, a depth CNN for a single-view depth prediction and a pose CNN for the ego-motion estimation. The sub-networks jointly supervise each other during training. Given a temporal pair of video frames

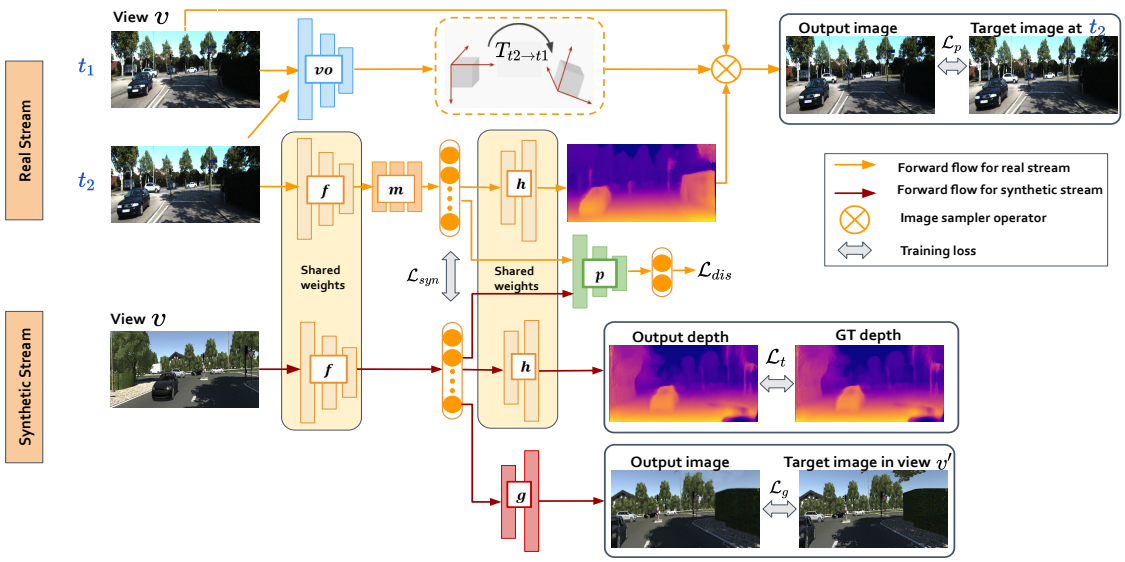

Figure 2. Schematic diagram of the proposed SynDeMo during training. At test time, we use only the real data stream, which includes two sub-networks that can be used independently, one for single-view depth estimation, and one for ego-motion estimation.

$(I_{t1}, I_{t2}) \in \mathbb{R}^{H\times W\times 3}$, the outputs of both networks are used to inverse warp the source view I_{t1} to synthesize an adjacent target image in the sequence I_{t2}. Knowing the camera intrinsic matrix $K \in \mathbb{R}^{3\times 3}$ (for simplicity, we assume K of all the views to be identical), the image synthesis task can be formulated by,

$$I_{v,t1\to t2} = I_{v,t1}\left[proj\left(D_{v,t2}, T_{t2\to t1}, K\right)\right] \tag{1}$$

where $T_{t2\to t1}$ is the relative pose for the source view $I_{v,t1}$, with respect to the target image $I_{v,t2}$'s pose taken from a given viewpoint v. $D_{v,t2}$ is the target view's depth estimation from a viewpoint v. We only use a single image view and for the sake of simplicity, the index of a viewpoint v is used. $proj$ is the resulting pixel coordinates of the projected depth $D_{v,t2}$ in $I_{v,t1}$ and $[\cdot]$ is the image sampler operator from the spatial transformer network (STN) [22], which is sub-differentiable.

Existing self-supervised methods [13, 51, 16] used the average photometric error across two temporal adjacent views to estimate the depth as the intermediary variable. However, using average photometric error results in artifacts in the presence of occluded regions, which are visible in one image but not the other. Here, per-pixel minimum of source image is used to enhance the occlusion boundaries, as proved by [17]. We use a combination of SSIM [43] and l_1 as a photometric error loss function ρ in the training process,

$$\mathcal{L}_p = \min_{t1} \rho\left(I_{v,t2}, I_{v,t1\to t2}\right),$$
$$\rho\left(I_a, I_b\right) = \alpha\frac{1 - SSIM\left(I_a, I_b\right)}{2} + (1-\alpha)\left\|I_a - I_b\right\|_1 \tag{2}$$

In addition, following [21, 16], to regularize the depth estimates in texture-less image regions, we incorporate the edge-aware smoothness loss that takes the gradients of the corresponding image into account,

$$\mathcal{L}_s = \sum_{i,j}^{W,H} \left|\partial_x d_{i,j}\right| e^{-\left|\partial_x I_{i,j}\right|} + \left|\partial_y d_{i,j}\right| e^{-\left|\partial_y I_{i,j}\right|} \tag{3}$$

where $d_{i,j}$ is the inverse depth map. $\partial_x(\cdot)$ and $\partial_y(\cdot)$ denote gradients in horizontal and vertical direction, respectively.

