Unsupervised Monocular Depth Learning in Dynamic Scenes

Hanhan Li[1] **Ariel Gordon**[1,2] **Hang Zhao**[3] **Vincent Casser**[3] **Anelia Angelova**[1,2]
{uniqueness@google, gariel@google, hangz@waymo, casser@waymo, anelia@google}.com
[1]Google Research [2]Robotics at Google [3]Waymo LLC

Abstract: We present a method for jointly training the estimation of depth, ego-motion, and a dense 3D translation field of objects relative to the scene, with monocular photometric consistency being the sole source of supervision. We show that this apparently heavily underdetermined problem can be regularized by imposing the following prior knowledge about 3D translation fields: they are sparse, since most of the scene is static, and they tend to be piecewise constant for rigid moving objects. We show that this regularization alone is sufficient to train monocular depth prediction models that exceed the accuracy achieved in prior work for dynamic scenes, including methods that require semantic input. [1]

Keywords: Unsupervised, Monocular Depth, Object Motion

1 Introduction

Understanding 3D geometry and object motion from camera images is an important problem for robotics applications, including autonomous vehicles [1] and drones [2]. While robotic systems are often equipped with various sensors, depth prediction from images remains appealing in that it solely requires an optical camera, which is a very cheap and robust sensor. Object motion estimation is a nontrivial problem across various sensors.

Estimating depth and object motion in 3D given a monocular video stream is an ill-posed problem, and generally heavily relies on prior knowledge. The latter is readily provided by deep networks, that can learn the priors through training on large collections of data.

Multiple methods have been devised for providing supervision to these networks. While depth prediction networks can be supervised by sensors [3, 4, 5], datasets providing object 3D motion supervision for image sequences are scarce. Self-supervised methods, which rely mainly on the monocular video itself for supervision, have been attracting increasing attention recently [6, 7, 8, 9, 10, 11, 12, 13], due to the virtually unlimited abundance of unlabeled video data.

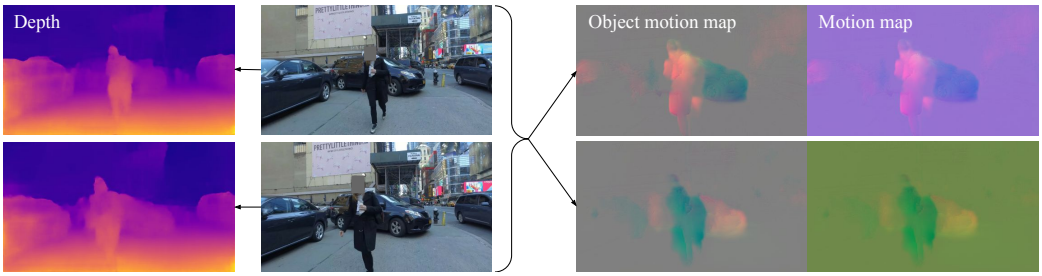

Figure 1: Depth prediction (for each frame separately) and motion map prediction (for a pair of frames), shown on a training video from YouTube. The total 3D motion map is obtained by adding the learned camera motion vector to the object motion map. Note that the motion map is mostly zero, and nearly constant throughout a moving object. This is a result of the motion regularizers used.

[1]Code is available at github.com/google-research/google-research/tree/master/depth_and_motion_learning

4th Conference on Robot Learning (CoRL 2020), Cambridge MA, USA.

Self-supervised learning of depth estimation is based on principles of structure from motion (SfM): When the same scene is observed from two different positions, the views will be consistent if a correct depth is assigned to each pixel, and the camera movement is correctly estimated. As such, these methods tend to suffer from many of the challenges of SfM: textureless areas, occlusions, reflections, and – perhaps above all – moving objects. Complete view consistency can only be achieved if the motion of every object between the capture times of the two frames is correctly accounted for. If any point in the scene is presumed to be in motion, it carries four unknowns (depth and three motion components), which is far too many for epipolar geometry constraints to disambiguate. Self-supervised methods thus often rely on additional cues.

One source of additional information is semantics [12]. Movable objects, such as vehicles can be identified using an auxiliary segmentation model, and a network can be trained to estimate the motion of each object separately. However such techniques depend on access to an auxiliary segmentation model, capable of segmenting out all classes of movable objects to appear in the video.

Other approaches utilize different types of prior knowledge. For example, a common case of object motion in the self-driving setting is where the observing car follows another car, at approximately the same velocity. The observed car thus appears static. Godard *et al.* [13] propose a method that identifies this case by detecting regions that do not change between frames and excludes these regions from the photometric consistency loss. It is a very common case in the KITTI dataset, and addressing it results in significant improvements in depth estimation metrics. Yet, the method remains limited to only one specific type of object motion.

Lastly, there are approaches where optical flow is learned jointly with depth, unsupervised [14]. However, stereo input is used to disambiguate the depth prediction problem.

