Instance-Aware Multi-Object Self-Supervision for Monocular Depth Prediction

Houssem Eddine Boulahbal, Adrian Voicila, and Andrew I. Comport

Abstract—This letter proposes a self-supervised monocular image-to-depth prediction framework that is trained with an end-to-end photometric loss that handles not only 6-DOF camera motion but also 6-DOF moving object instances. Self-supervision is performed by warping the images across a video sequence using depth and scene motion including object instances. One novelty of the proposed method is the use of the multi-head attention of the transformer network that matches moving objects across time and models their interaction and dynamics. This enables accurate and robust pose estimation for each object instance. Most image-to-depth predication frameworks make the assumption of rigid scenes, which largely degrades their performance with respect to dynamic objects. Only a few state-of-the-art (SOTA) papers have accounted for dynamic objects. The proposed method is shown to outperform these methods on standard benchmarks and the impact of the dynamic motion on these benchmarks is exposed. Furthermore, the proposed image-to-depth prediction framework is also shown to be competitive with SOTA video-to-depth prediction frameworks.

Index Terms—Depth prediction, motion prediction, multi-object detection.

I. INTRODUCTION

MONOCULAR depth prediction is a prominent problem in computer vision, with many applications in robotics, AR/VR, autonomous driving and its related downstream tasks, e.g. automatic emergency braking (AEB). While Lidar scans provide high accuracy depth measurements, generating high quality depth from a monocular camera is attractive due to the cost-efficiency and availability of such systems. Recent developments based on deep learning methods [1], [4], [10], [20], [25], [26], [28], [33], [35], [41] have demonstrated competitive depth prediction quality. These methods are formulated via self-supervised learning based on making use of monocular video during training. Thus, large and accurate ground-truth datasets are not required. One way to design such a training model is to introduce a depth network alongside a pose network that can estimate the pose between two video frames. These networks are optimized using a photometric loss between the target image and a set of warped images obtained using the pose and the depth.

A common assumption for self-supervised training is the rigid scene assumption, i.e. a static scene and a moving camera. However, this assumption is often violated due to the motion of moving objects present in the scene. A possible solution is to mask the pixels of dynamic objects using either a learned mask [41], semantic guidance [17] or auto-masking [9], [35]. However, these solutions are merely a workaround to avoid...
non rigid-scenes and they subsequently miss data on moving objects that could otherwise be useful for further constraining depth prediction. [18], [25], [32], [37], [40] have addressed the moving object in the scene. [18], [32], [37] learn a per-object semantic segmentation mask and a motion field that accounts for the moving object. [25], [40] rely on the optical flow which does not explicitly model the 6-DOF motion of objects separately. These networks are optimized for local rigidity and the notion of the class and the possible dynamics for each class is not taken into account.

A proposition is made in this letter to alleviate this assumption. Non-rigid scenes are learnt by factorizing the motion into the dominant ego-pose and a piece-wise rigid pose for each dynamic object explicitly. Therefore, for static objects, only the ego-pose is used for the warping, while the dynamic objects are subject to two transformations using the motion of the camera and the motion of each moving object. The proposed method explicitly models the motion of each object allowing accurate warping of the scene elements. In order to model this motion, The proposed method makes use of the multi-head attention of the transformer network that matches moving objects across time and models their interaction and dynamics. This enables accurate and robust pose estimation for each object instance. The proposed method achieves SOTA results on the KITTI benchmark. In summary, the contributions of this letter are:

- A novel network architecture based on the transformers multi-head attention that explicitly models the dynamics of moving objects.
- An accurate and robust per-object pose is obtained by matching and modeling the interaction of the objects across time.
- High quality depth prediction achieving competitive performance with respect to state-of-the-art results on the KITTI benchmark [8].
- The demonstration that the KITTI benchmark has a bias favoring static scenes and a method to test the quality of moving object depth prediction.

