ISSN 1060-992X, Optical Memory and Neural Networks, 2023, Vol. 32, Suppl. 2, pp. S206–S213. © The Author(s), 2023. This article is an open access publication.

Influence of Neural Network Receptive Field on Monocular Depth and Ego-Motion Estimation

S. A. Linok[a, *] and D. A. Yudin[a, b]

[a] *Moscow Institute of Physics and Technology, Moscow, Moscow Oblast, 141701 Russia*

[b] *AIRI (Artificial Intelligence Research Institute), Moscow, Russia*

**e-mail: linok.sa@phystech.edu*

Received August 15, 2023; revised September 1, 2023; accepted September 5, 2023

Abstract—We present an analysis of a self-supervised learning approach for monocular depth and ego-motion estimation. This is an important problem for computer vision systems of robots, autonomous vehicles and other intelligent agents, equipped only with monocular camera sensor. We have explored a number of neural network architectures that perform single-frame depth and multi-frame camera pose predictions to minimize photometric error between consecutive frames on a sequence of camera images. Unlike other existing works, our proposed approach called ERF-SfMLearner examines the influence of the deep neural network receptive field on the performance of depth and ego-motion estimation. To do this, we study the modification of network layers with two convolution operators with extended receptive field: dilated and deformable convolutions. We demonstrate on the KITTI dataset that increasing the receptive field leads to better metrics and lower errors both in terms of depth and ego-motion estimation. Code is publicly available at github.com/linukc/ERF-SfMLearner.

Keywords: self-supervised learning, monocular ego-motion estimation, monocular depth estimation

DOI: 10.3103/S1060992X23060103

1. INTRODUCTION

Today, deep neural networks are the most popular tool in autonomous systems development. They can be effective prediction models based on different input sensor data [1]. We are especially interested in neural depth and ego-pose estimation for the onboard monocular camera. This is essential both for the tasks of detecting and tracking three-dimensional objects [2] and for high-quality mapping of the area in which the camera is moving [3]. Also, monocular camera depth estimation is much more complicated than the reconstruction of depth maps from a stereo pair of images [4, 5], where we can accurately estimate the scale and pixel disparity both analytically and using neural networks [6].

Existing results [7, 8] show that neural networks can successfully cope with this problem on one level with feature-based methods. However, they are still significantly affected by noise in the reconstructed depth maps.

For each specific task it is also necessary to have a large data set. To compile such a set for the ego-motion estimation, it is necessary to have accurate equipment for taking ground truth (GT) values. It is not always possible, especially if you plan to test the performance of the algorithm with your data. Therefore, we chose an joint self-supervised learning approach [7] as the main algorithm, which does not require pre-labeling for training, but uses an additional single-frame depth and multi-frame pose predictions to minimize the photometric error.

The process of choosing neural network architectures is empirical since there are no mathematically proved rules for their compilation. Different models can solve the same problem with a big difference in the final metrics. Our approach builds upon the insight that the receptive field is an important hyperparameter that greatly affects the ability of neural networks to perform a task. Experiments on the KITTI [9] dataset show the trueness of this assumption for both pose and depth prediction.

In summary, our work makes the following main contributions:

• a novel convolutional neural approach called ERF-SfMLearner for monocular depth and ego-motion estimation with extended receptive field;

• analysis of neural network receptive filed influence on monocular depth and ego-motion estimation on KITTI dataset with different resolutions of the input image.

2. RELATED WORK

Most self-supervised methods use single-frame depth and multi-frame pose predictions to minimize photometric error from source to a target frame from the sequence of images. This idea was first introduced by Zhou et al. [7]. Based on this principle a lot of works were proposed. Mahjourian et al. [10] offer an approach to combine photometric loss with geometric constraints using 3D-based loss. Godard et al. [11] in Monodepth2 enhance reprojection loss and design it to robustly handle occlusions. Subsequent methods propose improvements based on various techniques, such as supervision from optical flow [12–16], semantic segmentation [17, 18] or combination of these techniques [19, 20]. Wang et at. [21] propose to synthesize new views from raw images, thereby enriching the training data and improving the performance of the pose network. Tak-Wai Hui in [22] rethink the utilization of image sequence in the RNN acitecture. Suri et at. in [23] introduce pose constraints to reduce depth inconsistencies and scale ambiguity. Lee et at. in [24] suggest to integrate IMU sensor to disambiguate depth scale.

