2020 IEEE International Conference on Robotics and Automation (ICRA)
31 May - 31 August, 2020. Paris, France

DeepMEL: Compiling Visual Multi-Experience Localization into a Deep Neural Network

Mona Gridseth and Timothy D. Barfoot

Abstract— **Vision-based path following allows robots to autonomously repeat manually taught paths. Stereo Visual Teach and Repeat (VT&R) [1] accomplishes accurate and robust long-range path following in unstructured outdoor environments across changing lighting, weather, and seasons by relying on colour-constant imaging [2] and multi-experience localization [3]. We leverage multi-experience VT&R together with two datasets of outdoor driving on two separate paths spanning different times of day, weather, and seasons to teach a deep neural network to predict relative pose for visual odometry (VO) and for localization with respect to a path. In this paper we run experiments exclusively on datasets to study how the network generalizes across environmental conditions. Based on the results we believe that our system achieves relative pose estimates sufficiently accurate for in-the-loop path following and that it is able to localize radically different conditions against each other directly (i.e. winter to spring and day to night), a capability that our hand-engineered system does not have.**

I. INTRODUCTION

Vision-based path following algorithms have enabled robots to repeat paths autonomously in unstructured and GPS-denied environments. Furgale et al. [1] perform accurate metric and long-range path following with their VT&R system, which relies on a local relative pose map removing the need for global localization. The authors use sparse SURF features [4] to match images when performing VO and localization. Paton et al. extend VT&R to autonomous operation across lighting, weather, and seasonal change by adding colour-constant images [2] and multi-experience localization [3]. Multi-experience localization collects data every time the robot repeats a path and the most relevant experiences are chosen for feature matching.

Developing a robust and accurate VT&R system has taken a large research and engineering effort. As a result we can use outdoor datasets collected with VT&R across lighting and seasonal change to compile multi-experience localization into a deep neural network (DNN) for relative pose estimation. VT&R, which is shown to achieve high-accuracy path following [5], stores data in a spatio-temporal pose graph (see Figure 2). The pose graph contains the relative pose between temporally adjacent keyframes derived from VO and the relative pose of a keyframe with respect to the mapped path. Each traversal of the path is stored as a new experience. We sample relative poses between keyframes that are localized across different experiences and use them as labels for our training data. We design the DNN based on previous work by Melekhov et al. [6]. In particular, our DNN takes two pairs of stereo images and regresses the relative robot pose. In multi-experience localization VT&R relies on gradually adding new experiences over time to be able to localize when the environment changes. We aim to localize radically different path traversals against each other without the use of such intermediate bridging experiences.

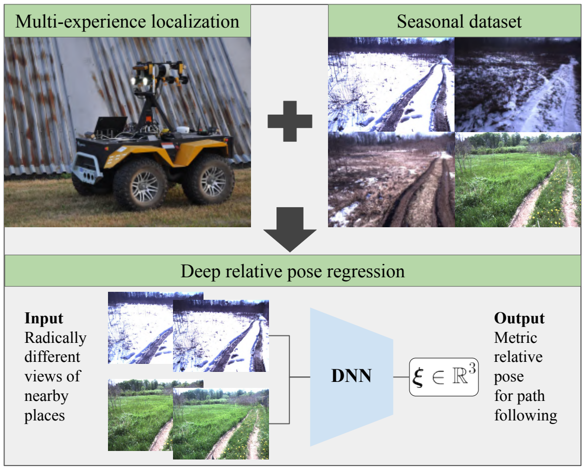

Multi-experience localization
Seasonal dataset
Deep relative pose regression
Input
Radically different views of nearby places
DNN
$\xi \in \mathbb{R}^3$
Output
Metric relative pose for path following

Fig. 1: We compile multi-experience localization for path following into a DNN. We use datasets collected with VT&R across different lighting and seasons to train the DNN to perform 3 degrees of freedom (DOF) relative pose estimation under changing environmental conditions.

We conduct experiments to test the ability of our regressor to generalize across large appearance change. In VT&R VO is used to propagate the current pose forward, while localization provides a pose correction by estimating the relative pose of the live frame with respect to the map. Since both VO and localization compute relative poses, we test our network's performance on both of these tasks. Using the exact same network architecture, we train one network with temporally adjacent keyframes for VO and one network with keyframes localized across different experiences.

The remainder of this paper is outlined as follows: Section II discusses related work, Section III gives the details of the network architecture and loss function, Section IV explains our training procedure and lays out the experiments, while Section V provides the results.

II. Related Work

VT&R [1] performs accurate [5] and robust autonomous path following. Moreover, the addition of colour-constant imagery [2] and multi-experience localization [3] enables the system to handle lighting, weather, and seasonal change.