II. RELATED WORK

A. Self-Supervised Depth Prediction

Depth prediction has been successful with self-supervised learning from videos. The seminal work of Zhou et al. [41] introduced the core idea to jointly optimize the pose and depth network using image reconstruction and photometric loss. To account for the ill-posed nature of this task, several works have addressed different challenges. [20], [32], [37] addressed the rigid scene assumption. [10], [28] proposed more robust image reconstruction losses to reject the outliers such as occlusion during training. [4] have addressed learning the camera parameters for better generalization. [1], [4], [26], [33] have addressed the scale ambiguity problem and propose to enforce depth scale and structure consistency. [21], [35] employed a test-time refinement by allowing the model parameters to vary dynamically during inference using a photometric loss. Similarly, the proposed paper jointly optimize the pose and depth using the photometric loss as supervisory signal and addresses specifically the problem of moving objects in the scene.

B. Camera and Object Motion Factorization

Supervising the depth with a photometric loss is problematic when moving objects are present in the scene. This challenge has gained attention in the literature. A common solution is to disentangle the dominant ego-motion and the object motion. [4], [14], [25], [40] leverage an optical flow network to detect moving objects by comparing the optical flow with depth-based mapping. [19] learns a monocular depth in order to estimate the motion field as two stage learning. [32] learns a per-object semantic segmentation mask and a motion field is obtained by factorization of the motion of each mask and the ego-motion. [27] addresses the object motion without additional labels by proposing a scene decomposition into a fixed number of components where a pose is predicted for each component. [37] relaxes the problem using local rigidity within a predefined window. [20] leverages the geometric consistency of depth, ego-pose and optical flow and categorises each pixel as either rigid motion, non-rigid/object motion or occluded/non-visible regions. A recent work that is closest to the proposed method is Insta-DM [18]. In that method, the source and target images are masked with semantic masks and an object PoseNet is used to learn the pose from the masked RGB images. Alternatively, the method proposed in this letter factorizes the motion into ego-motion and object-motion and exploits a transformer attention network to perform instance segmentation and learn a per-object motion.

III. METHOD

A. Problem Formulation

The aim of monocular depth prediction is to learn an accurate depth map through the mapping \( D_t = f(I_t; \theta) \) where \( I_t \in \mathbb{R}^{W \times H \times 3} \) the target image and \( D_t \in \mathbb{R}^{W \times H \times 1} \) target depth. In self-supervised learning, this model is trained via novel view synthesis by warping a set of source frames \( I_i \) to the target frame \( I_t \) using the learned depth \( D_t \) and the target to source pose \( ^*T_i \in SE(3) \). Prior methods assume a static scene observed by a camera undergoing ego-motion. This fundamental assumption is often violated when moving objects are present in the scene. A common solution is to mask the dynamic objects’ pixels. These solutions aim to assist the static scene assumption. In this letter, rather than enforcing the rigid scene restriction, a proposition is made to alleviate this restriction. For each pixel, a global rigid-scene pose and a piece-wise rigid pose for each dynamic object is learned. This is more precise and consistent with the non-rigid real-world situations. An instance segmentation network [23] is extended to incorporate the pose information so that the network learns an additional 6-DOF pose for each instance. Therefore, each instance \( i \) is represented by the class \( c_i \), bounding box \( B_i \), mask \( M^i \) and the additional pose \( ^*T_i \in SE(3) \) as illustrated in Fig. 2. The per-instance warping is defined as:

\[
\hat{p}_i \sim K \sum_{i=0}^{m} [M^i_{p_i}T^i_{p_i} + (1 - M^i_{p_i})I_{4}] ^*T_iD_iK^{-1}p_i \tag{1}
\]
where $m$ is the number of dynamic object instances and $I_4$ a $4 \times 4$ identity matrix. For simplicity the homogeneous pose and projection transformations are omitted in (1). The mask $M^t_i$ is used to transform only the dynamic object $i$ with its pose $T^i_t$. Rigid scene points are transformed only with the pose $^sT^i_t$. Using the (1), the image $\hat{I}_t$ is obtained by inverse warping.