Scale Ambiguity. A common issue in estimating depth and pose from monocular video, as pointed out in previous work [49, 46, 3], is scale ambiguity. In the absence of other source of information, the camera translations' absolute scaling between any two adjacent frames is not fully known. In addition, any two disparity values differing only in scale are equivalent in the projective space. Thus, photometric warp error used for image reconstruction is scale-invariant. We exploit the additional geometric information from the synthetic images and their corresponding depth maps to improve both depth and ego-motion estimation for single-view monocular video.

3.2. Learning from Synthetic Data

Our method builds on the observation that using a new auxiliary supervision signal, we can learn scene geometry from synthetic imagery and learn to align synthetic images with real monocular images. The use of synthetic images rendered from different views enables a network to learn a geometry-aware representation of multi-view images, and

constrains the camera motion and scene depth. Inference (i.e. ego-motion estimation and depth) without any scale confusion is then possible using only monocular video.

Our training stage consists of two steps. First, we pretrain the depth estimation part (modules f and h) of the network, in Fig. 2 with synthetic data stream, only. Second, to adapt the learned depth estimator to real data with no corresponding ground-truth depth maps, we propose learning the whole model jointly with both data streams. This step includes two auxiliary tasks. (1) The first task is to reduce the domain discrepancy between unlabeled real images and synthetic images. (2) The second task is to retain the scene geometry present in the synthetic input images. Finally, for inference at test time, we only use single-view real data stream of the network that takes monocular video sequences as input and uses depth and camera motion estimators to estimate depth and ego-motion, respectively. Below, we describe the loss terms used for training our framework.

Task Loss. Our depth estimator is based on an encoder-decoder architecture. The feature extraction module f is a convolutional neural network that maps an input image to its latent representation. The decoder module h is a sub-pixel convolutional neural network that generates a depth map of the input image when giving its latent representation. The synthetic data stream takes synthetic images as input, while the real data stream takes monocular color images as input and uses the mapping network m to map the real images to the synthetic images in the latent space. The parameters of the feature extractor f and depth decoder h are shared between two data streams. Since we only have access to the ground truth depth maps of synthetic images during training, we train the depth estimator part of the network to measure per-pixel difference between the predicted depth map $\hat{D} = h\left(f\left(I^s\right)\right)$ and the synthetic (ground truth) depth map as our task loss,

$$\mathcal{L}_t = \sum_k \left\| \hat{D}_k - D_k \right\|_1 \tag{4}$$

where D_k is the ground truth and $\hat{D}_k$ is the depth prediction for the k^{th} synthetic image I^s. k ranges over all the synthetic images of a sequence taken from all viewpoints.

Spatial Self-Supervised Loss. We propose using synthetic images rendered from different views to capture 3D geometry-aware latent space, without requiring any depth annotation. We train an image view decoder g to predict the appearance of a synthetic image $I^s_{v'}$ rendered from view v' given a synthetic image I^s_v rendered from view v by enforcing spatial consistency between them. The crux of this self-supervised learning is that if the decoder is able to reconstruct another view of the image solely from the latent representation, the latent representation must capture scene geometry information. When this latent representation space is aligned with the latent space of real data, it can help to improve geometry predictions by constraining the sub-networks used for real data. Doing so, given a geometry-aware feature representation of the synthetic image $z = f\left(I^s_v\right)$ rendered from view v, we learn a view decoder g to reconstruct the image's appearance $\hat{I}^s_{v'} = g\left(f\left(I^s_v\right)\right)$ of view v'. We employ a spatial self-supervised loss,

$$\mathcal{L}_g = \sum_k \left\| I^s_{v',k} - \hat{I}^s_{v',k} \right\|_1 \tag{5}$$

where $I^s_{v',k}$ is the synthetic image rendered from view v' and $\hat{I}^s_{v',k}$ is the model prediction for the k^{th} image of view v'. However, using self-supervised loss for synthetic data alone cannot guarantee that the joint latent representation of real and synthetic data retains geometry information as the feature distributions of these two domains can split apart. Below, we will show how we address this issue.

Synergistic Loss. To apply the depth estimator to real images, we train a mapping network m to map the features extracted from real images to the features extracted from synthetic images. However, it is difficult to obtain many one-to-one corresponding real and synthetic images to mimic the discrepancy between real and synthetic data. To address this, we apply the GAN loss [18] to the shared latent representation space to bridge the gap between transformed features of real images and features of synthetic images. As a consequence, images with similar pose and geometry are embedded to similar latent space, apart from whether the images are real or synthetic.

We added a discriminator p, which is trained to identify the real and fake images using the latent representation. The feature mapping function m has to play two roles: fooling the discriminator by making the latent representation of real images indistinguishable from the latent representation of synthetic images; and minimizing the distance between these two feature representations directly in the presence of one-to-one correspondence between real and synthetic images. We substituted the vanilla formulation of the GAN with a least-squares loss GAN [34], which has proven to be more stable during learning. As input, the discriminator function p takes a joint latent representation z and outputs a scalar $\hat{y} = p\left(z\right)$ between $[0, 1]$, which should be $y_r = 1$ for real and $y_s = 0$ for synthetic images. Thus, the discriminator is learned to minimize deviations from target values for predictions on synthetic and real images,

$$\mathcal{L}_{dis} = \sum_{k \in \mathcal{R}} \left(\hat{y}_k - y_r\right)^2 + \sum_{k \in \mathcal{S}} \left(\hat{y}_k - y_s\right)^2 \tag{6}$$

