Deep Global-Relative Networks for End-to-End 6-DoF Visual Localization and Odometry

Yimin Lin\textsuperscript{1}, Zhaoxiang Liu\textsuperscript{1}, Jianfeng Huang\textsuperscript{1}, Chaopeng Wang\textsuperscript{1}, Guoguang Du\textsuperscript{1}, Jinqiang Bai\textsuperscript{2}, Shiguo Lian\textsuperscript{1}, Bill Huang\textsuperscript{1}

Abstract—For the autonomous navigation of mobile robots, robust and fast visual localization is a challenging task. Although some end-to-end deep neural networks for 6-DoF Visual Odometry (VO) have been reported with promising results, they are still unable to solve the drift problem in long-range navigation. In this paper, we propose the deep global-relative networks (DGRNets), which is a novel global and relative fusion framework based on Recurrent Convolutional Neural Networks (RCNNs). It is designed to jointly estimate global pose and relative localization from consecutive monocular images. DGRNets include feature extraction sub-networks for discriminative feature selection, RCNNs-type relative pose estimation sub-networks for smoothing the VO trajectory and RCNNs-type global pose regression sub-networks for avoiding the accumulation of pose errors. We also propose two loss functions: the first one consists of Cross Transformation Constraints (CTC) that utilize geometric consistency of the adjacent frames to train a more accurate relative sub-networks, and the second one is composed of CTC and Mean Square Error (MSE) between the predicted pose and ground truth used to train the end-to-end DGRNets. The competitive experiments on indoor Microsoft 7-Scenes and outdoor KITTI dataset show that our DGRNets outperform other learning-based monocular VO methods in terms of pose accuracy.

I. INTRODUCTION

The problem of visual localization has drawn significant attention from many researchers over the past few decades. Solutions for overcoming this problem come from computer vision and robotic communities by means of Structure from Motion (SfM) and visual Simultaneous Localization and Mapping (vSLAM) [1], [2]. Many variants of these solutions have started to make an impact in a wide range of applications, including autonomous navigation and augmented reality.

During the past few years, most of traditional visual localization techniques have been proposed and grounded on the estimate of the camera motion among a set of consecutive frames with geometric methods. For example, the feature-based method uses the projective geometry relations between 3D feature points of the scene and their projection on the image plane [3], [4], or the direct method minimizes the gradient of the pixel intensities across consecutive images [5], [6].

However, these techniques are critical to ideal and controlled environments, e.g., with a large amount of texture, unchanged illumination and without dynamic objects. Obviously, their performance drops quickly when facing those challenging and unpredictable scenarios.

Recently, a great breakthrough has been achieved in the Deep Learning (DL), through the application of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), e.g., for the object recognition and scene classification tasks. Therefore, learning-based visual odometry in the past few years has seen an increasing attention of the computer vision and robotic communities due to its potentials in learning capability and the robustness to camera parameters and challenging environments. However, so far they are still unable to outperform most state-of-the-art feature-based localization methods. The drift from the true trajectory due to accumulation of errors over time is inevitable in those learning based relative system. This is partly due to their inability to model all of the 3D structural constraints of the environment while learning from limited training datasets. Therefore, this is where global place recognition and camera relocalization plays a significant role in reducing global drifts. On the other hand, relative motion information from odometry can also be used to improve the global localization accuracy.

Guided by the previous considerations, in this work, we explore an efficient strategy for solving the drift problem by jointing visual odometry estimation and global pose regression. According to the high-level system overview in Fig. 1, our main contributions are as follows:
1) We demonstrate the DGRNets architecture consisting of the feature extraction sub-networks, the RCNNs-type relative pose estimation sub-networks, and finally fuse the global and relative 6-DoF pose by connecting to each other.

2) The training strategy: we firstly train the feature extraction and relative pose estimation sub-networks from a sequence of raw RGB images, and then the whole architecture is trained in an end-to-end manner to fill the rest of the global regression sub-networks according to different scenes.

3) We design two loss functions to improve the accuracy of DGRNets. For training the relative sub-networks, the CTC is proposed to enforce the geometric consistency between each other within a batch of frames. For training the whole networks, we minimize both CTC and the pose MSE.

4) We evaluate our DGRNets using 7-Scenes and KITTI datasets, and the results show it achieves state-of-the-art performance for learning-based monocular camera localization.