### B. Architecture

In order to explicitly model the motion of the moving objects, an instance pose head is introduced into an instance segmentation network. EfficientPS [23] has demonstrated SOTA results for panoptic and instance segmentation and is therefore adopted in this letter for depth prediction. It consists of the EfficientNet backbone [29], BiFPN [30], the DPC [2] semantic head, the MaskRCNN instance segmentation head [11], the novel instance pose head, an ego-pose head and a depth head. During training, the FPN features ($P_4, P_8, P_{16}, P_{32}$) are extracted for the source $I_t$ and target frames $I_{t-1}, I_{t+1}$. These features are pooled using the proposals of the RPN and the ROI Align modules. The class, bounding box, and instance mask heads use only the features of frame $I_t$. The Instance pose head uses both source and target frames as input. This head output a 6 axis-angle parameters per instance. Similarly, the ego-pose head uses the both source and target frames $P_t$ FPN features as input. This head output a 6 axis-angle parameters for the ego-pose. The depth head input the FPN features of the source frame $I_t$ and output a multi-scale depth.

1) **Instance Pose Head:** The key idea of this letter is to factorize the motion by explicitly estimating the 6–DOF pose of each object in addition to the dominant ego-pose. In order to accurately estimate this motion, the objects should be matched and tracked temporally and its interaction should be modeled. Inspired by the prior work on object tracking [22], [38], a novel instance pose head that extends the instance segmentation is proposed using transformer module [31]. This head makes use of the multi-head attention to learn the association and interaction of the object across time.

The RPN network yields $N$ proposals. The features of each proposal are pooled using a ROI Align module. These features are extracted for the three frames. Therefore, the input of the instance pose head is $b \times (s+1) \times N \times 256 \times 14 \times 14$. Where $b$ is the batch size and $s$ is the number of sources images. The first operation is to project these features into the transformer embedding. The linear projection layer flatten the 3 last dimensions and a linear layer is used to learn an embedding of each proposals. This mapping is defined as $\text{Linearprojection} : \mathbb{R}^{B \times (s+1) \times N \times 512} \rightarrow \mathbb{R}^{B \times (s+1) \times N \times 512}$.

The input of the encoder-decoder transformer is a $(s+1)N$ sequence with 512 features. The transformer-encoder multi-head attention enables the matching of target frame proposals with respect to the source proposals across time, while the feed-forward learns the matched-motion features. For the transformer-decoder, only the target proposals are used for input. The multi-head attention aggregates the matched-motion features of the encoder to the target proposals and further learns the interactions of the objects by learning an attention between the proposals. Finally a linear layer is used to predict the 6–DOF pose per object yielding $B \times N \times s \times 6$ using a 6 axis-angle convention parameters. The non-maximum-suppression used for the object detection head is employed to filter the $N = 1000$ proposals pose keeping only the relevant objects. The object pose is predicted only for the filtered objects.

2) **Ego-Pose Branch:** The ego-pose branch estimates the dominant pose of the camera. The architecture is similar to [9]. Since the low-level features that allow matching are usually extracted in the first layers, the $P_4$ features of the FPN for source
and target features are used. This network outputs 6 parameters for the pose transformation using the axis-angle convention.

3) Depth Branch: The depth branch consists of convolution layers with skip connections from the FPN module as in [9]. Similar to prior work [9], [35], [41], a multi-scale depth is estimated in order to resolve the issue of gradient locality. The prediction of depth at each scale consists of a convolution with a kernel of $1 \times 1$ and a Sigmoid activation. The output of this activation, $\sigma$, is re-scaled to obtain the depth $D = \frac{1}{\alpha \sigma + \beta}$, where $\alpha$ and $\beta$ are chosen to constrain $D$ between 0.5 and 100 units, similar to [9].

To maintain self-supervised learning setting, a frozen pre-trained EfficientPS that was trained on the Cityscapes benchmark [6] is used. This pretrained model achieve $PQ = 50.2$ and $SQ = 76.8$ on Cityscapes test benchmark. As the representation that was trained for panoptic segmentation may ignore details that are crucial for depth prediction. A duplicate of the Backbone and FPN is used for the Depth and pose heads. This allows learning features optimized for depth prediction without degrading the performances of the panoptic segmentation heads.