Convolutional Neural Networks (CNN) have been included in different parts of the visual pose estimation pipeline

All authors are with the University of Toronto Institute for Aerospace Studies (UTIAS), 4925 Dufferin St, Ontario, Canada. mona.gridseth@robotics.utias.utoronto.ca, tim.barfoot@utoronto.ca

978-1-7281-7395-5/20/$31.00 ©2020 IEEE

to tackle appearance change. Several such approaches are tested against the Long-Term Visual Localization Benchmark [7]. Examples include learning robust descriptors [8]–[14], semantic information [15], [16], and place recognition [17], and transforming whole images to different conditions [18].

Others have in turn focused on replacing the whole pose-estimation pipeline with neural networks by regressing pose directly from images in an end-to-end fashion, several examples of which are presented in a survey on deep learning and visual simultaneous localization and mapping (SLAM) [19]. Early work on absolute pose regression came from the development of PoseNet [20]. The system is based on a pre-trained GoogleNet architecture and regresses 6-DOF pose for metric relocalization of a monocular camera. Kendall et al. extended the work to use a Bayesian neural network providing relocalization uncertainty [21] and an improved loss function [22]. Naseer et al. [23] improve on PoseNet by generating additional augmented data leading to improved accuracy, while Walch et al. [24] perform structured dimensionality reduction on the CNN output with the help of long short-term memory (LSTM) units. In [25] and [26] the authors were able to reduce localization error by passing sequential image data to recurrent models with LSTM units.

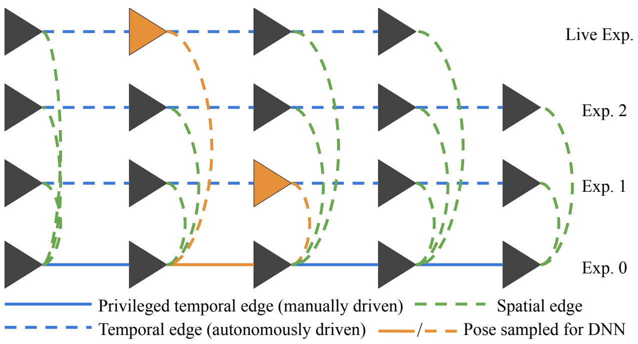
Melekhov et al. [6] use a Siamese CNN architecture based on AlexNet [27] to compute relative camera pose from a pair of images. Similarly Bateux et al. [28] regress relative pose for use in visual servoing. VO is a special case of relative pose estimation, which has been explored extensively in the context of deep learning [29]–[33]. In several examples authors combine CNNs with LSTM units to incorporate a sequence of data [30], [32]. Iyer et al. [32] use geometric consistency constraints to train their network in a self-supervised manner. In a different approach, Peretroukhin et. al. have combined deep learning with traditional pose estimation by learning pose corrections [34] and rotation [35], which they fuse with relative pose estimates.

Relative pose estimation has also been used as a tool to regress absolute pose. In particular, Laskar et al. [36] combine relative pose regression with image retrieval from a database. Balntas et al. [37] retrieve nearest neighbours based on learned image features before regressing relative pose to refine the absolute pose. Saha et al. [38] classify anchor points to which they regress relative pose. Oliveira et al. [39] combine the outputs of two DNNs for visual odometry and absolute pose estimation, respectively, to accomplish topometric localization. The work was further extended by using multi-task learning for localization [40]. In our work we focus on robustness to large appearance change for relative pose regression and show experimental results on two challenging outdoor paths.

III. Methodology

A. System Overview

VT&R stores image keyframes as vertices and relative poses between them as edges in a spatio-temporal pose graph. Figure 2 illustrates a pose graph with temporal edges derived from VO and spatial edges that connect keyframes from autonomous repeats to keyframes on the manually driven teach pass. Our system estimates both relative pose for VO as well as metric localization with respect to the path. We use the same neural network architecture and train two networks separately on data for VO and localization. Since VT&R provides highly accurate path following [5], we sample relative pose labels from the VT&R pose graph. The DNN takes as input RGB stereo images from a pair of keyframes and regresses a 3-DOF relative pose given in the robot frame. For path following the offset from the path and heading are the most important DOF and so we opt to estimate $\boldsymbol{\xi} = \begin{bmatrix} x & y & \theta \end{bmatrix}^T \in \mathbb{R}^3$.



Live Exp.
Exp. 2
Exp. 1
Exp. 0
Privileged temporal edge (manually driven)
Temporal edge (autonomously driven)
Spatial edge
Pose sampled for DNN

Fig. 2: Image keyframes and the relative poses between them are stored in a spatio-temporal pose graph. Temporal edges represent relative poses from VO and spatial edges give the relative pose between a keyframe on a live experience and a keyframe on the privileged teach path. Several live keyframes may localize to the same privileged keyframe. The vertices and edges in orange show how we can sample relative poses in time and space, by compounding spatial and temporal transforms, to use as labels for training the DNN.