The main contribution of this paper is a method for learning jointly depth, ego-motion and a dense object motion map in 3D from *monocular video only*, where unlike prior work, our method:

- Does not utilize *any* auxiliary signals apart from the monocular video itself: neither semantic signals, nor stereo, nor any kind of groundtruth.
- Accounts for any object motion pattern that can be approximated by a rigid object translation in an arbitrary direction.

A key contribution of our paper is a novel regularization method for the residual translation fields, based on the $\frac{1}{2}$ norm, which casts the residual motion field into the desired pattern described above.

In our method, a deep network predicts a dense 3D translation field (from a pair of frames). The translation field can be decomposed to the sum of background translation relative to the camera (due to ego-motion), which is constant, and an object translation field, which accounts for the motion of every point in the field of view relative to the scene. Another network predicts depth for each pixel (from each frame separately), totaling four predicted quantities per pixel. Figure 1 illustrates the depth and translation fields. At inference time, depth is obtained from a single frame, whereas camera motion and the object translation field are obtained from a pair of frames.

Since we aim to only use a monocular video for supervision, this problem requires significant amounts of regularization. We aim to strike the right balance between regularizing sufficiently, while at the same time preserving the ability to model diverse and complex motion patterns in highly dynamic scenes. We achieve this by utilizing two observations about the nature of residual translation fields: (1) They are *sparse*, since typically most of the pixels in a frame belong to the background or static objects, and (2) They tend to be *constant throughout a rigid moving object* in 3D space.

We evaluate the performance of the method on four challenging datasets with dynamic scenes. We establish new state-of-the-art results for unsupervised depth prediction on Cityscapes [15] and the Waymo Open Dataset [16], and match the state of the art on KITTI[2] [17]. To further demonstrate the generality of our approach, we also train and qualitatively evaluate our model on a collection of public YouTube videos taken with hand-held cameras while walking in a variety of environments. Qualitative results from all the above datasets are shown in Figure 2.

[2]Since dynamic scenes are rare in the KITTI dataset, the improvements of our method on this benchmark are less pronounced.