where $\mathcal{R}$ and $\mathcal{S}$ denote the set of real and synthetic images, respectively. Eventually the mapping function m is trained to meet the requirements as discussed above. We define our synergistic loss $\mathcal{L}_{syn}$,

$$\mathcal{L}_{syn} = \sum_{k \in \mathcal{C}} \| f\left(I_k^s\right) - m\left(f\left(I_k^r\right)\right) \|_2^2 + \lambda_r \sum_{k \in \mathcal{R}} \left(\hat{y}_k - y_s\right)^2 \tag{7}$$

where $\mathcal{C}$ is the set of available corresponding real and synthetic images and I_k^s and I_k^r are the k^{th} synthetic and real image, respectively. λ_r is the trade-off weight to balance GAN related loss term.

Overall Objective. Finally, the proposed training loss $\mathcal{L}$ joins the synergistic loss $\mathcal{L}_{syn}$ to align feature distribution of real and synthetic images, task loss $\mathcal{L}_t$ for synthetic data only, spatial self-supervised loss $\mathcal{L}_g$ to leverage multi-view synthetic images, smoothness loss term $\mathcal{L}_s$ and the photometric loss $\mathcal{L}_p$,

$$\mathcal{L} = \mathcal{L}_{syn} + \lambda_t \mathcal{L}_t + \lambda_g \mathcal{L}_g + \lambda_s \mathcal{L}_s + \lambda_p \mathcal{L}_p \tag{8}$$

where λ_t, λ_g, λ_s and λ_p are weighting terms.

4. Experiments

In this section, we verify the effectiveness of our SynDeMo and its components in isolation through extensive experiments. More results are given in the supplementary material.

4.1. Datasets

Synthetic dataset for pre-training. We pre-train the depth estimation part (modules f and h) of the network using Virtual KITTI (vKITTI) [11], a synthetic dataset, which serves as a proxy to the real KITTI dataset [14]. This dataset contains 21,260 image-depth paired frames generated under different environmental conditions.

Datasets for joint training. After the pre-training stage, we train the whole model (Eq. 8) jointly with real and synthetic data. For unlabeled real data, we use the KITTI dataset [14] as the main training and validation dataset. We only use a single-view monocular video and follow the same training protocol as in [51]; about 40k frames for training and 4k for validation. In addition, for a fair comparison with state-of-the-art methods [52, 48], we also use the CityScapes dataset [6] with the same training setting as in [51] for training our model including both depth and pose sub-networks.

4.2. Implementation Details

In the following subsection, we provide more detail about the network architecture and experimental setup used for training.

Architecture. We relied on successful architecture in [51] for our pose CNN (sub-network vo). The pose CNN receives monocular temporal frames concatenated along their color channels and then outputs the 6-DoF ego-motion. Our depth CNN is based on encoder-decoder architecture with skip-connections. For an encoder (module f), we use the ResNet50-1by2 as a variant of ResNet50 [20]. Our depth decoder, h, consists of sub-pixel convolution layers followed by instance normalization [2]. The parameters of the modules f and h are shared between two streams. For both view decoder g and depth decoder h, we use the same architecture except for the last prediction layer. We feed the last features from the depth encoder f into the mapping module m, which is a fully-connected network with two residual blocks. Each block consists of two hidden layers of dimension 512 with a ReLU activation function in between. The discriminator p has the same architecture as the mapping module m with an additional linear layer to return a single output.

Details for experimental setup. We pre-train the depth estimation part using the vKITTI dataset for about 150k iterations and subsequently train the whole network jointly with real monocular frames and synthetic images for 130k iterations. We use Adam optimizer [24] ($\beta_1 = 0.5, \beta_2 = 0.999$) with a base learning rate of 2e-4. We train our network with a mini-batch size of 6. For each mini-batch, we independently sample a set of synthetic images and a set of temporal consecutive image pairs from a video. We experimentally found the weights of different loss components in Eq. 7 and Eq. 8 and set $\lambda_r = 0.01$, $\lambda_t = 0.05$, $\lambda_g = 0.1$, $\lambda_s = 0.01$ and $\lambda_p = 0.5$. We also set $\alpha = 0.85$ in Eq. 2. The input images are resized to 608×160. Our depth CNN outputs inverse depth and we convert the predicted inverse depth to depth map for evaluation. Following [48], we clipped our depth maps from within 0.001m to 80m. To reduce our reliance on the corresponding set of images $\mathcal{C}$ between domains, we only use images of sequence 02 (road in urban area) from vKITTI.

4.3. Experimental Evaluation

Single-view depth estimation. We conduct quantitative comparisons with state-of-the-art methods for single-view depth estimation task using KITTI test split [8]. We use evaluation protocol adopted by prior work [51]. The ground truth depth maps were produced by projecting 3D Velodyne LiDAR points on to the image plane. Only pixels with ground truth depth values are used for evaluation. In Table 1 we show that our full SynDeMo model surpasses the state-of-the-art unsupervised methods, which are substantially trained on the KITTI dataset. Our model also achieves competitive performance with respect to supervised learning methods [25, 45] trained either with stereo

image pairs or with real depth ground truth. Qualitative results are shown in Fig. 3. Our SynDeMo can produce pleasing results with fine structures.