II. RELATED WORK

Over the past years, there are numerous approaches that have been proposed for visual localization. In this section, we discuss traditional geometry-based and recent learning-based localization approaches.

A. Geometry-based localization

Geometry-based localization estimates the camera motion among a set of consecutive frames with geometric methods. A variety of geometric methods can be classified into feature-based and direct methods.

Feature-based methods: most feature-based methods work by detecting feature points and matching them between consecutive frames. To improve pose accuracy, it minimizes the projective geometry errors between 3D feature points of the scene and their projection on the image plane, e.g., PTAM [4] is a classical vSLAM system. However, it may suffer from drift since it does not address the principle of loop closing. More recently, the ORB-SLAM algorithm by Mur-Artal et al. [3] is state-of-the-art vSLAM system designed for sparse feature tracking and reached impressive robustness and accuracy. In practice, it also suffers from a number of problems such as the inconsistency in initialization, and the drift caused by pure rotation.

Direct methods: in contrast, direct methods estimate the camera motion by minimizing the photometric error over all pixels across consecutive images. Engel at al. [5] developed LSD-SLAM, which is one of the most successful direct approaches. Direct methods do not provide better tolerance towards changing lighting conditions and often require more computational costs than feature-based methods since they work a global minimization using all the pixels in the image.

B. Learning-based localization

Even though Deep Neural Networks (DNNs) are not a novel concept, their popularity has grown in recent years due to a great breakthrough has been achieved in the computer vision community. Inspired by these achievements, lots of learning-based visual relocalization and odometry systems have been widely proposed to improve the 6-DoF pose estimation.

Visual relocalization: Learning-based relocalization systems are designed to learn from recognition to relocalization with very large scale classification datasets. For example, Kendall et al. proposed PoseNet [7], which was the first successful end-to-end pre-trained deep CNNs approach for 6-DoF pose regression. In addition, Clark et al. [8] introduced deep CNNs with Long-Short Term Memory (LSTM) units to avoid overfitting to training data while PoseNet needs to deal with this problem with careful dropout strategies.

Visual odometry: learning-based visual odometry systems are employed to learn the incremental change in position from images. LS-VO [9] is a CNNs architecture proposed to learn the latent space representation of the input Optical Flow field with the motion estimate task. SfM-Net [10] is a self-supervised geometry-aware CNNs for motion estimation in videos that decomposes frame-to-frame pixel motion in terms of scene and object depth, camera motion and 3D object rotations and translations. Recently, most state-of-the-art deep approaches to visual odometry employ not only CNNs, but also sequence-models, such as long-short term memory (LSTM) units [11], to capture long term dependencies in camera motion.

More recently, learning-based global and relative networks are designed for 6-DoF global pose regression and odometry estimation from consecutive monocular images. VLoc-Net [12] was a fusion architecture incorporates a global pose regression sub-networks and a Siamese-type relative pose estimation sub-networks. It takes two consecutive monocular images as input and jointly regress the 6-DoF global pose as well as the 6-DoF relative pose between the images. Brahmhhatt et al. [13] proposed a MapNet that enforces geometric constraints between relative poses and absolute poses in network training. Clark et al. [14] have presented a CNNs+Bi-LSTMs approach for 6-DoF video-clip relocalization that exploits the temporal smoothness of the video stream to improve the localization accuracy of the global pose estimation. According to those recent studies, in this paper we consider ways in which we can leverage the camera re-localization to improve the accuracy of 6-DoF image-sequences.

III. PROPOSED MODEL

In this section, we detail our learning-based global and relative fusion framework for jointly estimating global pose and odometry from consecutive monocular images. The proposed DGRNets are shown in Fig. 2. As we can see that it mainly consists of CNN-based feature extraction networks, RCNNs-type relative pose estimation sub-networks, and RCNNs-type
global pose regression sub-networks as well. More details are given in the next sections.

A. Network Architecture

1) **CNN-based feature extraction networks**: In order to learn effective features that are suitable for the global and relative pose estimation problem automatically, CNN-based feature extraction networks are developed to perform feature extraction on the monocular RGB image. We build upon this networks using the first three residual blocks of the ResNet-50 [15], whose structure is similar to VLocNET [12]. Each residual unit has a bottleneck architecture consisting of $1 \times 1$ convolution, $3 \times 3$ convolution, $1 \times 1$ convolution layers. Each of the convolutions is followed by batch normalization, scale and Exponential Linear Units (ELUs) [16].