B. Network Architecture

Our DNN architecture is inspired by the one presented in [6]. As in [6], the convolutional part of the DNN is taken from the AlexNet architecture [27]. We opt to input a stack of all four RGB images to the network as in [30], resulting in 12 input channels. Experimenting with a Siamese architecture did not cause improvements in our case. Our images are different in size (512 × 384) from the standard input to AlexNet (224 × 224) and hence we make use of Spatial Pyramid Pooling (SPP) [43] as in [6] to reduce the size of our feature map before the fully connected layers. SPP lets us create a fixed-sized output while maintaining spatial information by pooling the responses of each feature in spatial bins (we use max pooling). The size of the output is the number of bins times the number of features. We use four levels of pyramid pooling with the following bins: $5 \times 5, 3 \times 3, 2 \times 2$, and 1×1. Finally, we keep the same fully connected layers as in AlexNet, but add one more fully connected layer with 3 connections to regress the 3-DOF pose. An overview of the network can be seen in Figure 3.

C. Loss Function

We use a simple quadratic loss function that takes the difference in target and predicted coordinates. Translation and rotation are manually weighted using a diagonal matrix $\mathbf{W}$ with 1.0 on the diagonal for x and y and 10.0 for θ. As pointed out in [22], angles may wrap around 2π, but this

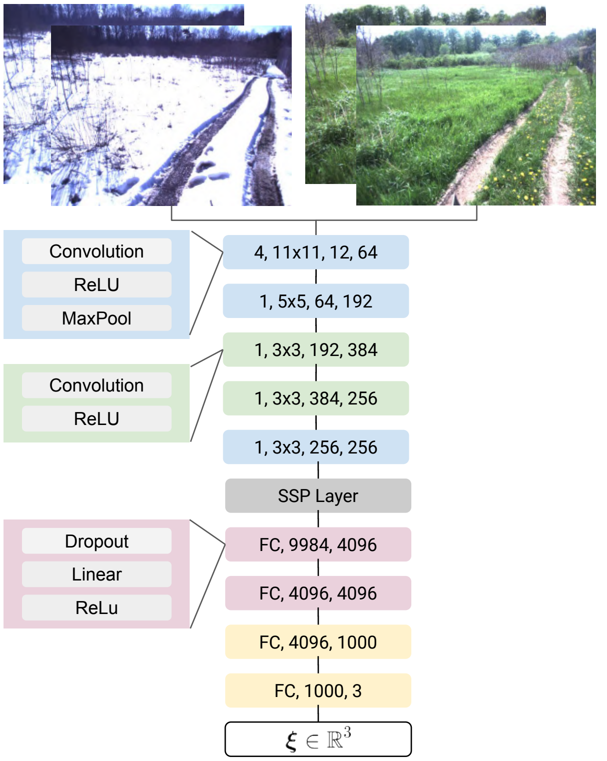


Fig. 3: The neural network takes two sets of RGB stereo images and produces a 3-DOF relative pose. The architecture contains convolution layers, spatial pyramid pooling, and fully connected layers. We list the stride, kernel size, and number of input and output channels for the convolutional layers as well as input and output sizes for the fully connected layers.

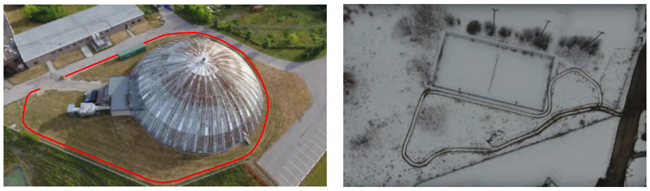


Fig. 4: Aerial view of the paths for the UTIAS In The Dark and UTIAS Multi Season datasets.

is not a problem we would encounter as we are estimating small relative poses. The loss is

$$\mathcal{L} = \frac{1}{2}\left(\boldsymbol{\xi} - \hat{\boldsymbol{\xi}}\right)^T \mathbf{W}\left(\boldsymbol{\xi} - \hat{\boldsymbol{\xi}}\right), \tag{1}$$

where $\boldsymbol{\xi}$ represents the target pose that we have sampled from the VT&R pose graph and $\hat{\boldsymbol{\xi}}$ is the estimated pose.

IV. Experiments

We conduct experiments to test relative pose estimation for VO and localization, where localization is performed between stereo camera frames taken during different times of day, weather, and seasons. The experiments make use of data collected with a Clearpath Grizzly RUV with a maximum speed of 1 m/s equipped with a factory-calibrated PointGrey Bumblebee XB3 stereo camera with 24 cm baseline, see Figure 1. We use VT&R with colour-constant images [2] and multi-experience localization [3] to label the data. Multi-experience localization stores each traversal of the path in the pose graph. During a repeat a set of the experiences most similar to the current conditions are chosen for feature matching when localizing with respect to the path.