Odometry estimation. We used the official KITTI visual odometry split [15] to evaluate the performance of our method. For a fair comparison with the baselines, we followed the odometry split and evaluation protocol as in [51] to train and evaluate our model. Following [51], we measured the Absolute Trajectory Error (ATE) [35] averaged over every 5-frame snippets as the performance metric. First, the result was compared to a well-established SLAM framework, ORB-SLAM [35], which involves global bundle adjustment and loop closure detection. Here we use two variants of monocular ORB-SLAM: "The ORB- SLAM (short)", which runs on 5-frame snippets, and "ORB- SLAM (full)", which uses the whole sequence to recover odometry. As shown in Table 2, our method surpasses the state-of-the-art unsupervised learning methods on two test sequences. It also shows superior performance with respect to ORB-SLAM, even though our method uses a rather short sequence. This reveals that our method takes advantage of geometric information from synthetic data that allows us to improve single-view depth estimation, and consequently constrains the output of pose CNN, yielding to recover scale-consistent odometry using only a single-view video at test time.

The qualitative odometry results for the KITTI odometry testing sequence 09 are shown in Fig. 4. The results are compared with the powerful stereo based odometry learning method, deep feature reconstruction network (DFR-Net) [49] and ORB-SLAM [35] (with and without loop closure). We post-process the scale for the ORB-SLAM method as it suffers from a scale ambiguity. The estimated trajectories of our method without any post processing are qualitatively closest to the ground truth among all the methods. These qualitative results align well with the quantitative numbers in Table 2. We also report the average translational error and average rotational error for the complete test sequences in the supplementary material.

4.4. Ablation Study

We investigate the effectiveness of our contributions by comparing our full model with the baselines based on the same experimental setting. For each ablation experiment, we turn-off some of the loss terms in the final objective function and then generate the related results for evaluation. Our baselines are:

- **Baseline**: Baseline model (Sec. 3.1) trained with a temporal photometric loss and a smoothness loss ($\mathcal{L}_p + \mathcal{L}_s$) only on monocular sequences;
- **SynDeMo (Real&Synth$^{\ominus}$\| Feat Align+View)**: SynDeMo trained jointly with unlabeled synthetic and real images by adding multi-view self-supervised loss and synergistic loss to the first baseline ($\mathcal{L}_p + \mathcal{L}_s + \mathcal{L}_g + \mathcal{L}_{syn}$) ;
- **SynDeMo (Real&Synth. | Feat Align)**: SynDeMo trained using labeled synthetic images and real monocular images by adding the task loss and the synergistic loss to the first baseline ($\mathcal{L}_p + \mathcal{L}_s + \mathcal{L}_t + \mathcal{L}_{syn}$);
- **SynDeMo (Real&Synth. | View)**: SynDeMo trained using both data streams by adding only multi-view self-supervised loss for labeled synthetic images to the first baseline ($\mathcal{L}_p + \mathcal{L}_s + \mathcal{L}_t + \mathcal{L}_g$);
- **SynDeMo (Real&Synth. | Full)**: SynDeMo trained with the full training objectives.

Scene geometry to solve motion and scale confusion. Our SynDeMo trained with each of the proposed loss terms, resulting in a notable performance gain in both depth (Table 1) and odometry estimation (Table 2) compared to our **Baseline**, which is trained solely with temporal photometric loss and smoothness loss on real monocular images. In particular, our full model trained on KITTI resulted in an improvement over all depth evaluation metrics, e.g., a gain of about 14% in accuracy for $\delta < 1.25$, compared to our **Baseline**.

Another challenging scenario for monocular trained approaches is the moving object. In particular, in the "car following" scenario, when the car moves at the same speed as the camera, the car is usually projected into infinite depth (Fig. 5, bottom row), yielding small photometric error during training. Fig. 5 shows examples of dynamic scenes from the Cityscapes dataset, which contains many moving objects. Our monocular trained **Baseline** fails for moving objects. However, if we incorporate additional scene geometry information from synthetic data into our learning framework **(Real&Synth. | View)**, such motion confusion can be alleviated jointly with monocular videos. Finally, if we take advantage of both loss terms in our final model, **SynDeMo (Real&Synth. | Full)**, depth prediction quality further improves.

Domain independence. We also apply our fully trained model (without fine-tuning) to the Make3D dataset [38] to see how our SynDeMo generalizes over unseen data domains. Table 3 shows that our SynDeMo significantly outperforms all previous depth estimation methods, whether pre-trained in an unsupervised mode or with cross-domain synthetic data [1, 50] with corresponding depth ground truth. We also observe that using our full training objec-