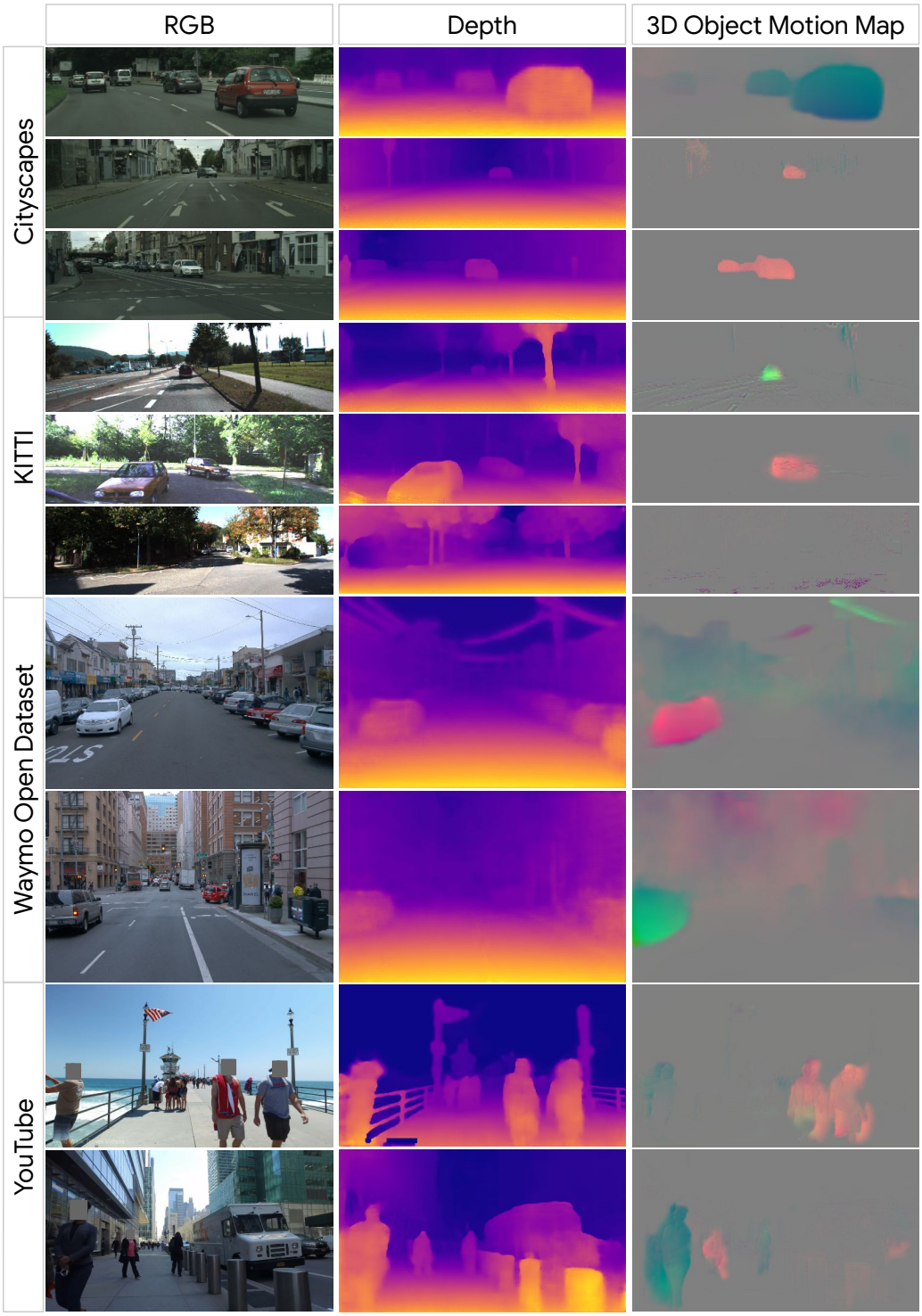

Figure 2: Qualitative results of our unsupervised monocular depth and 3D object motion map learning in dynamic scenes across all datasets: Cityscapes, KITTI, Waymo Open Dataset and YouTube.

2 Related Work

Structure from Motion and Multiview Stereo. Depth estimation is an important task for 3D scene understanding and robotics. Traditional computer vision approaches rely on identifying correspondences between keypoints in two or more images of the same scene and using epipolar geometry to solve for their depths [18, 19, 20]. These methods yield sparse depth maps. They can be applied to dynamic scenes in a multi-camera setting ("multiview stereo"), or to static scenes in a single moving camera setting ("structure from motion"). Please see [21] for a detailed survey.

Depth Estimation. In deep-learning based approaches [3, 4, 22, 5, 23], a deep network predicts a dense depth map. These networks can be trained by direct supervision, such as through LiDAR sensors. Similar approaches are used for other dense predictions such as surface normals [24, 25]. More recently, deep-learning approaches have been used in conjunction with classical computer vision techniques to learn depth and ego-motion prediction [6, 26, 27, 28, 29, 7]. Instead of identifying keypoints in scenes and finding correspondences, deep networks predict dense depth maps and camera motion, and these are used for differentiably warping pixels from one view to another. By applying photometric consistency losses on a transformed and the corresponding reference view, a supervision signal for the model is derived. A lot of progress in monocular depth learning has followed [28, 30, 29, 7, 31, 12, 32], and some work proposed to use stereo inputs for training, with the purpose of producing more accurate monocular depth at inference [27, 28, 14, 31, 33]. Alternative approaches learn to produce the stereo disparity [34, 35, 36], or apply depth completion from an online depth sensor [37, 38].

Depth and Motion. When the camera and objects move with respect to the scene, enforcing consistency across views requires estimating motion of both the camera as well as individual objects. As estimating depth in dynamic scenes is very challenging, many approaches have used multiple views to obtain depth [39, 40]. Several approaches have recently been proposed for simultaneously learning depth, camera motion, and object motion from monocular videos. Yin *et al.* [7] and Zhou *et al.* [9] learn to jointly predict depth, ego-motion and optical flow. The first stage of their method estimates depth and camera motion, and thus the optical flow induced by camera motion. A second stage estimates the residual optical flow due to object motion relative to the scene. The residual flow is used to mask out moving objects and to reason about occlusions. Luo *et al.* [10] also jointly optimize depth, camera motion, and optical flow estimation, using a holistic motion parser. Stereo input is used to disambiguate the depth prediction training. Casser *et al.* [12, 41] estimate the motion of objects in the scenes, with the assistance of pre-trained segmentation models, leading to significant improvement in depth estimation for moving objects. Gordon *et al.* [11] introduce a differentiable way of handling occlusions but still require an auxiliary segmentation model for moving objects. Li *et al.* [32] are able to handle scenes with moving people but require human-segmentation masks and relies on a dataset with "static" people, observed from multiple views. Godard *et al.* [13] address object motion by detecting cases where an object is static with respect to the camera (i. e. moving at the same speed), which addresses a very common case of object motion observed from vehicles in traffic. Our work provides a more general way of performing unsupervised depth learning in dynamic scenes; unlike previous work, we do not need to segment out objects in order to estimate their motion, nor do we assume stereo data. Unlike prior work that uses residual optical flow to reason about moving objects and occlusions, our method directly regularizes motion in 3D, which turns out to lead to better depth prediction accuracy.