A. Training and Testing

We train, validate, and test our system on two outdoor paths. The first dataset, called UTIAS In The Dark, covers a 250 m path following a paved road and grass in an area with buildings. The path is repeated once per hour for over 24 hours covering significant lighting change. The robot has headlights for driving during the night. The path has 45 repeats from which we choose 5 for testing and use the remaining for training and validation. We only train and test our network once for each path and do not re-train for each test condition. The path in the second dataset, called UTIAS Multi Season, is about 160 m. It covers an area with rugged terrain and vegetation. Data is collected from winter to spring and includes a total of 138 repeats, 8 of which are held out for testing. Figure 4 shows an aerial view of the paths.

Data collected during path traversals by the hand-engineered VT&R system is organized in a spatio-temporal pose graph illustrated in Figure 2. With the help of multi-experience localization, VT&R is able to localize back to the teach pass during long-term driving, providing us with training data across large environmental change. In order to generate pose labels for training and validation for VO, we sample the temporal edges between immediately adjacent keyframes. For localization we can sample randomly from the graph in both space and time, allowing us to generate large datasets connecting keyframes from both similar and radically different conditions. An example of such a sample is illustrated in orange in Figure 2. We randomly pick a vertex from an autonomous repeat and find to which privileged teach keyframe this vertex is localized. If we want to sample in space we can pick another teach keyframe in the same area. Finally, we randomly pick an autonomous repeat vertex localized to the chosen teach vertex. We compound the

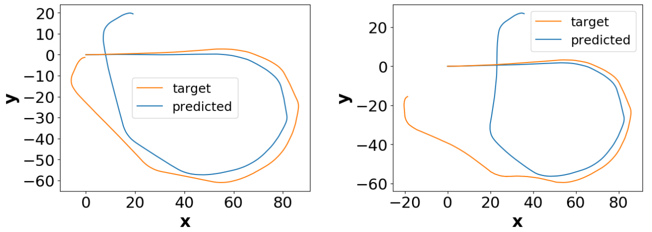


Fig. 5: Integrated VO for day and evening representing one of the most and least accurate results, respectively.

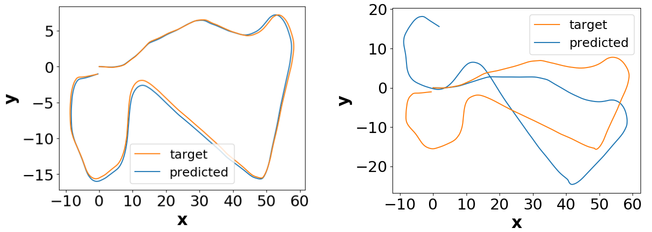


Fig. 6: Integrated VO for sunny weather with snow on the ground and overcast weather with no snow representing one of the most and least accurate results, respectively.

TABLE I: RMSE for each DOF for the UTIAS Multi Season dataset. The diagonal entries are VO results, while the off-diagonal entries are localization results. The rows are used as repeats and the columns as teach runs. The green and red cells are better and worse performing examples, respectively, picked for further qualitative analysis.