C. Objective Functions

Let $L$ be the objective function. The self-supervised setting casts the depth learning problem into image reconstruction problem through the reverse warping. Thus, learning the parameters $\theta$ involves learning $\hat{\theta} \in \Theta$ such $\hat{I}_t = I_t$. Learning involves minimizing the objective function:

$$\hat{\theta} = \arg \min_{\theta \in \Theta} \frac{1}{n} \sum_{t} L(f(I_t, \hat{I}_t; \theta))$$

(2)

where $n$ is the number of training examples. The selected surrogate losses that minimize (2) are:

- **Photometric loss**: Following [9], [26], [41] The photometric loss seeks to reconstruct the target image by warping the source images using the static/dynamic pose and depth. An $L_1$ loss is defined as follows:

$$L_{rec}(I_t, \hat{I}_t) = \sum_{p} |I_t(p) - \hat{I}_t(p)|$$

(3)

where $\hat{I}_t(p)$ is the reverse warped target image obtained by (1). This simple $L_1$ is regularized using SSIM [34] that has a similar objective to reconstruct the image. The final photometric loss is defined as:

$$L_{pe}(I_t, \hat{I}_t) = \sum_{p} [(1 - \alpha) \text{SSIM}[I_t(p) - \hat{I}_t(p)]$$

$$+ \alpha |I_t(p) - \hat{I}_t(p)|]$$

(4)

- **Depth smoothness**: An edge-aware gradient smoothness constraint is used to regularize the photometric loss. The disparity map is constrained to be locally smooth through the use an edge-weighted $L_1$ penalty, as discontinuities often occur at image gradients. This regularization is defined as [13]:

$$L_s(D_t) = \sum_{p} [\partial_p D_t(p) + |\partial_p I_t(p)|]$$

(5)

The final objective function is defined as:

$$L = L_{pe} + \alpha_d L_s$$

(6)

IV. EXPERIMENT

A. Setting

- KITTI benchmark [8]: Following the prior work [9], [33], [35], [39], [41], the Eigen et al. [7] split is used with Zhou et al. [41] pre-processing to remove static frames. For evaluation, the metrics of previous works [7] are used for the depth.

- Implementation details: PyTorch [24] is used for all models. The networks are trained for 40 epochs and 20 for the ablation, with a batch size of 2. The Adam optimizer [16] is used with a learning rate of $\beta = 10^{-4}$ and $(\beta_1, \beta_2) = (0.9, 0.999)$. Exponential moving average of the model parameters is used with a decay of $\text{decay} = 0.995$. As the training proceeds, the learning rate is decayed at epoch 15 to $10^{-5}$. The SSIM weight is set to $\alpha = 0.15$ and the smoothing regularization weight to $\alpha_d = 0.001$. The depth head outputs 4 depth maps. At each scale, the depth is up-scaled to the target image size. The hyperparameters of EfficientPS are defined in [23] with $N = 1000$ before the Non-maximum-suppression. Two source images are used $I_{t-1}$ and $I_{t+1}$ are used. The input images are resized to $192 \times 640$. Two data augmentations were performed: horizontal flips with probability $p = 0.5$ and color jitter with $p = 1$.

B. Results

During the evaluation, the depth is capped to 80 m. To resolve the scale ambiguity, the predicted depth map is multiplied by the median scaling. The results are reported in Table I. The proposed method achieves competitive performances compared to the state-of-the-art (SOTA) and outperforms [35] with respect to the $SqRel$ with an improvement of 6.62%. As expected, the proposed method is superior than the prior works that factorize the motion using the optical flow [25], [40] as their estimated motion is only local and do not account for the class and of the object. Besides, it outperforms other similar methods [18] that factorize using the a pose for each object. Fig. 1 illustrates the qualitative result comparison. As observed, the proposed method enables high quality depth prediction. Compared to the SOTA methods, our methods is able to represent well the dynamic objects i.e. the pedestrians that are crossing the street, the biker on the left. As the network did not mask the dynamic objects during training, the dynamic objects are better learned compared to the methods that masks the dynamic objects [26], [35].

1) Dynamic and Static Evaluation: In contrast to training, where the points are categorized into moving and static object’s points, testing is performed on all points that have Lidar ground truth. This does not take into account the relevance of the points and the static/dynamic category. Moving objects are crucial for autonomous driving applications. However, with this testing
setup, it is not possible to convey how the model performs on moving objects, especially for methods that masks moving objects during training. This begs the question as to whether a model trained with rigid scene assumption is still able to predict a depth for moving object (probably due to the depth smoothness regularization and stationary cars) however, its quality is very degraded compared to the static objects.