2) **RCNNs-type relative pose estimation sub-networks**: Following the feature extraction networks, the deep RCNNs are designed to model dynamics and relations among a sequence of CNNs features. It takes a consecutive monocular RGB images as input, and uses the last two residual blocks of the ResNet-50 (Res 4 and Res 5) to concatenate features from the two individual CNN-based feature extraction streams. Note that the output dimension of this layer is $W \times H \times 1024$. As described in DeepVO [17], two Long Short-Term Memory (LSTMs) [18] are employed as RNNs to find and exploit correlations among images taken in long trajectories and each of the LSTM layers has 1000 hidden states. The RCNNs output pose estimation at each time step with a fully-connected layer $f_{c1}$ whose dimension is 1024. It corresponds in shape to the output of the relative RCNNs unit before the fusion stage. Note that the cell of LSTM stores the past few global poses and therefore it is able to improve the predicted pose accuracy of current image.

Specifically, the drawback of the most global pose regression [7] is that a pose can only be determined in a known training environment. So it is time consuming to retrain the whole networks according to different scenes. As shown in Fig. 3, in order to retrain our deep model faster, we design a scheme that the pose regression networks are divided into CNN1: feature extraction (as described in Section A-1) and RCNN2: RCNNs-type global pose regression sub-networks. Different scenes are fed into the common CNN1 to produce an effective feature in the monocular image, which is then passed through individual RCNN2 to learn for saving their landmark $S_i$. Thereby, we only need to retrain the RCNN2 for different scenes and each image still yields an accurate pose estimate at each $S_i$ through the networks.

Finally, the following fusion stage concatenates features from the two relative and global sub-networks, and reshapes its output to 1024, namely $f_{c3}$. We also add two inner-product layers for regressing the translation $T_i$ and quaternion $Q_i$, namely $f_{c4}$ and $f_{c5}$. Obviously, the dimensions of $f_{c4}$ and $f_{c5}$ layers are 3 and 4, respectively.

B. Geometric Consistency Loss

Here, we introduce CTC that are based on the fundamental concepts of composition of rigid-body transformations. Fig. 4 shows a sequential set of frames $F = \{I_0, I_1, I_2, I_3, I_4\}$. Note that $P_i = (Q_i, T_i)$ is a 6-DoF predicted pose, where $T_i$ and $Q_i$ denote the translation and quaternion of frame $i$. 
respectively. We train the networks to predict the transforms between each other: \([P_{01}, P_{12}, P_{23}, P_{34}, P_{02}, P_{24}, P_{04}]\). As an example, the predicted transform \(P_{01}\) from \(I_0\) to \(I_1\) should be equal to the product of the two \(\hat{P}_0\) and \(P_1\) transforms, where \(\hat{P}_i\) indicates the ground truth of frame \(i\), thus:

\[
P_{01} = \hat{P}_1 P_0^{-1} = \hat{P}_{01}
\]

Note that using (1) in practice, there exist errors in the predicted and ground truth, so we have CTC functions:

\[
\begin{align*}
L_0 &= \|P_{01} - \hat{P}_{01}\|^2, \\
L_1 &= \|P_{12} - \hat{P}_{12}\|^2, \\
L_2 &= \|P_{23} - \hat{P}_{23}\|^2, \\
L_3 &= \|P_{34} - \hat{P}_{34}\|^2, \\
L_4 &= \|P_{02} - \hat{P}_{02}\|^2, \\
L_5 &= \|P_{24} - \hat{P}_{24}\|^2, \\
L_6 &= \|P_{04} - \hat{P}_{04}\|^2
\end{align*}
\]

where \(\|\cdot\|^2\) is MSE. So the relative loss function which consists of (2) are shown as:

\[
\theta = \arg \min_\theta \frac{1}{N} \sum_{i=1}^{N} \sum_{k=0}^{6} (L_k)
\]

where \(\theta\) is the relative RCNNs parameters and \(N\) is the number of samples. We use this optimization (3) to train our relative RCNNs sub-networks. Note that, these constrains can be equal to a Local Bundle Adjustment in traditional vSLAM system [3], also known as windowed optimization. It is an efficient way to maintain a good quality pose over a local number of frames. So the CTC here are better strategies to learn about spatial relations of the environment. To train our 6-DoF end-to-end pose regression system, we can jointly use the global and relative loss function as follows:

\[
w = \arg \min_w \frac{1}{N} \sum_{i=1}^{N} \left\{ \sum_{k=0}^{6} (L_k^i) + \sum_{j=0}^{4} \|P_j^i - \hat{P}_j^i\|^2 \right\}
\]

where \(w\) is the global RCNNs parameters which cover the fusion module. It is obvious that (4) tries to minimize the Euclidean distance between the ground truth pose and estimated one while enforcing the geometric consistency between each other within a batch of frames.