		Snow		Some snow		No snow		Green	
		Sunny	Overcast	Sunny	Overcast	Sunny	Overcast	Sunny	Overcast
Snow	Sun	$x : 0.015$ $y : 0.0039$ $\theta : 0.11$	$x : 0.073$ $y : 0.023$ $\theta : 0.54$	$x : 0.086$ $y : 0.028$ $\theta : 0.55$	$x : 0.086$ $y : 0.041$ $\theta : 0.85$	$x : 0.070$ $y : 0.017$ $\theta : 0.40$	$x : 0.075$ $y : 0.023$ $\theta : 0.49$	$x : 0.089$ $y : 0.028$ $\theta : 0.59$	$x : 0.082$ $y : 0.024$ $\theta : 0.49$
	Overcast	$x : 0.073$ $y : 0.031$ $\theta : 0.57$	$x : 0.032$ $y : 0.0032$ $\theta : 0.11$	$x : 0.074$ $y : 0.031$ $\theta : 0.60$	$x : 0.088$ $y : 0.046$ $\theta : 0.81$	$x : 0.074$ $y : 0.027$ $\theta : 0.55$	$x : 0.080$ $y : 0.037$ $\theta : 0.59$	$x : 0.099$ $y : 0.040$ $\theta : 0.79$	$x : 0.088$ $y : 0.035$ $\theta : 0.68$
Some snow	Sun	$x : 0.077$ $y : 0.029$ $\theta : 0.64$	$x : 0.075$ $y : 0.032$ $\theta : 0.65$	$x : 0.014$ $y : 0.0059$ $\theta : 0.13$	$x : 0.086$ $y : 0.037$ $\theta : 0.92$	$x : 0.070$ $y : 0.028$ $\theta : 0.61$	$x : 0.074$ $y : 0.029$ $\theta : 0.59$	$x : 0.100$ $y : 0.039$ $\theta : 0.81$	$x : 0.091$ $y : 0.035$ $\theta : 0.70$
	Overcast	$x : 0.13$ $y : 0.071$ $\theta : 2.1$	$x : 0.13$ $y : 0.069$ $\theta : 2.1$	$x : 0.12$ $y : 0.066$ $\theta : 2.4$	$x : 0.019$ $y : 0.0025$ $\theta : 0.13$	$x : 0.12$ $y : 0.065$ $\theta : 1.3$	$x : 0.13$ $y : 0.069$ $\theta : 2.1$	$x : 0.12$ $y : 0.071$ $\theta : 1.4$	$x : 0.13$ $y : 0.076$ $\theta : 1.6$
No snow	Sun	$x : 0.056$ $y : 0.017$ $\theta : 0.39$	$x : 0.061$ $y : 0.020$ $\theta : 0.41$	$x : 0.065$ $y : 0.023$ $\theta : 0.48$	$x : 0.079$ $y : 0.031$ $\theta : 0.63$	$x : 0.011$ $y : 0.0033$ $\theta : 0.10$	$x : 0.057$ $y : 0.021$ $\theta : 0.42$	$x : 0.076$ $y : 0.025$ $\theta : 0.52$	$x : 0.074$ $y : 0.019$ $\theta : 0.46$
	Overcast	$x : 0.071$ $y : 0.025$ $\theta : 0.49$	$x : 0.067$ $y : 0.034$ $\theta : 0.47$	$x : 0.070$ $y : 0.028$ $\theta : 0.57$	$x : 0.082$ $y : 0.033$ $\theta : 0.75$	$x : 0.057$ $y : 0.024$ $\theta : 0.47$	$x : 0.012$ $y : 0.0042$ $\theta : 0.012$	$x : 0.082$ $y : 0.031$ $\theta : 0.61$	$x : 0.074$ $y : 0.028$ $\theta : 0.54$
Green	Sun	$x : 0.097$ $y : 0.034$ $\theta : 0.63$	$x : 0.10$ $y : 0.039$ $\theta : 0.92$	$x : 0.10$ $y : 0.042$ $\theta : 0.81$	$x : 0.095$ $y : 0.045$ $\theta : 0.96$	$x : 0.088$ $y : 0.031$ $\theta : 0.66$	$x : 0.097$ $y : 0.036$ $\theta : 0.74$	$x : 0.019$ $y : 0.0033$ $\theta : 0.14$	$x : 0.070$ $y : 0.029$ $\theta : 0.50$
	Overcast	$x : 0.090$ $y : 0.029$ $\theta : 0.55$	$x : 0.099$ $y : 0.032$ $\theta : 0.69$	$x : 0.10$ $y : 0.033$ $\theta : 0.65$	$x : 0.10$ $y : 0.042$ $\theta : 0.94$	$x : 0.10$ $y : 0.026$ $\theta : 0.52$	$x : 0.097$ $y : 0.030$ $\theta : 0.61$	$x : 0.089$ $y : 0.030$ $\theta : 0.52$	$x : 0.013$ $y : 0.0035$ $\theta : 0.14$

TABLE II: RMSE for each DOF for the UTIAS In The Dark dataset. The diagonal entries are VO results, while the off-diagonal entries are localization results. The rows are used as repeats and the columns as teach runs. The green and red cells are better and worse performing examples, respectively, picked for further qualitative analysis.

	Morning	Sun Flare	Day	Evening	Night
Morning	$x : 0.0073$ $y : 0.0022$ $\theta : 0.080$	$x : 0.012$ $y : 0.0053$ $\theta : 0.17$	$x : 0.013$ $y : 0.0058$ $\theta : 0.15$	$x : 0.013$ $y : 0.0079$ $\theta : 0.15$	$x : 0.013$ $y : 0.0093$ $\theta : 0.21$
Sun Flare	$x : 0.010$ $y : 0.0051$ $\theta : 0.12$	$x : 0.0079$ $y : 0.0023$ $\theta : 0.084$	$x : 0.011$ $y : 0.0060$ $\theta : 0.14$	$x : 0.013$ $y : 0.0069$ $\theta : 0.15$	$x : 0.013$ $y : 0.0086$ $\theta : 0.20$
Day	$x : 0.012$ $y : 0.0058$ $\theta : 0.13$	$x : 0.011$ $y : 0.0058$ $\theta : 0.14$	$x : 0.0074$ $y : 0.0021$ $\theta : 0.079$	$x : 0.013$ $y : 0.0069$ $\theta : 0.15$	$x : 0.013$ $y : 0.0087$ $\theta : 0.20$
Evening	$x : 0.019$ $y : 0.011$ $\theta : 0.22$	$x : 0.019$ $y : 0.011$ $\theta : 0.22$	$x : 0.019$ $y : 0.011$ $\theta : 0.22$	$x : 0.0091$ $y : 0.0037$ $\theta : 0.092$	$x : 0.020$ $y : 0.012$ $\theta : 0.28$
Night	$x : 0.015$ $y : 0.013$ $\theta : 0.29$	$x : 0.015$ $y : 0.014$ $\theta : 0.31$	$x : 0.016$ $y : 0.013$ $\theta : 0.29$	$x : 0.016$ $y : 0.014$ $\theta : 0.31$	$x : 0.0047$ $y : 0.0041$ $\theta : 0.092$

transforms to get the relative pose associated with our pair of keyframes. For this paper we sample only in time and do not move along the graph in the spatial direction. For the UTIAS In The Dark dataset our training and validation sets have $360,000$ and $40,000$ samples for localization, respectively. For VO we get $64,530$ and 7170 samples. The UTIAS Multi Season dataset has $450,000$ and $50,000$ samples for localization training and validation, respectively. For VO we have $69,659$ training samples and 7739 validation samples.