However, most of these methods inherit learning setup and neural architectures from [7]. Our analysis shows that they can be more improved if they leverage our findings on the importance of the neural network receptive field in the task of self-supervised monocular depth and ego-motion estimation.

3. METHOD

3.1. ERF-SfMLearner Architecture

In a deep learning context, receptive field (RF) is defined as the size of the region in the input that produces the feature. Basically, it is a measure of association of an output feature (of any layer) to the input region (patch). Specifically, for self-supervised pose and depth estimation, we would like each output feature of the encoder to have a big receptive field, so as to ensure that no crucial information was not taken into account [25]. We establish a strong baseline for our algorithm by following practices from this work [7]. In baseline method, neural networks are implemented as a convolution network (Fig. 1, top). So, for a more fair comparison, we choose two convolution operators type to effectively increase the receptive field of the neural network without global architecture redesign: dilated convolution and deformable convolution (Figs. 2, 3).

3.2. Learning Approach

An overview of the baseline approach is shown in Fig. 1. It can learn depth and camera motion from unlabeled data. The method consists of two parts: depth prediction network and pose estimation network, which are trained jointly.

For two adjacent frames, I_t and I_s, if the depth map of I_t and the relative pose between the two views are given, then $I_{s'}$ view can be reconstructed from I_t. Taking I_t as input, depth prediction network generates depth map, denoted as D_t. The relative camera pose between two views can be estimated from the pose estimation network, denoted as $T_{t\to s}$. Denote p_t as the homogeneous coordinates of a pixel in I_t and $p_{s'}$ as the corresponding pixel in $I_{s'}$. The projected coordinates then can be expressed as:

$$p_{s'} \sim KT_{t\to s}D_tK^{-1}p_t, \tag{1}$$

where K is the camera intrinsic matrix, $T_{t\to s}$ is the camera coordinate transformation matrix from the I_t frame to the I_s frame, D_t is the depth value of the p_t pixel in the I_t frame, and the coordinates are homogeneous.

The loss function includes photometric, smooth and regularization losses [7].

3.3. Receptive Field Extension with Dilated Convolution

Dilated convolution is a very similar to a basic convolution operator. In essence, dilated convolutions introduce another parameter, denoted as r, called the dilation rate. Dilations incorporate holes in a convolutional kernel [26]. The "holes" basically define a spacing between the values of the kernel. So, while the number of weights in the kernel is unchanged, the weights are no longer applied to spatially adjacent samples.