IV. EXPERIMENTAL EVALUATION

In this section, we evaluate our proposed DGRNets architecture in comparison to the state-of-the-art on both indoor and outdoor datasets, followed by detailed analysis on the architectural decisions and finally, we demonstrate the efficacy of learning visual localization models.

A. Evaluation Datasets

We evaluate DGRNets on two well-known datasets: Microsoft 7-Scenes [19] and KITTI Visual Odometry benchmark [20]. We follow the original train and test splits provided by other literatures to facilitate comparison and benchmarking.

1) Microsoft 7-Scenes: it is a dataset that collect RGB-D images from seven different scenes in an indoor office environment. All scenes were recorded from a handheld Kinect RGB-D camera at 640×480 resolution. The dataset provides the ground truth poses extracted using KinectFusion. Each sequence was recorded with motion blur, perceptual aliasing and textureless features in the room, thereby making it a challenging dataset for relocalization and tracking.

2) KITTI Visual Odometry benchmark: it consists of 22 stereo sequences and they provide 11 sequences (00-10) with ground truth trajectories for training and 11 sequences (11-21) without ground truth for evaluation. This high-quality dataset was recorded with long sequences of varying speed, including a set of 41000 frames captured at 10 fps and a total driving distance of 39.2 km with frequent loop closures which are of interest in SLAM. So it is very popular for the monocular Visual Odometry algorithms.

B. Network Training

The network models were implemented with the TensorFlow framework and trained with NVIDIA GTX 1080.
GPUs and Intel Core i7 2.7GHz CPU. Adam optimizer was employed to train the networks for up to 2000 epochs with parameter $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The learning rate started from 0.001 and decreased by half for every 1/5 of total iterations. The sequence length of images fed to the relative and global pose estimator was 5. The size of image used by the networks was 224×224 pixel.

C. Microsoft 7-Scenes Datasets

In this experiment, we compare the performance of DGRNets with other state-of-the-art deep learning-based relocalization and tracking methods, namely PoseNet [7], DeepVO [21] and VLocNet [12]. In order to implement fair qualitative and quantitative comparison, we use the same 7-Scenes datasets for training and testing as described in [12], [14]. For each scene, we show their average translational and rotational errors in Table I.

It shows that DGRNets outperform previous CNN-based PoseNet by 95.9% in positional error and 74.8% in orientation error. Taking Pumpkin as an example, we achieve a positional error reduction from 0.47m for PoseNet to 0.022m for our method. The reason is that PoseNet always results in noisy predictions on single image. In contrast, the RCNNs in DGRNets constrict the motion space while using sequential images to improve global relocalization accuracy. Therefore, this experiment results validate that DGRNets have the effectiveness of using geometric constraints from consecutive images for improving relocalization accuracy. Furthermore, it can be seen that the proposed DGRNets significantly outperform the DeepVO approach in all of the test scenes, resulting in a 90% and 43% increase in position and orientation accuracy, respectively. The DeepVO network tries to regress the VO but probably suffers from high drifts. The reason is that the orientation changes in the training data are usually small and orientation is more prone to overfitting. However, our system reduces the drift over time due to the global pose regression strategy as done in the traditional visual SLAM system. In addition, DGRNets also perform better than VLocNet, and the orientation and positional error reduced by more than 31% and 62%, respectively. The main reason we found from VLocNet is that their global pose regression and visual odometry networks are predicted independently. But in our framework, we do fuse the results from global regression and relative pose estimation. In summary, these experimental results validate that our strategy is able to filter out the noise by fusing a series of measurements observed from global and relative networks over time.