In order to address this question, the performances of static/dynamic are evaluated separately. A mask of dynamic objects is used to segment moving objects and the assessment results strongly suggest that the performance of the proposed method still yields better results on average with respect to dynamic objects. This suggests that the transformer-based instance pose head is able to temporally match the instances and predict an instance pose. It should be noted, however, that the Insta-DM has matched binary instance masks implicitly learnt in the network. Even so, the proposed method benchmark is biased towards static scene. In order to unbiased the evaluation, the mean per-category is used to balance the influence. The proposed method outperform the video-to-depth prediction method [35] with $\Delta SqRel = -0.344$ m. The analysis of Table II and Fig. 1 suggests that models with rigid scene assumption is still able to predict a depth for moving object (probably due to the depth smoothness regularization and stationary cars) however, its quality is very degraded compared to the static objects.

Moreover, the results reported in Table II show that the proposed method outperforms Insta-DM [18] with respect to both the static and dynamic objects. Insta-DM [18] proposes an Obj-PoseNet $O_{\psi} : \mathbb{R}^{2 \times H \times W \times 3} \rightarrow \mathbb{R}^{6}$ that takes per-object matched binary instance masks ($M_1$, $M_2$) and outputs the object pose. It should be noted, however, that the Insta-DM has an unfair advantage since object matching (via binary masks) is provided as input in a supervised learning approach while the proposed method is self-supervised with matching being implicitly learnt in the network. Even so, the proposed method still yields better results on average with respect to dynamic objects. This suggests that the transformer-based instance pose head is able to temporally match the instances and predict an accurate pose.

2) Ablation Study: Table III illustrate an ablation study performed to validate the contribution of the proposed method. The results strongly suggest that the performance of the proposed network is mainly obtained by the introduction of the motion factorization through the proposed instance pose head.

- **A1 versus A4**: Introducing a more complex architecture did not contribute to the improvement of the performances.
- **A4 versus A5**: Sharing the backbone for the depth network did not contribute to the improvement of the performances. However, it did reduce the complexity of the network.
- **A5 versus A6**: Introducing the piece-wise rigid pose warping induces an improvement of $\Delta Sqrel = 14.1\%$
- **A2 versus A3 versus A4**: The pose head is sensitive to the choice of the features level. $P_3$ is the optimal level for this application.

This results suggest that not only the models learn an accurate depth but also accurate instance pose. This result demonstrates
TABLE III

AN ABLATION STUDY OF THE PROPOSED METHOD. THE EVALUATION WAS DONE ON KITTI BENCHMARK USING EIGEN SPLIT [7]. AS OBSERVED, THE EFFECT OF THE BACKBONE IS MINIMAL A1 VS A5, THE CHOICE OF THE INPUT FEATURE FOR EGO-POSE HEAD IS SENSIBLE A2 VS A3 VS A4, THE PERFORMANCE OF THE PROPOSED METHOD IS OBTAINED MAINLY BY THE INTRODUCTION OF THE PIECE-WISE RIGID POSE A5 VS A6. INCREASING THE COMPLEXITY OF THE MODEL ALLOW BETTER PERFORMANCES AND BETTER TRAINING STABILITY A6 VS A7

| Ablation | Backbone | Ego-pose input feature | Shared backbone | Piece-wise rigid pose | Abs Rel | Sq Rel | RMSE | RMSE log |
|----------|----------|------------------------|----------------|----------------------|---------|--------|------|---------|
| A1       | ResNet18 [13] | Layer5 | | | 0.121 | 0.914 | 4.890 | 0.196 |
| A2       | EfficientNet-b5 | $P_{b6}$ | - | - | 0.132 | 0.906 | 4.981 | 0.205 |
| A3       | EfficientNet-b5 | $P_{b6}$ | - | - | 0.127 | 0.933 | 5.040 | 0.201 |
| A4       | EfficientNet-b5 | $P_{b6}$ | - | - | 0.121 | 0.894 | 4.886 | 0.197 |
| A5       | EfficientNet-b5 | $P_{b6}$ | - | - | 0.120 | 0.925 | 4.880 | 0.194 |
| A6       | EfficientNet-b5 | $P_{b6}$ | ✓ | ✓ | 0.113 | 0.795 | 4.689 | 0.190 |
| A7       | EfficientNet-b6 | $P_{b6}$ | ✓ | ✓ | 0.110 | 0.719 | 4.486 | 0.184 |