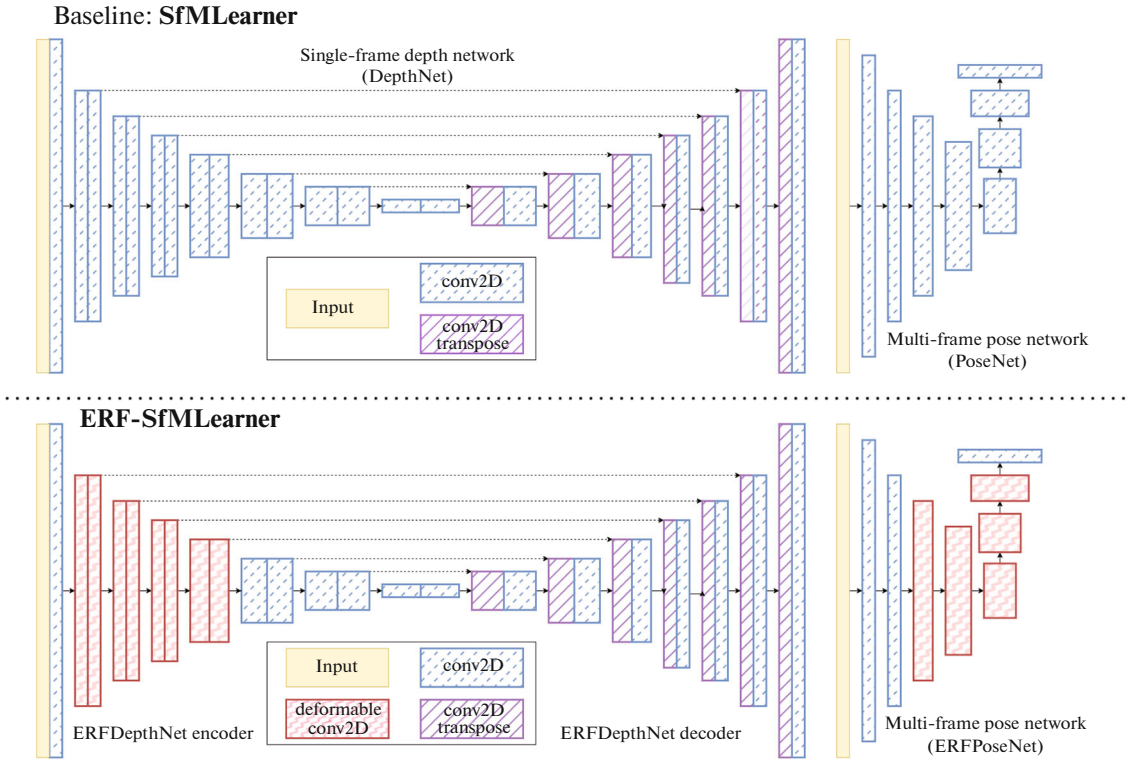

Fig. 1. Baseline network architecture (SfMLearner) and ERF-SfMLearner (extended receptive field SfMLearner). Each rectangular block indicates the output channels after convolution operation. DepthNet has "U-net" like architecture with multi-scale side predictions. The kernel size is 3 for all the layers except for the first 4 conv layers with 7, 7, 5, 5, respectively. PoseNet predicts 6-DoF relative pose. Kernel size is 3 for all the layers except for the first two conv where we use kernel 7 and 5, respectively. ERFDepthNet encoder shares the same architecture with baseline besides 4 first blocks of deformable convolutions (DFC ERFDepthNet, Fig. 3a). In ERFPoseNet 4 blocks of deformable convolution place at the end of the encoder (DFC ERFPoseNet, Fig. 2c).

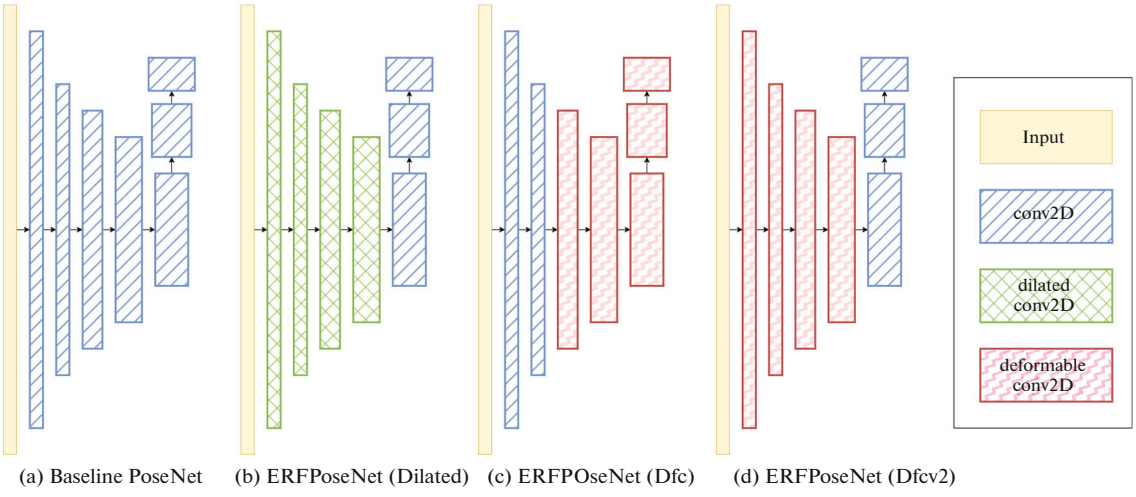

Fig. 2. Network architectures for ERFPoseNet. (a) PoseNet from baseline paper. (b–d) Different ERFPoseNets with extended RF.