D. KITTI Datasets

Next, we additionally deploy experiments in an outdoor environment for analyzing the large-scale VO performance. KITTI is much larger than typical indoor datasets like 7-Scenes, where sequence 00, 02, 08 and 09 are used for training the RCNNs-type relative sub-networks. As described in [17], the trajectories are segmented to different lengths to generate almost 7410 samples in total for training. The trained models are tested on the sequence 03, 04, 05, 06, 07 and 10. Note that, we extract 1/2 images from each sequence to compose datasets for training the global pose regression sub-networks, and the rest of them are used for accuracy evaluation. The performance of the DGRNets models is analyzed according to the KITTI VO/SLAM evaluation metrics, which are averaged Root Mean Square Errors (RMSEs) of the translational and rotational errors for all subsequences of lengths ranging from 100 to 800 meters and different speeds.

The KITTI testing results of our experiments are shown in Table II. We show quantitative comparison against two state-of-the-art VO approaches including traditional stereo odometry (VISO2) [22] and RCNN-type DeepVO [17]. The proposed method significantly outperforms the DeepVO approach in all of the test sequences, resulting in a 70% and 67% increase in translation and rotation accuracy, respectively. As shown in Fig. 5, DeepVO suffers from high drifts as the length of the trajectory increases and the errors of the rotation significantly increase because of significant changes on rotation during car driving. Unlike that, our DGRNets produce relatively accurate and consistent trajectories against to the ground truth. These owe to the global and relative architecture with the proposed CTC loss. In addition, it is able to match the performance of state-of-the-art local feature-based approaches VISO2. Although our rotational errors slightly worse than that of the VISO2, this may be due to the fact that our model are trained without enough data to cover the velocity and orientation variation. Finally, we can see that the absolute scale to each sequence is completely maintained during the end-to-end training.

### Table I

**RESULTS ON MICROSOFT 7-SCENES**

| Scene  | PoseNet | DeepVO | VLocNet | DGRNets |
|--------|---------|--------|---------|---------|
| Chess  | 0.32    | 8.12   | 0.06    | 2.61    |
| Fire   | 0.47    | 14.4   | 0.10    | 4.33    |
| Heads  | 0.29    | 12.0   | 0.35    | 7.11    |
| Office | 0.48    | 7.68   | 0.10    | 3.11    |
| Pumpkin| 0.47    | 8.42   | 0.11    | 3.30    |
| RedKitchen | 0.59 | 8.64   | 0.10    | 2.58    |
| Stairs | 0.47    | 13.8   | 0.45    | 9.18    |

Average $0.44$ 10.4 0.18 4.60 $0.048$ 3.80 0.018 2.62

$^1 t_{rel}$:average translational drift (m) on length, $r_{rel}$:average rotational drift (‘) on length.

### Table II

**RESULTS ON KITTI SEQUENCES**

| Seq. | DeepVO | VISO2 | DGRNets |
|------|--------|-------|---------|
| 03   | 8.49   | 6.89  | 3.21    |
| 04   | 7.19   | 6.97  | 2.12    |
| 05   | 2.62   | 3.61  | 1.53    |
| 06   | 5.42   | 5.82  | 1.58    |
| 07   | 3.91   | 4.60  | 1.85    |
| 10   | 8.11   | 8.83  | 1.17    |

Average $5.96$ 6.12 1.89 1.96 1.75 2.00

$^1 t_{rel}$:average translational RMSE drift (%) on length of 100m-800m, $r_{rel}$:average rotational RMSE drift (‘/100m) on length of 100m-800m.
In the next step, we plan to extend the ability of global networks to work at any unknown environment and increase the robustness of place recognition in cases where great illumination and appearance changes.

V. CONCLUSIONS

In this paper, we addressed the challenge of learning-based visual localization of a camera or an autonomous system with the novel DGRNets. It mainly consists of feature extraction sub-networks that determine the most discriminative feature as an input for the next two RCNNs, RCNNs-type relative sub-networks that estimate the egomotion of the camera and constrict the motion space while regressing the global localization, and RCNNs-type global sub-networks that are competent to model the 3D structural constraints of the environment while learning from the first two assistant networks. Furthermore, we presented the CTC loss function for training the relative sub-networks and loss function including CTC and global pose MSE for learning the global implicitly RCNNs from input image to camera pose. The indoor and outdoor experimental evaluations indicate that our global and relative model works well and outperforms existing state-of-the-art learning-based methods. In summary, it has been verified that DGRNets can produce accurate localization and be adopted to maintain a large feature map for drift correction under long range pose estimation.