Method	Dataset	Supervision	Error Metric ↓				Accuracy Metric ↑		
			Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Eigen et al. [8] (NeurIPS 2014)	K	Real Depth	0.203	1.548	6.307	0.246	0.702	0.890	0.958
Godard et al. [16] (CVPR 2017)	K	Stereo	0.133	1.140	5.527	0.229	0.830	0.936	0.970
Godard et al. [16] (CVPR 2017)	CS+K	Stereo	0.121	1.032	5.200	0.215	0.854	0.944	0.973
Zhan et al. [49] (CVPR 2018)	K	Stereo	0.144	1.391	5.869	0.241	0.803	0.928	0.969
Kuznietsov et al. [25] (CVPR 2017)	K	Stereo+Real Depth	0.113	0.741	4.621	0.189	0.862	**0.960**	**0.986**
Yang et al. [45] (ECCV 2018)	K	Stereo DSO [42]	**0.097**	**0.734**	**4.442**	0.187	**0.888**	0.958	0.980
Zhou et al. [51] (CVPR 2017)	K	Monocular	0.208	1.768	6.856	0.283	0.678	0.885	0.957
Mahjourian et al. [33] (CVPR 2018)	K	Monocular	0.163	1.240	6.220	0.250	0.762	0.916	0.968
Yang et al. [47] (CVPR 2018)	K	Monocular	0.162	1.352	6.276	0.252	-	-	-
Yin et al. [48] (CVPR 2018)	K	Monocular	0.155	1.296	5.857	0.233	0.793	0.931	0.973
Godard et al. [16] (CVPR 2017)	K	Monocular	0.154	1.218	5.699	0.231	0.798	0.932	0.973
Zou et al. [52] (ECCV 2018)	K	Monocular	0.150	1.124	5.507	0.223	0.806	0.933	0.973
Wang et al. [41] (CVPR 2018)	K	Monocular	0.151	1.257	5.583	0.228	0.810	0.936	0.974
Zheng et al. [50] (ECCV 2018)	K+vK	Monocular+Synthetic Depth	0.169	1.230	4.717	0.245	0.769	0.912	0.965
Baseline	K+vK	Monocular	0.196	1.695	6.454	0.273	0.719	0.903	0.964
SynDeMo (Real&Synth$^{\ominus}$| Feat Align+View)	K+vK	Monocular	0.145	1.401	5.873	0.248	0.801	0.925	0.967
SynDeMo (Real&Synth. | Feat Align)	K+vK	Monocular+Synthetic Depth	0.126	1.105	5.402	0.221	0.843	0.939	0.969
SynDeMo (Real&Synth. | View)	K+vK	Monocular+Synthetic Depth	0.123	1.056	5.298	0.219	0.850	0.942	0.971
SynDeMo (Real&Synth. | Full)	K+vK	Monocular+Synthetic Depth	0.116	0.746	4.627	0.194	0.858	0.952	0.977
Zhou et al. [51] (CVPR 2017)	CS+K	Monocular	0.198	1.836	6.565	0.275	0.718	0.901	0.960
Mahjourian et al. [33] (CVPR 2018)	CS+K	Monocular	0.159	1.231	5.912	0.243	0.784	0.923	0.970
Yang et al. [47] (CVPR 2018)	CS+K	Monocular	0.159	1.345	6.254	0.247	-	-	-
Yin et al. [48] (CVPR 2018)	CS+K	Monocular	0.153	1.328	5.737	0.232	0.802	0.934	0.972
Zou et al. [52] (ECCV 2018)	CS+K	Monocular	0.146	1.182	5.215	0.213	0.818	0.943	0.978
Wang et al. [41] (CVPR 2018)	CS+K	Monocular	0.148	1.187	5.496	0.226	0.812	0.938	0.975
Casser et al. [3] (AAAI 2019)	CS+K	Monocular	0.108	0.825	4.750	**0.186**	0.873	0.957	0.982
Baseline	CS+K+vK	Monocular	0.190	1.635	6.415	0.268	0.801	0.927	0.969
SynDeMo (Real&Synth$^{\ominus}$| Feat Align+View)	CS+K+vK	Monocular	0.143	1.379	5.721	0.239	0.806	0.932	0.971
SynDeMo (Real&Synth. | Feat Align)	CS+K+vK	Monocular+Synthetic Depth	0.121	1.039	5.272	0.216	0.855	0.946	0.973
SynDeMo (Real&Synth. | View)	CS+K+vK	Monocular+Synthetic Depth	0.113	0.743	4.623	0.189	0.861	0.954	0.979
SynDeMo (Real&Synth. | Full)	CS+K+vK	Monocular+Synthetic Depth	0.112	0.740	4.619	0.187	0.863	0.958	0.983

Table 1. Evaluation of depth estimation results for the KITTI test set [8]. For datasets used for training, K is the real KITTI dataset [14], CS is Cityscapes [6] and vK is the virtual KITTI dataset [11]. The overall best results are shown in **bold** and the best results within each block are underlined.