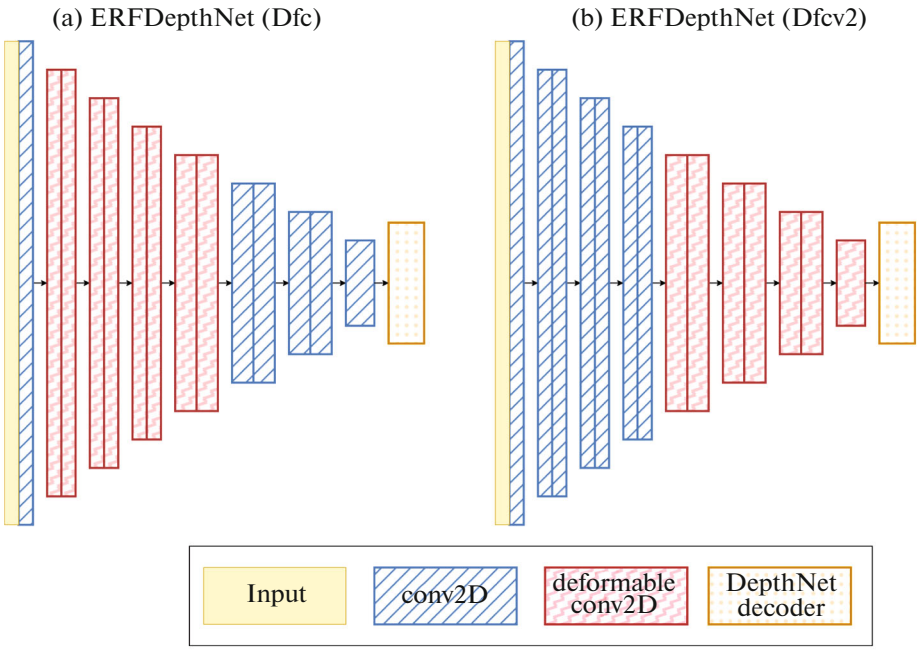

Fig. 3. Different ERFDepthNet encoder's architectures.

3.4. Receptive Field Extension with Deformable Convolution

Deformable convolution adds 2D offsets to the regular grid sampling locations in the standard convolution. It enables free form deformation of the sampling grid. The offsets are learned from the preceding feature maps, via additional convolutional layers. Thus, the deformation is conditioned on the input features in a local, dense, and adaptive manner [27]. In our work we also examine second version of deformable convolution: each sample not only undergoes a learned offset, but is also modulated by a learned feature amplitude. The network module is thus given the ability to vary both the spatial distribution and the relative influence of its samples [28].

4. EXPERIMENTS

We evaluate the performance of our methods and compare them with baseline's approach on multi-frame ego-motion estimation and single-view depth as well. We use the KITTI dataset [9] for benchmarking.

Dataset. We use monocular image sequences for training and test. The original image size is 375 × 1242, and images are downsampled during training. In order to compare fairly with baseline, we use two different splits of the KITTI dataset: KITTI Odometry[1] to train networks and evaluate ERFPoseNet and Kitti Eigen split[2] to test ERFDepthNet. We train models on KITTI Odometry sequence 00–08 and evaluate the pose error on sequence 09 and 10.

Training details. For all the experiments we set epoch-size = 1000, sequence-length = 5, photo-loss-weight = 1, mask-loss-weight = 0, smooth-loss-weight = 0.2. During training, we used batch normalization for all the layers except for the output layers, and the Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, learning rate of 0.0002 and mini-batch size of 4. More details can be found in our repository.[3]

4.1. Pose Estimation

To evaluate the impact of RF on ego-motion prediction we use different ERFPoseNet's architectures as visualized in Fig. 2 and jointly train them with the DepthNet from the baseline. To resolve scale ambi-

[1] https://www.cvlibs.net/datasets/kitti/eval_odometry.php.
[2] https://github.com/tinghuiz/SfMLearner/tree/master/data/kitti.
[3] github.com/linukc/ERF-SfMLearner.