When processing the test runs we keep the data sequential to assess performance in a realistic scenario. We test localization across environmental change by performing localization for every pair of runs in the test set (one run is used as the teach pass and the other as the autonomous repeat). Specifically, this lets us localize radically different traversals directly without the use of any intermediate bridging experiences. We test VO standalone for the same runs.

Path following is performed by alternating between using VO to propagate the pose forward and localization to provide a pose correction. We perform two experiments for localization. In order to test the localization network standalone we compute the relative pose between the live and teach keyframes that are localized to each other by VT&R and compare directly to the VT&R labels. Note that VT&R does not consider global pose estimates, only the relative pose of the robot with respect to the teach pass. We also include a qualitative path following experiment, where we test the VO and localization networks together. We start by computing the relative pose between the initial live and teach keyframe pair. Next we use VO, as computed by the network, to propagate this pose forward for a window of possible next teach keyframes and choose the teach keyframe that gives the smallest new relative pose. As the correction step we use the localization network to compute a relative pose between the live and teach keyframes. We combine the propagated pose with the pose from localization by computing a weighted average with weights 0.3 and 0.7, respectively.

We train our network on an NVIDIA GTX 1080 Ti GPU with a batch size of 64 and use early stopping based on the validation loss to determine the number of epochs. We use the Adam optimizer [44] with learning rate 0.0001 and other parameters set to their default values. Network inference runs at a minimum 50 fps on a Lenovo laptop with one GPU.

V. Results

We conduct standalone experiments for the localization and VO networks for data with large appearance variation in an outdoor environment. Tables I and II list the root mean squared error (RSME) for each run in the test sets. We compare performance with the VT&R system, which we know has centimeter-level error on kilometer-scale repeats [5]. The values on the diagonal are results for VO, while the rest are for localization. The rows represent repeat runs while the columns are used as teach runs. For the paths in our datasets the the relative pose values for x can typically fall between 0 to 30 cm. y normally varies between +/- 10 cm, but can reach almost 40 cm on sharp turns. Similarly θ mostly varies between +/- 5 degrees, but may reach almost 40 degrees on sharp turns. If the network had only learned