3 Method

In our approach, depth, ego-motion, and object motion are learned simultaneously from monocular videos using self-supervision. We use pairs of adjacent video frames ($\mathbf{I_a}$ and $\mathbf{I_b}$) as training data. Our depth network predicts a depth map $D(u, v)$ (where u and v are the image coordinates) at the original resolution from a single image, and we apply it independently on each of the two frames. The two depth maps are concatenated with $\mathbf{I_a}$ and $\mathbf{I_b}$ in the channel dimension and are fed into a motion prediction network (Figure 1). The latter predicts a 3D translation map $T_{\text{obj}}(u, v)$ at the original resolution for the moving objects and a 6D ego-motion vector M_{ego}. M_{ego} consists of the 3 Euler angles that parameterize the 3D rotation matrix $\mathbf{R}$ and elements of the 3D translation vector T_{ego}. The object motion relative to the camera is defined as rotation $\mathbf{R}$ followed by a translation

$$T(u, v) = T_{\text{obj}}(u, v) + T_{\text{ego}}. \tag{1}$$

We propose new motion regularization losses on $T(u, v)$ (Sec. 3.2.1) which facilitate training in highly dynamic scenes. The overall training setup is shown in Figure 3. Figure 2 visualizes examples of the learned motion $T_{\text{obj}}(u, v)$ and disparity $d(u, v) = 1/D(u, v)$ per frame.

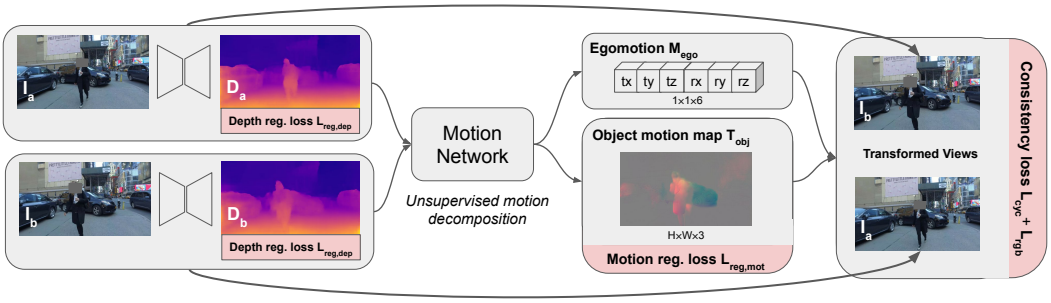

Figure 3: Overall training setup. A depth network is independently applied on two adjacent RGB frames, $\mathbf{I_a}$ and $\mathbf{I_b}$, to produce the depth maps, $\mathbf{D_a}$ and $\mathbf{D_b}$. The depth maps together with the two original images are fed into the motion network, which decomposes the motion into a global ego-motion estimate M_{ego} and a spatial object motion map T_{obj}. Given motion and depth estimates, a differentiable view transformer allows transitioning between them. Losses are highlighted in red. For example, we use a motion regularization loss (Section 3.2.1) on the motion map, and a motion cycle consistency loss and a photometric consistency loss (Section 3.2.3). At inference time, a depth map is obtained from a single frame, whereas a 3D motion map and ego-motion are obtained from two consecutive frames.

3.1 Depth and Motion Networks

Our depth network is an encoder-decoder architecture, identical to the one in Ref. [6], with the only difference that we use a softplus activation function for the depth, $z(\ell) = \log(1 + \mathrm{e}^{\ell})$. In addition, randomized layer normalization [11] is applied before each $relu$ activation in the network.

The input to the motion network is a pair of consecutive frames, concatenated along the channel dimension. The motion prediction is similar to the one in Ref. [11], with the difference that each input image has four channels: The three RGB channels, and the predicted depth as the fourth channel. The rationale is that having depth as a fourth channel, even if predicted rather than exactly measured, provides helpful signals for the task of estimating motion in 3D.

3.2 Losses

Training is driven by a number of self-supervised losses as is standard in monocular or stereo-based unsupervised depth and ego-motion learning. The total loss is the sum of three components, the motion regularization, the depth regularization, and the consistency regularization. The motion regularization and consistency regularization are applied twice on the frame pair, and the frame order is reversed in the second application. The depth regularization is applied independently on each one of pair. We describe the losses below, starting with the motion regularization losses, which are a key contribution of this work.

3.2.1 Motion Regularization

The regularization $L_{\text{reg,mot}}$ on the motion map $T_{\text{obj}}(u, v)$ consists of the ***group smoothness loss*** L_{g1} and the $L_{1/2}$ ***sparsity loss***. The group smoothness loss L_{g1} on $T_{\text{obj}}(u, v)$ minimizes changes within the moving areas, encouraging the motion map to be nearly constant throughout a moving object. This is done in anticipation that moving objects are mostly rigid. It is defined as:

$$L_{g1}[T(u, v)] = \sum_{i \in \{x,y,z\}} \iint \sqrt{\left(\partial_u T_i(u, v)\right)^2 + \left(\partial_v T_i(u, v)\right)^2} \mathrm{d}u\mathrm{d}v \tag{2}$$

The $L_{1/2}$ sparsity loss on $T_{\text{obj}}(u, v)$ is defined as:

$$L_{1/2}[T(u, v)] = 2 \sum_{i \in \{x,y,z\}} \langle |T_i| \rangle \iint \sqrt{1 + |T_i(u, v)| / \langle |T_i| \rangle} \mathrm{d}u\mathrm{d}v \tag{3}$$

where $\langle|T_i|\rangle$ is the spatial average of $|T_i(u,v)|$. The coefficients are designed in this way so that the regularization is self-normalizing. In addition, it approaches L_1 for small $T(u,v)$, and its strength becomes weaker for larger $T(u,v)$. We visualize its behavior in the Supplemental Material. Overall, the $L_{1/2}$ loss encourages more sparsity than the L_1 loss.