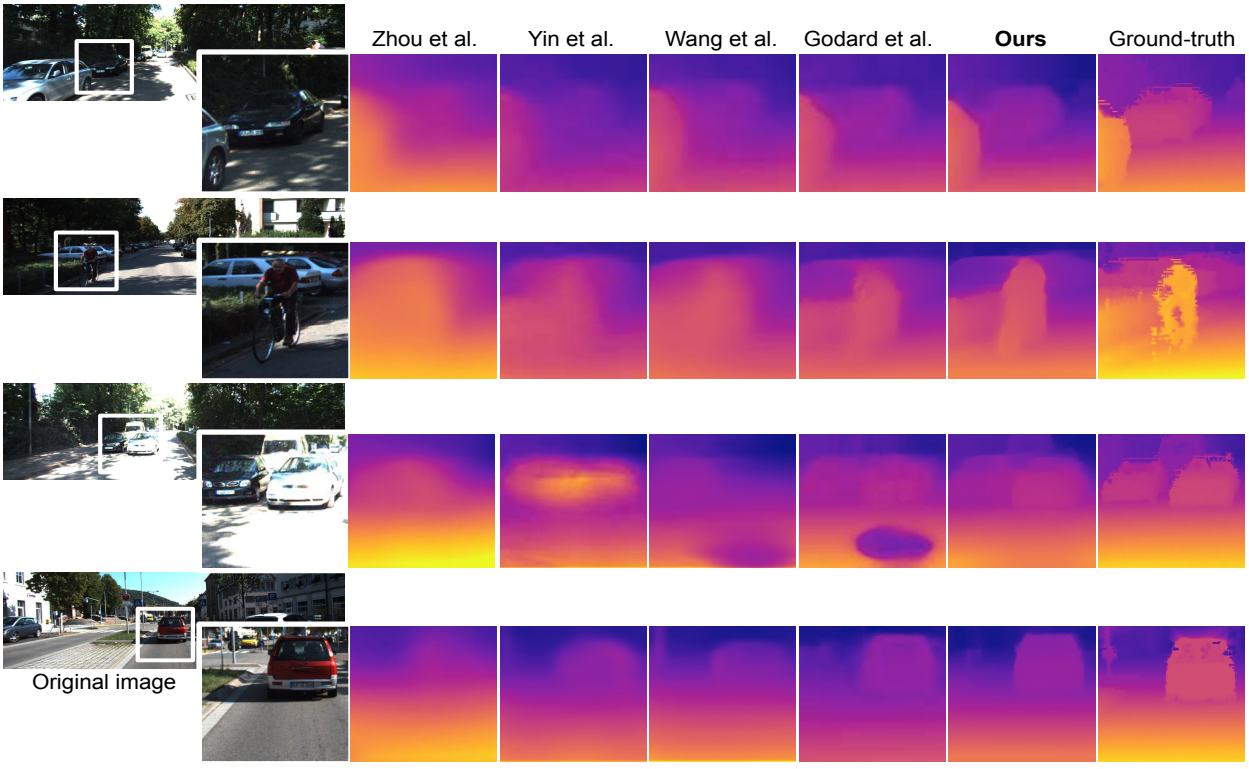

Figure 3. Qualitative comparison of different depth estimation methods for the KITTI dataset [8]. Best viewed in color.

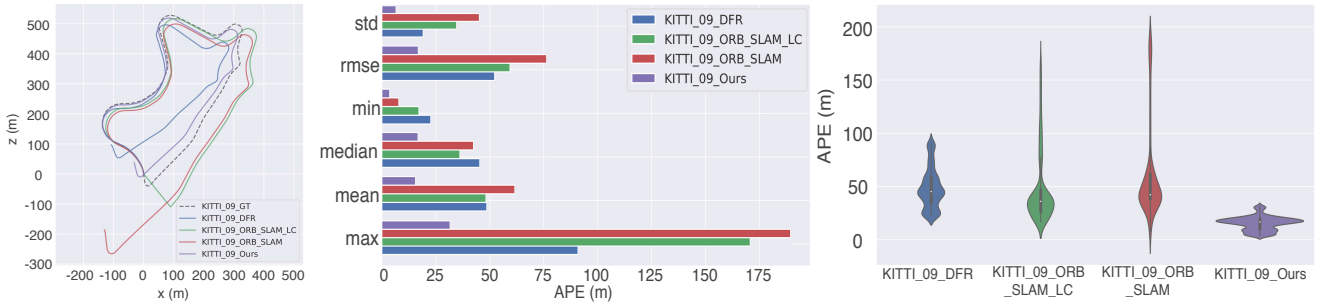

Figure 4. Qualitative results on visual odometry of our SynDeMo (Real&Synth. | Full) compared with ORB-SLAM [35] (with and without loop closure), DFR-Net [49] and the ground truth on the KITTI testing sequence 09. From left to right: pose trajectories, pose statistics, and the violin histogram poses.

Method	Seq. 09	Seq. 10
ORB-SLAM (full) [35] (IEEE T-RO 2015)	0.014±0.008	0.012 ±0.011
ORB-SLAM (short) [35] (IEEE T-RO 2015)	0.064±0.141	0.064±0.130
Mean Odom.	0.032±0.026	0.028±0.023
Zhou et al. [51] (CVPR 2017)	0.021±0.017	0.020±0.015
Yin et al. [48] (CVPR 2018)	0.012±0.007	0.012±0.009
Mahjourian et al. [33] (CVPR 2018)	0.013±0.010	0.012±0.011
Zou et al. [52] (ECCV 2018)	0.017±0.007	0.015±0.009
Luo et al. [32] (arXiv 2018)	0.012±0.006	0.012±0.008
Baseline	0.019±0.015	0.018±0.016
SynDeMo (Real&Synth$^{\ominus}$| Feat Align+View)	0.014±0.008	0.013±0.015
SynDeMo (Real&Synth. | Feat Align)	0.013±0.010	0.012±0.025
SynDeMo (Real&Synth. | View)	0.012±0.005	0.012±0.007
SynDeMo (Real&Synth. | Full)	**0.011±0.007**	**0.011±0.015**

Table 2. Absolute Trajectory Error (ATE) on KITTI odometry dataset. The results of other baselines are taken from [52].