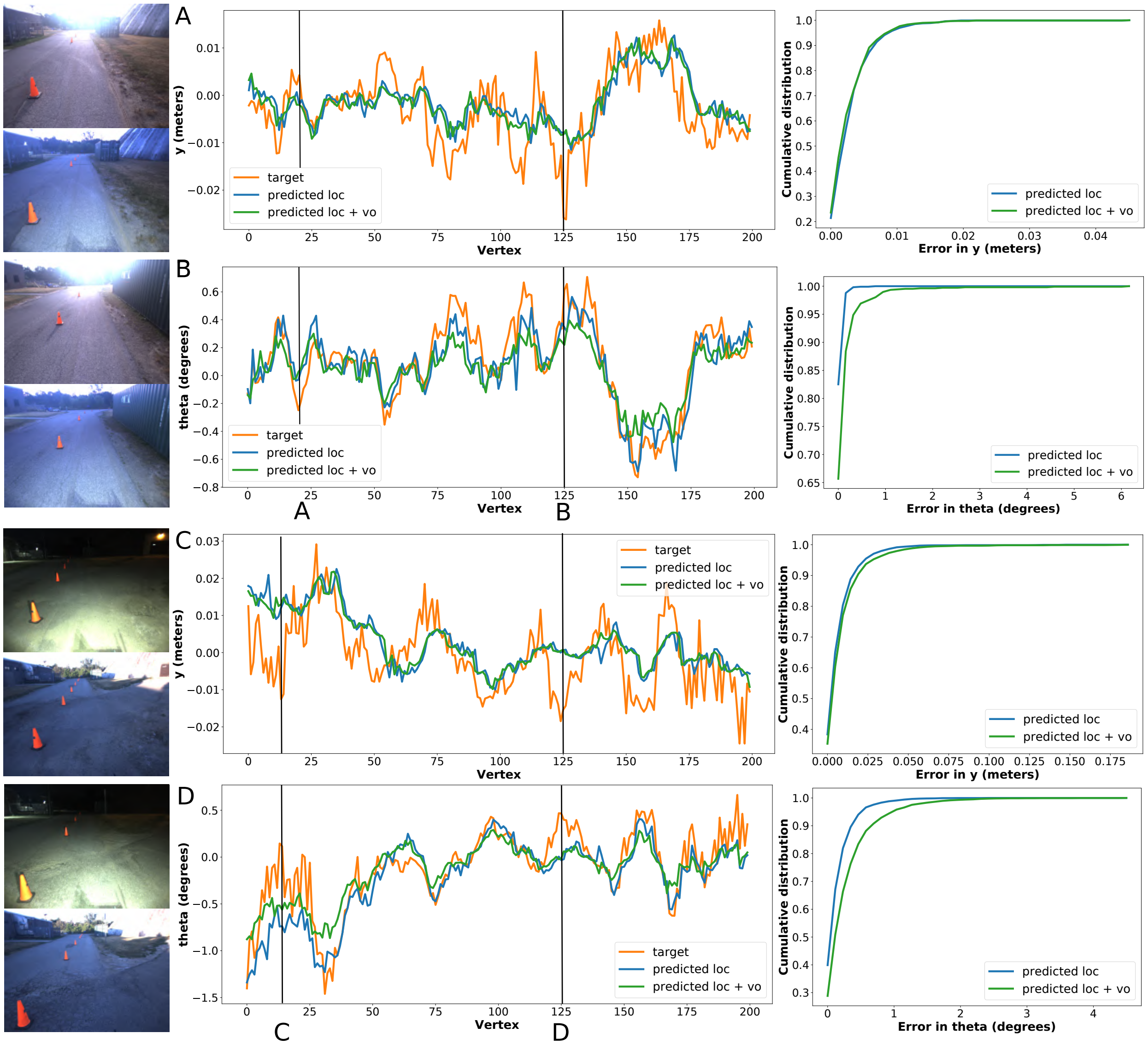


Fig. 7: The figure shows localization results for one of the better (top) test sequences from the UTIAS In The Dark dataset and one with larger errors (bottom). We plot the relative pose estimates for y and θ from the localization network (blue) as well as pose estimates from combining VO and localization (green) together with the target values for a segment of the full path. The plots on the right show the cumulative distribution of errors for the full path. The image pairs on the left provide anecdotal examples from the test sequence and are marked in the plot.

to randomly return small pose estimates, path following would not be possible due to the difference in relative pose size on straight road versus turns. Furthermore, repeat speed and the number of repeat keyframes can vary between runs and so simply replaying previous experiences would also fail quickly. With these approximate numbers in mind, we see that our system achieves low errors across a range of conditions. For tests across lighting change localizing evening and night repeats are the most challenging, but they do not perform much worse than the other combinations. For the seasonal tests we see that the network is able to localize runs as different as winter and spring. These examples show the system's potential to localize against large environmental changes directly without relying on intermediate experiences.

To supplement our quantitative findings we provide more detailed plots for two test cases from each dataset. We pick one of the best performing test cases (marked in green in the tables) and one of the cases with the largest errors (marked in red). We integrate the results from VO to show the full paths in Figures 5 and 6, while Figures 7 and 8 display localization results. For path following, the most important performance indicators are the lateral and heading errors with respect to the path. For a small segment of each path we plot the target y and θ values against those predicted standalone by the localization network as well as the path following method that combines VO and localization network outputs for prediction and correction. We think the latter method has a smoothing effect on the pose estimates. The fact that this method chooses different teach keyframes for localization than the original VT&R system may account for some of the discrepancy between the two solutions in Figure 8. Additionally, we plot the cumulative distribution of y and θ errors for the whole test sequence and include two example teach-and-repeat image pairs, illustrating the