The final motion regularization loss is a combination of the above losses:

$$L_{\text{reg,mot}} = \alpha_{\text{mot}} L_{g1}[T_{\text{obj}}(u,v)] + \beta_{\text{mot}} L_{1/2}[T_{\text{obj}}(u,v)] \tag{4}$$

where α_{mot} and β_{mot} are hyperparameters.

Strictly speaking, a piecewise-constant $T_{\text{obj}}(u,v)$ can describe any scene where objects are moving in pure translation relative to the background. However, when objects are rotating, the residual translation field is generally not constant throughout them. Since fast rotation of objects relative to the background is uncommon, especially in road traffic, we expect the piecewise-constant approximation to be appropriate.

3.2.2 Depth Regularization

We apply a standard edge-aware smoothness regularization on the disparity maps $d(u,v)$ as described in Godard *et al.* [27]. In other words, the regularization is weaker around pixels where color variation is higher:

$$L_{\text{reg,dep}} = \alpha_{\text{dep}} \iint (|\partial_u d(u,v)| \mathrm{e}^{-|\partial_u \mathbf{I}(u,v)|} + |\partial_v d(u,v)| \mathrm{e}^{-|\partial_v \mathbf{I}(u,v)|}) \mathrm{d}u\mathrm{d}v \tag{5}$$

where α_{dep} is a hyperparameter.

3.2.3 Consistency Regularization

The consistency regularization is the sum of the motion cycle consistency loss L_{cyc} and the occlusion-aware photometric consistency loss L_{rgb}. L_{cyc} encourages the forward and backward motion between any pair of frames to be the opposite of each other.

$$L_{\text{cyc}} = \alpha_{\text{cyc}} \frac{\|\mathbf{R}\mathbf{R}_{\text{inv}} - \mathbb{1}\|^2}{\|\mathbf{R} - \mathbb{1}\|^2 + \|\mathbf{R}_{\text{inv}} - \mathbb{1}\|^2} + \beta_{\text{cyc}} \iint \frac{\|\mathbf{R}_{\text{inv}} T(u,v) + T_{\text{inv}}(u_{\text{warp}}, v_{\text{warp}})\|^2}{\|T(u,v)\|^2 + \|T_{\text{inv}}(u_{\text{warp}}, v_{\text{warp}})\|^2} \mathrm{d}u\mathrm{d}v \tag{6}$$

where the 'inv' subscript indicates that the same quantity was obtained with the input frames reversed in order. α_{cyc} and β_{cyc} are hyperparameters.

L_{rgb} encourages photometric consistency of corresponding areas in the two input frames. Similar to prior works, it is a sum of a L1 loss and a SSIM structural similarity loss in the RGB space.

$$L_{\text{rgb}} = \alpha_{\text{rgb}} \iint |\mathbf{I}(u,v) - \mathbf{I}_{\text{warp}}(u,v)| \, \mathbb{1}_{D(u,v) > D_{\text{warp}}(u,v)} \mathrm{d}u\mathrm{d}v + \beta_{\text{rgb}} \frac{1 - \text{SSIM}(\mathbf{I}, \mathbf{I}_{\text{warp}})}{2} \tag{7}$$

where $\mathbf{I}$ and $\mathbf{I}_{\text{warp}}$ are the original image and warped image, $\mathbb{1}_{D(u,v) > D_{\text{warp}}(u,v)}$ is a mask introduced in Gordon *et al.* [11] to address occlusions. α_{rgb} and β_{rgb} are hyperparameters.

The frame warping is calculated using the camera intrinsic matrix $\mathbf{K}$, the rotation matrix $\mathbf{R}$, and the camera translation vector T as $z'p' = \mathbf{K}\mathbf{R}\mathbf{K}^{-1} zp + \mathbf{K}T$. Here p and z are the homogeneous pixel coordinates and the depth, and their primed counterparts are warped ones in the new frame.

4 Experiments

In this section, we present results on a variety of datasets, including Cityscapes, KITTI, Waymo Open Dataset and a collection of videos taken with moving cameras from YouTube. For all our experiments, the encoder part of the depth network is initialized from a network pretrained on ImageNet [42]. The camera intrinsic matrices are provided in all datasets except for Youtube videos, where they are learned as part of our models. The hyperparameter values we used are the same for all experiments and are given in the supplementary material.