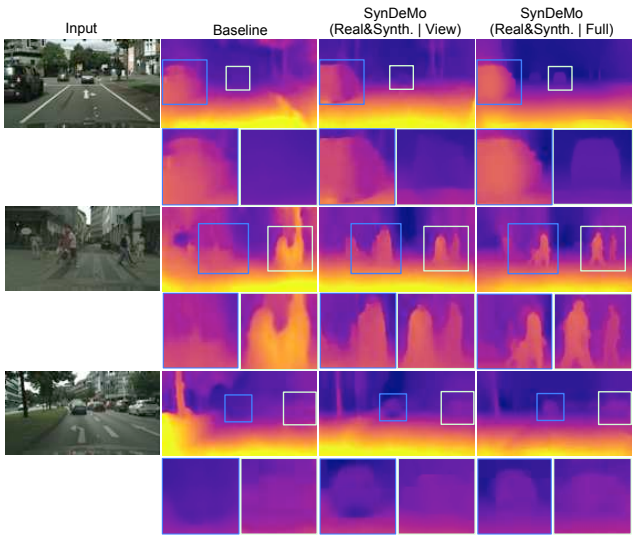

Figure 5. **Ablation study**. From left to right: input raw images, depth estimation results of the algorithm baseline, SynDeMo (Real&Synth. | View) and SynDeMo (Real&Synth. | Full).

tive is helpful in recovering scene geometry structure, when testing across different data domains.

Method	Supervision	Error Metric ↓			
		Abs Rel	Sq Rel	RMSE	RMSE log
Train set mean	-	0.876	12.98	12.27	0.307
Karsch et al. [23] (IEEE PAMI 2014)	depth	0.428	5.079	8.389	0.149
Liu et al. [31] (CVPR 2014)	depth	0.475	6.562	10.05	0.165
Laina et al. [26] (3DV 2016)	depth	0.204	**1.840**	5.683	0.084
Li et al. [28] (CVPR 2018)	depth	**0.176**	-	**4.260**	**0.069**
Zhou et al. [51] (CVPR 2017)	none	0.383	5.321	10.47	0.478
Godard et al. [16] (CVPR 2017)	pose	0.544	10.94	11.76	0.193
Zou et al. [52] (ECCV 2018)	none	0.331	2.698	6.890	0.416
Wang et al. [41] (CVPR 2018)	none	0.387	4.720	8.09	0.204
Baseline	none	0.375	5.305	10.15	0.445
SynDeMo (Real&Synth$^{\ominus}$| Feat Align+View)	none	0.330	2.692	6.850	0.412
Zheng et al. [50] (ECCV 2018)	synthetic depth	0.508	6.589	8.935	0.574
Atapour et al. [1] (CVPR 2018)	synthetic depth	0.423	9.343	9.002	0.122
SynDeMo (Real&Synth. | Feat Align)	synthetic depth	0.314	2.450	6.150	0.385
SynDeMo (Real&Synth. | View)	synthetic depth	0.311	2.440	6.137	0.379
SynDeMo (Real&Synth. | Full)	synthetic depth	0.295	2.155	5.874	0.115

Table 3. Results on the Make3D dataset [38]. Our results were obtained by the model trained on vKITTI + KITTI without training on Make3D data itself. Following the evaluation protocol of [16], the errors are only calculated, where ground truth depth is less than 70 meters. The overall best performance and the best results within each block are highlighted as **bold** or underlined, respectively.

5. Conclusion

We presented our SynDeMo for joint monocular depth and ego-motion estimation from video, which does not require any real ground truth depth samples or stereo image pairs for training. The estimated depth and pose from SynDeMo are both scaled thanks to the use of additional scene geometry information from synthetic images, a powerful cue for alleviating the domain gap. We propose a synergistic deep feature alignment by matching the feature distribution between real and synthetic domains. Our SynDeMo achieves state-of-the-art performance among unsupervised and cross-domain learning methods on both tasks of depth and visual odometry estimation.

The current limitation of our approach is that we are not modeling articulated objects or non-rigidity of the scene. A possible future extension of this work would be to model scene dynamics and articulated objects.

References

[1] Amir Atapour-Abarghouei and Toby P Breckon. Real-time monocular depth estimation using synthetic data with domain adaptation via image style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2800–2810, 2018.

[2] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[3] Vincent Casser, Soeren Pirk, Reza Mahjourian, and Anelia Angelova. Depth prediction without the sensors: Leveraging structure for unsupervised learning from monocular videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8001–8008, 2019.

[4] Chenyi Chen, Ari Seff, Alain Kornhauser, and Jianxiong Xiao. Deepdriving: Learning affordance for direct perception in autonomous driving. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2722–2730, 2015.

[5] Weifeng Chen, Zhao Fu, Dawei Yang, and Jia Deng. Single-image depth perception in the wild. In *Advances in Neural Information Processing Systems*, pages 730–738, 2016.

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

[7] Guilherme N DeSouza and Avinash C Kak. Vision for mobile robot navigation: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 24(2):237–267, 2002.