Table 1. Absolute Trajectory Error and Rotation Error on the KITTI Odometry split averaged over all 5 frame-snippets (lower is better)

Method (image resolution)	Seq. 09				Seq. 10			
	ATE_{mean}	ATE_{std}	RE_{mean}	RE_{std}	ATE_{mean}	ATE_{std}	RE_{mean}	RE_{std}
PoseNet 248 × 75	0.04	0.0419	0.0058	0.0035	0.0218	0.0147	0.0052	0.0035
PoseNet 310 × 94	0.0272	0.0251	0.0054	0.0033	0.0162	0.0105	0.0052	0.0034
PoseNet 416 × 128	0.021	0.0157	0.0048	0.0029	0.0145	0.009	0.0047	0.0034
ERFPoseNet (Dilated) 416 × 128	0.0187	0.0147	0.0048	0.003	0.0141	0.0092	0.0049	0.004
ERFPoseNet (Dfc) 416 × 128	0.018	0.0124	**0.0042**	0.0027	0.0135	0.0094	0.0043	0.0036
ERFPoseNet (Dfcv2) 416 × 128	0.0206	0.0138	0.0049	0.0028	0.0143	0.0091	0.0048	0.0035
PoseNet 620 × 188	0.0186	0.0117	0.0048	0.0036	0.0137	**0.0086**	0.0049	0.0042
ERFPoseNet (Dilated) 620 × 188	0.0182	0.0108	0.0048	0.0033	**0.0132**	0.0088	0.0047	**0.0033**
ERFPoseNet (Dfc) 620 × 188	**0.0165**	0.0087	0.0043	**0.0024**	**0.0132**	0.0095	**0.0042**	0.0036
ERFPoseNet (Dfcv2) 620 × 188	0.0173	0.0102	0.0048	0.0029	0.0136	0.0096	0.005	0.0045
PoseNet 1241 × 376	**0.0165**	**0.008**	0.0056	0.005	0.0148	0.0095	0.0056	0.0054

guity during evaluation, we first optimize the scaling factor for the predictions made by each method to best align with the ground truth, and then measure the Absolute Trajectory Error (ATE) and Rotation Error (RE) as the metrics (Table 1). RE between R_1 and R_2 is defined as the angle of $R_1 R_2^{-1}$ when converted to axis/angle:

$$RE = arccos\left((trace(R_1 R_2^{-1}) - 1)/2\right). \quad (2)$$

As shown in Table 1, RF increase helps to get lower errors and better metrics in ego-motion estimation. Moreover, dilated convolution slightly improves the metrics, but the use of deformable convolution for the last layers of the ERFPoseNet is a much more profitable method. As shown in Fig. 4, RF with deformable convolution covers the entire input, which could explain this result. On the other hand, applying deformable convolution to the four first layers of the ERFPoseNet (Dfcv2) leads to results worse than ERFPoseNet (Dilated) and similar to ERFPoseNet (Dfc).

4.2. Depth Estimation

To evaluate the impact of RF on depth prediction we also used ERFDepthNet architectures, shown in Fig. 3. We study changing convolution operations only in the encoder block and replacing them with the deformable convolutions (Table 2). Since the depth predicted by method is defined up to scale factor, for

Fig. 4. The receptive field (RF) of ERFPoseNet. Top, left to right: original image from KITTI dataset, RF for baseline PoseNet. Bottom, left to right: RF for ERFPoseNet (Dilated), RF for ERFPoseNet (Dfc) and ERFPoseNet (Dfcv2). We use backprop to compute the RF and exploit the fact that the values of the weights of the network are not relevant for computing RF. We change the weight for every layer to be 0.05 and the bias to be 0. To create a situation in which the gradient at the output of the model depends only on the location of the pixels a white image is passed into the network. For visualization, we only compute RF for the one pixel in the first channel of the penultimate conv layer—set the corresponding gradient value to 1 and all the others to 0. When we backpropagate this gradient to the input layer and light up the RF as a red mask.