Fig. 3. Qualitative results of the proposed method with SOTA methods [26], [35]. (a) The proposed method is on par with the baselines for static objects. (e) and (f) show cars as moving objects. Although the baselines are trained with auto-masking, the dataset is rich with static cars that are not masked during training, this provides clues to learn the depth for moving cars. (d-g) show more complex situations as pedestrians and bicycles tend to always move in the KITTI dataset. The qualitative results show that the proposed method outperforms the baselines. This is validated further by the quantitative results reported in Table II.

that the transformer network is able to match and learn the interaction of the objects across time. The model in A5 is on the same setting of the other SOTA methods [26], [35]. Despite using this low performance baseline, the introduction of the dynamic warping enabled the proposed method to achieve the SOTA results.

An interesting observation during training is that A6 under-fits the data (i.e., the validation loss is less than the learning loss). The test performances are not stable, the best model among the 20 epochs is reported for this backbone. In order to resolve this under-fitting, the complexity of the model is increased A7. This allow for a better stability of the training loss and test performance. The best results are obtained using this complexity. The additional instance pose results in an additional run-time overhead during training. The training time for 1 epoch for A5 and A7 is 233 mn and 58 mn trained on RTX3090 respectively. However, the additional run-time is only for the training. At test-time, the depth network require only a single pass of the image $I_t$ with roughly 34FPS for A7 model and 38FPS for A6 model using a single RTX3090.

V. CONCLUSION

In this letter a novel instance pose head is introduced for self-supervising monocular depth prediction. This head enables the factorization of the scene’s motion. Thus, alleviating the rigid scene assumption. It is shown that it achieves the SOTA results on the KITTI benchmark [8]. The ablation study further validates that the multi-head attention of the transformer network predicts an accurate object pose. Moreover, the impact of the dynamic motion on this benchmark is exposed. Namely, the bias towards static objects where 86.43% of the test pixels correspond to static objects. A mean per static/dynamic category metric is proposed to unbias the assessment.

REFERENCES

[1] J. Bian et al., “Unsupervised scale-consistent depth and ego-motion learning from monocular video,” in Proc. Adv. Neural Inf. Process. Syst., 2019, pp. 35–45.
[2] L.-C. Chen et al., “Searching for efficient multi-scale architectures for dense image prediction,” in Proc. Adv. Neural Inf. Process. Syst., 2018, pp. 8713–8724.
[3] P.-Y. Chen, A. H. Liu, Y.-C. Liu, and Y.-C. F. Wang, “Towards scene understanding: Unsupervised monocular depth estimation with semantic-aware representation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 2624–2632.

[4] Y. Chen, C. Schmid, and C. Sminchisescu, “Self-supervised learning with geometric constraints in monocular video: Connecting flow, depth, and camera,” in Proc. IEEE Int. Conf. Comput. Vis., 2019, pp. 7062–7071.

[5] J. Choi, D. Jung, D. Lee, and C. Kim, “SAFENet: Self-supervised monocular depth estimation with semantic-aware feature extraction,” in Proc. 34th Conf. Neural Inf. Process. Syst., NeurIPS, 2020.

[6] M. Cordts et al., “The cityscapes dataset for semantic urban scene understanding,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 3213–3223.

[7] D. Eigen, C. Puhrsch, and R. Fergus, “Depth map prediction from a single image using a multi-scale deep network,” in Proc. Adv. Neural Inf. Process. Syst., 2014, pp. 2366–2374.

[8] A. Geiger, P. Lenz, and R. Urtasun, “Are we ready for autonomous driving? The KITTI vision benchmark suite,” in Proc. Conf. Comput. Vis. Pattern Recognit., 2012, pp. 3354–3361.

[9] C. Godard, O. M. Aodha, M. Firman, and G. Brostow, “Digging into videos in the wild: Unsupervised monocular depth learning from unknown cameras,” in Proc. IEEE Int. Conf. Comput. Vis., 2019, pp. 3627–3637.