The next step, we plan to extend the ability of global networks to work at any unknown environment and increase the robustness of place recognition in cases where great illumination and appearance changes.

REFERENCES

[1] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, “Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age,” IEEE Transactions on Robotics, vol. 32, no. 6, pp. 1309–1332, 2016.
[2] O. Özyesil, V. Voroninski, R. Basri, and A. Singer, “A survey of structure from motion,” Acta Numerica, vol. 26, pp. 305–364, 2017.
[3] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, “Orb-slam: a versatile and accurate monocular slam system,” IEEE Transactions on Robotics, vol. 31, no. 5, pp. 1147–1163, 2015.
[4] G. Klein and D. Murray, “Parallel tracking and mapping for small ar workspaces,” in 6th IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR), 2007, pp. 225–234.
[5] J. Engel, T. Schöps, and D. Cremers, “Lsd-slam: Large-scale direct monocular slam,” in European Conference on Computer Vision (ECCV), 2014, pp. 834–849.
[6] J. Engel, V. Koltun, and D. Cremers, “Direct sparse odometry,” IEEE transactions on pattern analysis and machine intelligence (TPAMI), vol. 40, no. 3, pp. 611–625, 2018.
[7] A. Kendall, M. Grimes, and R. Cipolla, “Posenet: A convolutional network for real-time 6-dof camera relocalization,” in Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2015, pp. 2938–2946.
[8] F. Walch, C. Hazirbas, L. Leal-Taixe, T. Sattler, S. Hilsenbeck, and D. Cremers, “Image-based localization using lstms for structured feature correlation,” in Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, pp. 627–637.
[9] G. Costante and T. A. Ciaruffiglia, “Ls-vo: Learning dense optical subspace for robust visual odometry estimation,” IEEE Robotics and Automation Letters, vol. 3, no. 3, pp. 1735–1742, 2018.
[10] S. Vijayanarasimhan, S. Rico, C. Schmid, R. Sukthankar, and K. Fragkiadaki, “Sfm-net: Learning of structure and motion from video,” arXiv preprint arXiv:1704.07804, 2017.
[11] G. Iyer, J. K. Murthy, K. Gunshi Gupta, and L. Puill, “Geometric consistency for self-supervised end-to-end visual odometry,” arXiv preprint arXiv:1804.03789, 2018.
[12] A. Valada, N. Radwan, and W. Burgard, “Deep auxiliary learning for visual localization and odometry,” arXiv preprint arXiv:1803.03642, 2018.
[13] S. Brahmbhatt, J. Gu, K. Kim, J. Hays, and J. Kautz, “Geometry-aware learning of maps for camera localization,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 2616–2625.
[14] R. Clark, S. Wang, A. Markham, N. Trigoni, and H. Wen, “Vidloc: A deep spatio-temporal model for 6-dof video-clip relocalization,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 3, 2017, pp. 2652–2660.
[15] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), 2016, pp. 770–778.
[16] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, “Fast and accurate deep network learning by exponential linear units (elus),” arXiv preprint arXiv:1511.07289, 2015.
[17] S. Wang, R. Clark, H. Wen, and N. Trigoni, “Deeppvo: Towards end-to-end visual odometry with deep recurrent convolutional neural networks,” in IEEE International Conference on Robotics and Automation (ICRA)., 2017, pp. 2043–2050.
[18] W. Zaremba and I. Sutskever, “Learning to execute,” arXiv preprint arXiv:1410.4615, 2014.
[19] J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. Fitzgibbon, “Scene coordinate regression forests for camera relocalization in rgb-d images,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 2930–2937.
[20] A. Geiger, P. Lenz, and R. Urtasun, “Are we ready for autonomous driving? the kitti vision benchmark suite,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012, pp. 3354–3361.
[21] V. Mohanty, S. Agrawal, S. Datta, A. Ghosh, V. D. Sharma, and D. Chakravarty, “Deeppvo: A deep learning approach for monocular visual odometry,” arXiv preprint arXiv:1611.06089, 2016.
[22] A. Geiger, J. Ziegler, and C. Stiller, “Sterescan: Dense 3d reconstruction in real-time,” in IEEE Intelligent Vehicles Symposium (IV), 2011, pp. 963–968.