Table 2. Results for depth estimation on Eigen KITTI split: ERFDepthNets + PoseNet architectures. The errors are only computed where the depth is less than 80 m

Method (image resolution)	Scale		Error metric			Accuracy metric		
	PoseNet	GT	Abs Rel	Sq Rel	RMSE	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
ERFDepthNet (Dfc) 416 × 128		✓	0.1988	1.8269	6.6759	0.7091	0.8866	0.953
ERFDepthNet (Dfcv2) 416 × 128		✓	0.214	2.1015	6.8433	0.6805	0.8811	0.9503
ERFDepthNet (Dfc) 416 × 128	✓		0.3097	3.9491	7.7958	0.55	0.7944	0.8982
ERFDepthNet (Dfcv2) 416 × 128	✓		0.3303	4.6615	8.1794	0.5344	0.7709	0.8797
ERFDepthNet (Dfc) 620 × 188		✓	**0.1927**	1.8134	**6.3779**	**0.7334**	**0.9082**	**0.9639**
ERFDepthNet(Dfc) 620 × 188	✓		0.2832	3.2512	7.5411	0.5665	0.805	0.9015

Table 3. Results for depth estimation on Eigen KITTI split: DepthNets + ERFPoseNets architectures. The errors are only computed where the depth is less than 80 m

Method (image resolution)	Scale		Error metric			Accuracy metric		
	PoseNet	GT	Abs Rel	Sq Rel	RMSE	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
DepthNet + PoseNet 248 × 75		✓	0.2375	2.2019	7.4432	0.6201	0.8539	0.9373
DepthNet + PoseNet 310 × 94		✓	0.2215	2.0839	7.1324	0.656	0.8726	0.9463
DepthNet + PoseNet 416 × 128		✓	0.2135	1.8977	6.7938	0.6789	0.875	0.946
DepthNet + ERFPoseNet (Dilated) 416 × 128		✓	0.2041	**1.7651**	6.736	0.6922	0.8872	0.9539
DepthNet + ERFPoseNet (Dfc) 416 × 128		✓	0.2017	1.8202	6.7059	0.704	0.8902	0.9529
DepthNet + ERFPoseNet (Dfcv2) 416 × 128		✓	0.2083	2.0195	6.8569	0.695	0.884	0.9506
DepthNet + PoseNet 416 × 128	✓		0.354	5.0006	8.3413	0.5026	0.7519	0.8628
DepthNet + ERFPoseNet (Dilated) 416 × 128	✓		0.3305	3.6844	8.1672	0.4968	0.7378	0.8509
DepthNet + ERFPoseNet (Dfc) 416 × 128	✓		0.3132	4.7717	7.8997	0.5478	0.7907	0.8957
DepthNet + ERFPoseNet (Dfcv2) 416 × 128	✓		0.3123	4.2249	8.0961	0.5371	0.78	0.8903
DepthNet + PoseNet 620 × 188		✓	0.2034	2.1613	6.7045	0.7147	0.8929	0.9529
DepthNet + ERFPoseNet (Dilated) 620 × 188		✓	0.1977	1.9366	6.5297	0.7185	0.8948	0.9557
DepthNet + ERFPoseNet (Dfc) 620 × 188		✓	0.2125	2.7657	6.8724	0.7048	0.8874	0.948
ERFDepthNet (DFC) + ERF-PoseNet (Dfc) 620 × 188		✓	**0.1928**	1.8521	**6.4198**	**0.7425**	**0.8983**	**0.9631**
DepthNet + PoseNet 620 × 188	✓		0.3162	4.3905	7.9923	0.5455	0.7783	0.8821
DepthNet + ERFPoseNet (Dilated) 620 × 188	✓		0.3083	3.8333	7.8461	0.5499	0.7744	0.8769
DepthNet + ERFPoseNet (Dfc) 620 × 188	✓		0.2972	4.5935	7.9023	0.5773	0.8033	0.8994
ERFDepthNet (DFC) + ERF-PoseNet (Dfc) 620 × 188	✓		0.2865	3.4243	7.6243	0.5863	0.8092	0.9001
DepthNet + PoseNet 1241 × 376		✓	0.2202	3.0018	7.183	0.6976	0.8897	0.9515