[10] A. Gordon, H. Li, R. Jonschkowski, and A. Angelova, “Depth from videos in the wild: Unsupervised monocular depth learning from unknown cameras,” in Proc. IEEE Int. Conf. Comput. Vis., 2019, pp. 8976–8985.

[11] R. Mohan and A. Valada, “EfficientPS: Efficient panoptic segmentation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 10781–10790.

[12] A. Vaswani et al., “Attention is all you need,” in Proc. Adv. Neural Inf. Process. Syst., 2017.

[13] S. Vijayanarasimhan, S. Ricco, C. Schmid, R. Sukthankar, and K. Fragkiadaki, “SFM-Net: Learning of structure and motion from video,” 2017, arXiv:1704.07804.

[14] J. Hur and S. Roth, “Self-supervised monocular scene flow estimation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 7396–7405.

[15] Z. Yang, P. Wang, W. Xu, L. Zhao, and R. Nevatia, “Unsupervised learning of depth and egomotion,” in Proc. European Conf. Comput. Vis., Springer, Cham, 2020.

[16] M. Tan and Q. Le, “EfficientDet: Rethinking model scaling for convolutional neural networks,” in Proc. Int. Conf. Mach. Learn., 2019, pp. 6105–6114.

[17] Z. Wang, Y. Wang, L. Wang, Y. Zhan, Y. Wang, and H. Lu, “Can scale-consistent monocular depth be learned in a self-supervised scale-invariant manner?” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2021, pp. 12727–12736.

[18] S. Vijayanarasimhan, S. Ricco, C. Schmid, R. Sukthankar, and K. Fragkiadaki, “SFM-Net: Learning of structure and motion from video,” 2017, arXiv:1704.07804.

[19] J. Watson, O. M. Aodha, V. Prisacariu, G. Brostow, and M. Firman, “The temporal opportunist: Self-supervised multi-frame monocular depth,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2021, pp. 1164–1174.

[20] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, “Unsupervised learning of depth and ego-motion from video,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 1851–1858.

[21] R. Mohan and A. Valada, “EfficientPS: Efficient panoptic segmentation,” Int. J. Comput. Vis., vol. 129, no. 5, pp. 1551–1579, 2021.

[22] A. Paszke et al., “Pytorch: An imperative style, high-performance deep learning library,” in Proc. Adv. Neural Inf. Process. Syst., 2019, pp. 8024–8035.

[23] A. Gal et al., “Competitive collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., 2019, pp. 12232–12241.

[24] V. Guizilini, R. Ambrus, S. Pillai, A. Raventos, and A. Gaidon, “3D packing for self-supervised monocular depth estimation,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., 2020, pp. 2482–2491.

[25] Z. Yin and J. Shi, “GeoNet: Unsupervised learning of dense depth, optical flow and camera pose,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., 2019, pp. 1164–1174.

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

[27] P. D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014, arXiv:1412.6980.

[28] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, “Detectron2,” 2019.

[29] S. Vijayanarasimhan, S. Ricco, C. Schmid, R. Sukthankar, and K. Fragkiadaki, “SFM-Net: Learning of structure and motion from video,” 2017, arXiv:1704.07804.

[30] Y. Xu, A. Vedaldi, and J. F. Henriques, “Moving SLAM: Fully unsupervised deep learning in non-rigid scenes,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., 2021, pp. 4611–4617.

[31] Y. Xu, Y. Ban, G. Delorme, C. Gan, D. Rus, and X. Alameda-Pineda, “TransCenter: Transformers with dense queries for multiple-object tracking,” 2021, arXiv:2103.15145.

[32] Z. Yang, P. Wang, W. Xu, L. Zhao, and R. Nevatia, “Unsupervised learning of geometry from videos with edge-aware depth-normal consistency,” in Proc. 32nd AAAI Conf. Artif. Intell., 2018, pp. 7493–7500.

[33] Z. Yin and J. Shi, “GeoNet: Unsupervised learning of dense depth, optical flow and camera pose,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., 2018, pp. 1983–1992.

[34] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, “Unsupervised learning of depth and ego-motion from video,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 8844–8854.