The inference time of our depth prediction model is about 5.3ms per frame of resolution 480x192, at a batch size of 1 on a NVIDIA V100 GPU (unoptimized), which is equivalent to roughly 190 frames per second. As far as we are aware, this is among the fastest methods.

4.1 Cityscapes

The Cityscapes [15] dataset is an urban driving dataset, which is quite challenging for unsupervised monocular depth estimation, because of the prevalence of dynamic scenes. As a result, not many works published results on this dataset, with few exceptions [12, 11, 43]. We use standard evaluation protocols as in prior work [12, 43]. For training we combine the densely and coarsely annotated splits to obtain 22,973 image-pairs. For evaluation, we use the 1,525 test images. The evaluation uses the code and methodology from Struct2Depth [12].

Method	Uses semantics?	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Struct2Depth [12]	Yes	0.145	1.737	7.28	0.205	0.813	0.942	0.978
Gordon [11]	Yes	0.127	1.33	**6.96**	0.195	0.830	**0.947**	**0.981**
Pilzer [43]	**No**	0.440	6.04	5.44	0.398	0.730	0.887	0.944
Ours	**No**	**0.119**	**1.29**	**6.98**	**0.190**	**0.846**	**0.952**	**0.982**

Table 1: Performance comparison of unsupervised single-view depth learning approaches, for models trained and evaluated on Cityscapes using the standard split. The depth cutoff is 80m. Our model uses a resolution of 416×128 for input/output. The 'uses semantics' column indicates whether the corresponding method requires a pretrained mask network to help identify moving objects. Our approach does not use semantics information. 'Abs Rel', 'Sq Rel', 'RMSE', and 'RMSE log' denotes mean absolute error, squared error, root mean squared error, and root mean squared logarithmic error respectively. $\delta < x$ denotes the fraction with the ratio between groundtruth and prediction values between x and $1/x$. For the red metrics, lower is better; for the green metrics, higher is better.

Method	Abs Rel	Sq Rel	RMSE	RMSE log
Ours, $L_{1/2}$, without depth prediction inputs	0.125	1.41	7.39	0.200
Ours, L_1 instead of $L_{1/2}$	0.125	1.37	7.33	0.199
Ours, $L_{1/2}$	**0.119**	**1.29**	6.98	0.190
Ours, $L_{1/2}$, with mask	**0.119**	1.36	**6.89**	**0.188**

Table 2: Ablation study on Cityscapes. $L_{1/2}$ is the loss term defined in Eq. 3. 'with mask' means using a pretrained segmentation model to identify regions of potentially moving objects, as in prior work [12].

Table 1 compares the performance of our approach with prior works which reported results on Cityscapes. Our method is able to outperform all prior methods except for the metric (RMSE) compared to Ref. [11]. However the latter method uses semantic cues.

Table 2 shows an ablation study on this dataset, as it contains many dynamic scenes. If we do not feed the depth predictions into the motion network, performance deteriorates - 'Abs Rel' increases by about 0.006. If we use the L_1 loss instead of the $L_{1/2}$ loss to regularize the object motion map, 'Abs Rel' also increases by about 0.006. We also tried augmenting our approach with a detection model pretrained in the COCO [44] dataset. This model produces a mask for each image which is applied onto the object motion map. However, the detection model does not cause noticeable improvements for depth estimation.

4.2 KITTI

The KITTI [17] dataset is collected in urban environments and is a popular benchmark for depth and ego-motion estimation. It is accompanied with LiDAR data, which is used for evaluation only. While KITTI only has a small number of dynamic scenes, it is a very common dataset for evaluating depth models. We follow the Eigen split with 22,600 training image pairs and 697 evaluation pairs. We use the standard evaluation protocol, established by Zhou *et al.* [6] and adopted by many subsequent works. Table 3 shows that the performance of our model is on par with the state of the art.

4.3 Waymo Open Dataset

The Waymo Open Dataset [16] is currently one of the largest and most diverse publicly released autonomous driving datasets. Its scenes are not only dynamic but also comprise nighttime driving and diverse weather conditions. We experiment on this dataset to showcase the generality of our method. For training, we take $100,000$ image-pairs from $1,000$ front camera video sequences. For

Method	Uses semantics?	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Struct2Depth [12]	Yes	0.141	1.026	5.291	0.2153	0.8160	0.9452	0.9791
Gordon [11]	Yes	**0.128**	**0.959**	5.23	0.212	0.845	0.947	0.976
Yang [10]	**No**	0.141	1.029	5.350	0.216	0.816	0.941	0.976
Bian [45]	**No**	0.137	1.089	5.439	0.217	0.830	0.942	0.975
Godard [13]	**No**	**0.128**	1.087	5.171	**0.204**	**0.855**	**0.953**	**0.978**
Ours	**No**	0.130	**0.950**	**5.138**	0.209	0.843	0.948	**0.978**

Table 3: Performance comparison of unsupervised single-view depth learning approaches, for models trained and evaluated on KITTI using the Eigen Split. The depth cutoff is 80m. All results in the table (including ours) are for a resolution of 416×128 for input/output. Please refer to Table 1 for more details on the table schema.

evaluation, we use $1,500$ image-pairs from 150 sequences. Ground truth depth from LiDAR is only used for evaluation. Since previous unsupervised depth approaches did not publish results on this dataset, we compare using available open-source code of state-of-the-art methods, capable of handling dynamic scenes.