evaluation we multiply the predicted depth maps by a scalar s that matches the median with the ground-truth, i.e. $s = median(D_{gt})/median(D_{pred})$. This we call GT supervisor. As a result, lower errors and best accuracy show deformable convolution, applied to the first four DepthNet layers in ERFDepthNet(Dfc). Also we compare different baseline's DepthNet with ERFPoseNets (Table 3). With joint training, increasing of RF in the original PoseNet gives positive effect on DepthNet predictions, producing depth metrics comparable with ERFDepthNet (Dfc).

5. CONCLUSIONS

We present ERF-SfMLearner, a result of the analysis of receptive field importance in self-supervised deep learning method for monocular depth prediction and camera ego-motion estimation tasks. The experimental evaluation on KITTI dataset shows that bigger receptive field may be a one key to the successful solution of this task. The best result that we have been able to achieve is an increase of receptive field through the use of deformable convolution both for ERFPoseNet and ERFDepthNet models. Also, our work highlights an important fact: in joint training with self-supervised loss changing the architecture of one neural module can affect another module's result. With this knowledge, more advanced neural architectures can be proposed to better cope with the task of the monocular depth and ego-motion estimation and, as a consequence, with a high-quality mapping and better localization.

FUNDING

This work was partially supported by the Analytical Center for the Government of the Russian Federation in accordance with the subsidy agreement, agreement identifier 000000D730321P5Q 0002; grant no. 70-2021-00138.

CONFLICT OF INTEREST

The authors of this work declare that they have no conflicts of interest.

REFERENCES

1. Qusay Sellat and Kanagachidambaresan Ramasubramanian, Advanced techniques for perception and localization in autonomous driving systems: A survey, *Opt. Mem. Neural Networks,* 2022, vol. 31, no. 2, pp. 123–144.
2. Shepel, I., Adeshkin, V., Belkin, I., and Yudin, D.A., Occupancy grid generation with dynamic obstacle segmentation in stereo images, *IEEE Trans. Intell. Transp. Syst.,* 2021, vol. 23, no. 9, pp. 14779–14789.
3. Bokovoy, A., Muraviev, K., and Yakovlev, K., Map-merging algorithms for visual slam: Feasibility study and empirical evaluation, in *Russian Conference on Artificial Intelligence,* Springer, 2020, pp. 46–60.
4. Angermann, Ch., Schwab, M., and Haltmeier, M., Laubichler, Ch., and J'onsson, S., Unsupervised single-shot depth estimation using perceptual reconstruction, *Mach. Vision Appl.,* 2023, vol. 34, no. 5, p. 82.
5. Goshin, Y., Coplanarity-based approach for camera motion estimation invariant to the scene depth, *Opt. Mem. Neural Networks,* 2022, vol. 31 (Suppl. 1), pp. 22–30.
6. Kasatkin, N. and Yudin, D., Real-time approach to neural network-based disparity map generation from stereo images, in *International Conference on Neuroinformatics,* Springer, 2021, pp. 261–268.
7. Tinghui Zhou, Brown, M., Noah Snavely, and Lowe, D.G., Unsupervised learning of depth and ego-motion from video, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* 2017, pp. 1851–1858.
8. Muravyev, K., Bokovoy, A., and Yakovlev, K., tx2_fcnn_node: An open-source ros compatible tool for monocular depth reconstruction, *SoftwareX*, 2022, vol. 17, 100956.
9. Geiger, A., Lenz, Ph., and Urtasun, R., Are we ready for autonomous driving? The kitti vision benchmark suite, in *2012 IEEE Conference on Computer Vision and Pattern Recognition,* IEEE, 2012, pp. 3354–3361.
10. Mahjourian, R., Wicke, M., and Angelova, A., Unsupervised learning of depth and egomotion from monocular video using 3d geometric constraints, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* 2018, pp. 5667–5675.
11. Godard, C., Oisin Mac Aodha, Firman, M., and Brostow, G.J., Digging into selfsupervised monocular depth estimation, in *Proceedings of the IEEE/CVF International Conference on Computer Vision,* 2019, pp. 3828–3838.
12. 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,* 2018, pp. 1983–1992.