[8] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Advances in neural information processing systems*, pages 2366–2374, 2014.

[9] Friedrich Fraundorfer, Christopher Engels, and David Nistér. Topological mapping, localization and navigation using image collections. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3872–3877. IEEE, 2007.

[10] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2002–2011, 2018.

[11] Adrien Gaidon, Qiao Wang, Yohann Cabon, and Eleonora Vig. Virtual worlds as proxy for multi-object tracking analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4340–4349, 2016.

[12] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

[13] Ravi Garg, Vijay Kumar BG, Gustavo Carneiro, and Ian Reid. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *European Conference on Computer Vision*, pages 740–756. Springer, 2016.

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

[15] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361. IEEE, 2012.

[16] Clement Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 270–279, 2017.

[17] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel Brostow. Digging into self-supervised monocular depth estimation. *arXiv preprint arXiv:1806.01260*, 2018.

[18] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

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

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

[21] Philipp Heise, Sebastian Klose, Brian Jensen, and Alois Knoll. Pm-huber: Patchmatch with huber regularization for stereo matching. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2360–2367, 2013.

[22] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Advances in neural information processing systems*, pages 2017–2025, 2015.

[23] Kevin Karsch, Ce Liu, and Sing Bing Kang. Depth transfer: Depth extraction from video using non-parametric sampling. *IEEE transactions on pattern analysis and machine intelligence*, 36(11):2144–2158, 2014.

[24] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[25] Yevhen Kuznietsov, Jorg Stuckler, and Bastian Leibe. Semi-supervised deep learning for monocular depth map prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6647–6655, 2017.

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

[27] Ruihao Li, Sen Wang, Zhiqiang Long, and Dongbing Gu. Undeepvo: Monocular visual odometry through unsuper-

vised deep learning. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7286–7291. IEEE, 2018.

[28] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2041–2050, 2018.

[29] Fayao Liu, Chunhua Shen, and Guosheng Lin. Deep convolutional neural fields for depth estimation from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5162–5170, 2015.

[30] Fayao Liu, Chunhua Shen, Guosheng Lin, and Ian Reid. Learning depth from single monocular images using deep convolutional neural fields. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (10):2024–2039, 2016.

[31] Miaomiao Liu, Mathieu Salzmann, and Xuming He. Discrete-continuous depth estimation from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 716–723, 2014.

[32] Chenxu Luo, Zhenheng Yang, Peng Wang, Yang Wang, Wei Xu, Ram Nevatia, and Alan Yuille. Every pixel counts++: Joint learning of geometry and motion with 3d holistic understanding. *arXiv preprint arXiv:1810.06125*, 2018.

[33] Reza Mahjourian, Martin Wicke, and Anelia Angelova. Unsupervised learning of depth and ego-motion from monocular video using 3d geometric constraints. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5667–5675, 2018.

[34] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2794–2802, 2017.

[35] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: a versatile and accurate monocular slam system. *IEEE transactions on robotics*, 31(5):1147–1163, 2015.

[36] Georg Poier, Michael Opitz, David Schinagl, and Horst Bischof. Murauer: Mapping unlabeled real data for label austerity. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1393–1402. IEEE, 2019.

[37] Mahdi Rad, Markus Oberweger, and Vincent Lepetit. Feature mapping for learning fast and accurate 3d pose inference from synthetic images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4663–4672, 2018.

[38] Ashutosh Saxena, Sung H Chung, and Andrew Y Ng. Learning depth from single monocular images. In *Advances in neural information processing systems*, pages 1161–1168, 2006.

[39] Yi-Hsuan Tsai, Ming-Hsuan Yang, and Michael J Black. Video segmentation via object flow. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3899–3908, 2016.

[40] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7167–7176, 2017.

[41] Chaoyang Wang, José Miguel Buenaposada, Rui Zhu, and Simon Lucey. Learning depth from monocular videos using direct methods. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2022–2030, 2018.

[42] Rui Wang, Martin Schworer, and Daniel Cremers. Stereo dso: Large-scale direct sparse visual odometry with stereo cameras. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3903–3911, 2017.

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

[44] Dan Xu, Elisa Ricci, Wanli Ouyang, Xiaogang Wang, and Nicu Sebe. Multi-scale continuous crfs as sequential deep networks for monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5354–5362, 2017.

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

[46] Zhenheng Yang, Peng Wang, Yang Wang, Wei Xu, and Ram Nevatia. Every pixel counts: Unsupervised geometry learning with holistic 3d motion understanding. In *European Conference on Computer Vision*, pages 691–709. Springer, 2018.

[47] Zhenheng Yang, Peng Wang, Yang Wang, Wei Xu, and Ram Nevatia. Lego: Learning edge with geometry all at once by watching videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 225–234, 2018.

[48] Zhichao Yin and Jianping Shi. Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1983–1992, 2018.

[49] Huangying Zhan, Ravi Garg, Chamara Saroj Weerasekera, Kejie Li, Harsh Agarwal, and Ian Reid. Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 340–349, 2018.

[50] Chuanxia Zheng, Tat-Jen Cham, and Jianfei Cai. T2net: Synthetic-to-realistic translation for solving single-image depth estimation tasks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 767–783, 2018.

[51] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1851–1858, 2017.

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