Table 4 shows the performance of our model in comparison with prior works, both of which require semantics cues. Our model, which does not need any masks, outperforms previous works that additionally use object masks. We also apply object masks to our method (last row), and observe further improvements comparing to the maskless setting (second last row), showing the effectiveness of our proposed regularization with or without mask. Visual examples are shown in Figure 2 and in the Supplemental Material. As shown, our approach is successfully handling these challenging scenes.

Method	Abs Rel	Sq Rel	RMSE	RMSE log
Open-source code from [12], with Mask	0.180	1.782	8.583	0.244
Open-source code from [11], with Mask	0.168	1.738	7.947	0.230
Ours, without Mask	**0.162**	**1.711**	**7.833**	**0.223**
Ours, with Mask	**0.157**	**1.531**	**7.090**	**0.205**

Table 4: Performance on the Waymo Open Dataset. Our approach outperforms prior work, even though it does not require masks (top portion). With masks, it performs even better (bottom).

The supplementary material includes videos of our method running on validation sequences of the Waymo Open Dataset and Cityscapes.

4.4 YouTube dataset

To demonstrate that depth can be learned from videos in the wild, we randomly picked a collection of videos on YouTube taken with handheld cameras while walking. The videos contain many different scenes across quite a few cities, and they feature objects in a wide range of depths. Because the videos are taken with unknown cameras, we simultaneously learn an intrinsic matrix per video. An intrinsic matrix is parameterized as a set of four trainable variables: two for the focal lengths and two for the optical centers [11]. Visual depth maps are shown in Figure 2 and in the supplemental material.

5 Conclusions

This paper presents a novel unsupervised method for depth learning in highly dynamic scenes, which jointly solves for 3D motion maps and depth maps. Our model can be trained on unlabeled monocular videos without requiring any auxiliary semantic information. Our method is conceptually very simple, as we use end-to-end differentiable losses that encourage photometric consistency, motion smoothness, and motion sparsity. The main limitation is that object rotation and deformation is not explicitly handled, and that camera movement needs to be present to receive learning signals. We demonstrate the efficacy of the proposed approach on Cityscapes, KITTI, the Waymo Open Dataset, and YouTube data. For datasets rich in dynamic scenes, we outperform prior depth estimation benchmarks, including ones that utilize semantic cues.

References

[1] J. Janai, F. Güney, A. Behl, and A. Geiger. Computer vision for autonomous vehicles: Problems, datasets and state-of-the-art. *arXiv preprint arXiv:1704.05519*, 2017.

[2] S. Saripalli, J. F. Montgomery, and G. S. Sukhatme. Vision-based autonomous landing of an unmanned aerial vehicle. In *ICRA*, volume 3, pages 2799–2804. IEEE, 2002.

[3] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. In *NIPS*, pages 2366–2374, 2014.

[4] F. Liu, C. Shen, G. Lin, and I. Reid. Learning depth from single monocular images using deep convolutional neural fields. *IEEE transactions on pattern analysis and machine intelligence*, 38(10):2024–2039, 2015.

[5] Y. Kuznietsov, J. Stuckler, and B. Leibe. Semi-supervised deep learning for monocular depth map prediction. In *CVPR*, pages 6647–6655, 2017.

[6] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe. Unsupervised learning of depth and ego-motion from video. In *CVPR*, pages 1851–1858, 2017.

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

[8] A. Ranjan, V. Jampani, L. Balles, K. Kim, D. Sun, J. Wulff, and M. J. Black. Competitive collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation. In *CVPR*, pages 12240–12249, 2019.

[9] Y. Zou, Z. Luo, and J.-B. Huang. Df-net: Unsupervised joint learning of depth and flow using cross-task consistency. In *ECCV*, pages 36–53, 2018.

[10] C. Luo, Z. Yang, P. Wang, Y. Wang, W. Xu, R. Nevatia, and A. Yuille. Every pixel counts++: Joint learning of geometry and motion with 3d holistic understanding. *arXiv preprint arXiv:1810.06125*, 2018.

[11] A. Gordon, H. Li, R. Jonschkowski, and A. Angelova. Depth from videos in the wild: Unsupervised monocular depth learning from unknown cameras. In *ICCV*, October 2019.

[12] V. Casser, S. Pirk, R. Mahjourian, and A. 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.