13. 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, *IEEE Trans. Pattern Anal. Mach. Intell.,* 2019, vol. 42, no. 10, pp. 2624–2641.
14. Anurag Ranjan, Varun Jampani, Balles, L., Kihwan Kim, Deqing Sun, Wulff, J., and Black, M.J., Competitive collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* 2019, pp. 12240–12249.
15. Yuhua Chen, Cordelia Schmid, and Cristian Sminchisescu, Self-supervised learning with geometric constraints in monocular video: Connecting flow, depth, and camera, in *Proceedings of the IEEE/CVF International Conference on Computer Vision,* 2019, pp. 7063–7072.
16. Baigan Zhao, Yingping Huang, Wenyan Ci, and Xing Hu, Unsupervised learning of monocular depth and ego-motion with optical flow features and multiple constraints, *Sensors,* 2022, vol. 22, no. 4, p. 1383.
17. Xiaobin Wei, Jianjiang Feng, and Jie Zhou, Semantics-driven unsupervised learning for monocular depth and ego-motion estimation. arXiv preprint arXiv:2006.04371, 2020.
18. Jaehoon Choi, Dongki Jung, Donghwan Lee, and Changick Kim, Safenet: Self-supervised monocular depth estimation with semantic-aware feature extraction. arXiv preprint arXiv:2010.02893, 2020.
19. Fabio Tosi, Filippo Aleotti, Pierluigi Zama Ramirez, Matteo Poggi, Samuele Salti, Luigi Di Stefano, and Stefano Mattoccia, Distilled semantics for comprehensive scene understanding from videos, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* 2020, pp. 4654–4665.
20. Vertens, J. and Burgard, W., Usegscene: Unsupervised learning of depth, optical flow and ego-motion with semantic guidance and coupled networks. arXiv preprint arXiv:2207.07469, 2022.
21. Guangming Wang, Jiquan Zhong, Shijie Zhao, Wenhua Wu, Zhe Liu, and Hesheng Wang, 3d hierarchical refinement and augmentation for unsupervised learning of depth and pose from monocular video, *IEEE Trans. Circuits Syst. Video Technol.,* 2022, vol. 33, no. 4, pp. 1776–1786.
22. Tak-Wai Hui, Rm-depth: Unsupervised learning of recurrent monocular depth in dynamic scenes, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* 2022, pp. 1675–1684.
23. Zeeshan Khan Suri, Pose constraints for consistent self-supervised monocular depth and egomotion, in *Scandinavian Conference on Image Analysis,* Springer, 2023, pp. 340–353.
24. Chungkeun Lee, Changhyeon Kim, Pyojin Kim, Hyeonbeom Lee, and H. Jin Kim, Scale-aware visual-inertial depth estimation and odometry using monocular self-supervised learning, *IEEE Access,* 2023, vol. 11, pp. 24087–24102.
25. Adaloglou, N., Understanding the receptive field of deep convolutional networks, *AI Summer,* 2020.
26. Andr'e Araujo, Wade Norris, and Jack Sim, Computing receptive fields of convolutional neural networks, *Distill,* 2019, vol. 4, no. 11, p. e21.
27. Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei, Deformable convolutional networks, in *Proceedings of the IEEE International Conference on Computer Vision,* 2017, pp. 764–773.
28. Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai, Deformable convnets v2: More deformable, better results, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* 2019, pp. 9308–9316.
29. Basharov, I. and Yudin, D, Real-time deep neural networks for multiple object tracking and segmentation on monocular video, *Int. Arch. Photogramm., Remote Sens. Spat. Inf. Sci.*, 2021, vol. 44, pp. 15–20.