[13] C. Godard, O. M. Aodha, M. Firman, and G. Brostow. Digging into self-supervised monocular depth estimation. *ICCV*, 2019.

[14] Z. Yang, P. Wang, Y. Wang, W. Xu, and R. Nevatia. Every pixel counts: Unsupervised geometry learning with holistic 3d motion understanding. In *ECCV*, 2018.

[15] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016.

[16] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, V. Vasudevan, W. Han, J. Ngiam, H. Zhao, A. Timofeev, S. Ettinger, M. Krivokon, A. Gao, A. Joshi, Y. Zhang, J. Shlens, Z. Chen, and D. Anguelov. Scalability in perception for autonomous driving: Waymo open dataset, 2019.

[17] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013.

[18] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[19] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM transactions on graphics (TOG)*, volume 25, pages 835–846. ACM, 2006.

[20] J. L. Schonberger and J.-M. Frahm. Structure-from-motion revisited. *CVPR*, 2016.

[21] O. Özyeşil, V. Voroninski, R. Basri, and A. Singer. A survey of structure from motion. *Acta Numerica*, 26:305–364, 2017. doi:10.1017/S096249291700006X.

[22] Z. W. Yuanzhouhan Cao and C. Shen. Estimating depth from monocular images as classification using deep fully convolutional residual networks. *CoRR:1605.02305*, 2016.

[23] S. Tulsiani, T. Zhou, A. A. Efros, and J. Malik. Multi-view supervision for single-view reconstruction via differentiable ray consistency. *CVPR*, 2017.

[24] D. Eigen and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. *ICCV*, 2015.

[25] A. G. Xiaolong Wang, David F. Fouhey. Designing deep networks for surface normal estimation. *CVPR*, 2015.

[26] R. Garg, V. K. BG, G. Carneiro, and I. Reid. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *ECCV*, pages 740–756. Springer, 2016.

[27] C. Godard, O. Mac Aodha, and G. J. Brostow. Unsupervised monocular depth estimation with left-right consistency. In *CVPR*, pages 270–279, 2017.

[28] B. Ummenhofer, H. Zhou, J. Uhrig, N. Mayer, E. Ilg, A. Dosovitskiy, and T. Brox. Demon: Depth and motion network for learning monocular stereo. *CVPR*, 2017.

[29] R. Mahjourian, M. Wicke, and A. Angelova. Unsupervised learning of depth and ego-motion from monocular video using 3d geometric constraints. In *CVPR*, pages 5667–5675, 2018.

[30] Z. Yang, P. Wang, W. Xu, L. Zhao, and R. Nevatia. Unsupervised learning of geometry with edge-aware depth-normal consistency. *arXiv:1711.03665*, 2017.

[31] C. Wang, J. M. Buenaposada, R. Zhu, and S. Lucey. Learning depth from monocular videos using direct methods. *CVPR*, 2018.

[32] Z. Li, T. Dekel, F. Cole, R. Tucker, N. Snavely, C. Liu, and W. T. Freeman. Learning the depths of moving people by watching frozen people. In *CVPR*, 2019.

[33] H. Zhan, R. Garg, C. S. Weerasekera, K. Li, H. Agarwal, and I. Reid. Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction. *CVPR*, 2018.

[34] 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*, Oct 2017.

[35] S. Khamis, S. Fanello, C. Rhemann, A. Kowdle, J. Valentin, and S. Izadi. Stereonet: Guided hierarchical refinement for real-time edge-aware depth prediction. In *ECCV*, September 2018.

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

[37] F. Ma and S. Karaman. Sparse-to-dense: Depth prediction from sparse depth samples and a single image. In *ICRA*, 2018.

[38] Y. Zhang and T. Funkhouser. Deep depth completion of a single rgb-d image. In *CVPR*, 2018.

[39] R. Ranftl, V. Vineet, Q. Chen, and V. Koltun. Dense monocular depth estimation in complex dynamic scenes. In *CVPR*, page 4058–4066, 2016.

[40] S. Kumar, Y. Dai, and H. Li. Monocular dense 3d reconstruction of a complex dynamic scene from two perspective frames. In *ICCV*, 2017.

[41] V. Casser, S. Pirk, R. Mahjourian, and A. Angelova. Unsupervised monocular depth and ego-motion learning with structure and semantics. In *CVPR Workshops*, 2019.

[42] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255. Ieee, 2009.

[43] A. Pilzer, D. Xu, M. M. Puscas, E. Ricci, and N. Sebe. Unsupervised adversarial depth estimation using cycled generative networks. *3DV*, 2018.

[44] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755. Springer, 2014.

[45] J.-W. Bian, Z. Li, N. Wang, H. Zhan, C. Shen, M.-M. Cheng, and I. Reid. Unsupervised scale-consistent depth and ego-motion learning from monocular video. *arXiv preprint arXiv:1908.10553*, 2019.