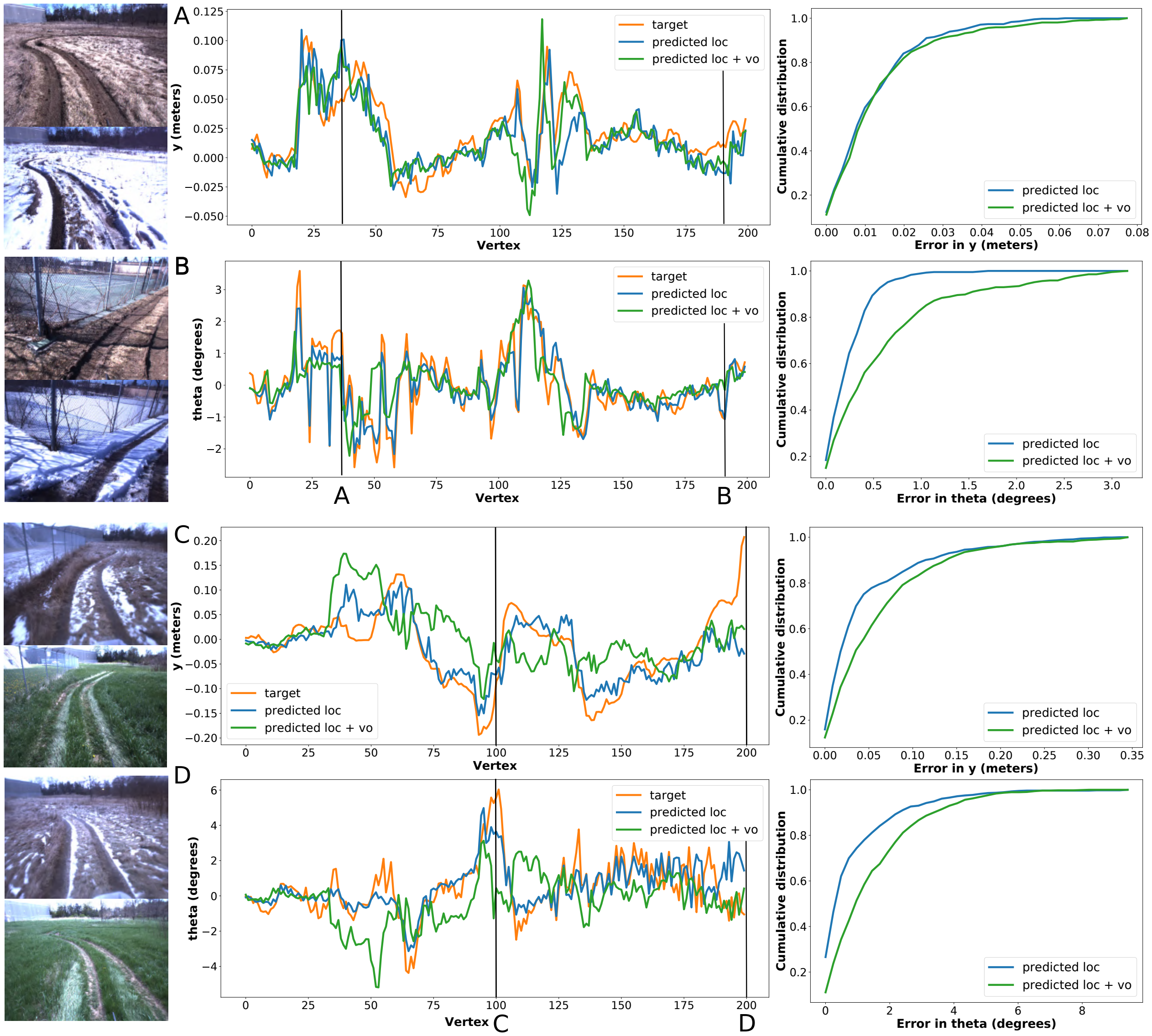


Fig. 8: The figure shows localization results for one of the better (top) test sequences from the UTIAS Multi Season dataset and one with larger errors (bottom). We plot the relative pose estimates for y and θ from the localization network (blue) as well as pose estimates from combining VO and localization (green) together with the target values for a segment of the full path. The plots on the right show the cumulative distribution of errors for the full path. The image pairs on the left provide anecdotal examples from the test sequence and are marked in the plot.

challenging environmental change. The path from the UTIAS In The Dark Dataset has less sharp turns and smaller lateral path offsets resulting in a smaller signal-to-noise ratio in the data making the value of y harder to predict, see Figure 7. Given the RMS errors as well as the plots from the example runs, we think that this localization system would be sufficiently accurate for path following in the loop.

VI. Conclusion and Future Work

In this work, we present a DNN that can perform relative pose regression for both VO and localization with respect to a path in an outdoor environment across illumination and seasonal change. We collect labels for training and testing from a spatio-temporal pose graph generated by VT&R. We conduct experiments across environmental change on two outdoor paths. The network carries out VO under different and challenging conditions, including night time driving. Furthermore, our network can perform localization for input image pairs from different times of day or seasons without the need of intermediate bridging experiences, which are necessary for long-term operation with the original VT&R system. From the performance we achieve on these datasets we believe that the localization system is sufficiently accurate for in-the-loop path following.

Tackling the localization problem across outdoor environmental change is a first step in applying deep learning more generally to path following. We want to improve the technique by enabling transfer to paths not seen during training. Ultimately we aim to close the loop in real time with the localization system we have developed in this paper.

Acknowledgment

We would like to thank Clearpath Robotics and the Natural Sciences and Engineering Research Council of Canada (NSERC) for supporting this work.

References

[1] P. Furgale and T. D. Barfoot, "Visual teach and repeat for long-range rover autonomy," *Journal of Field Robotics*, vol. 27, no. 5, pp. 534–560, 2010. [Online]. Available: http://dx.doi.org/10.1002/rob.20342

[2] M. Paton, K. MacTavish, C. J. Ostafew, and T. D. Barfoot, "It's not easy seeing green: Lighting-resistant stereo visual teach amp; repeat using color-constant images," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 1519–1526.

[3] M. Paton, K. MacTavish, M. Warren, and T. D. Barfoot, "Bridging the appearance gap: Multi-experience localization for long-term visual teach and repeat," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2016, pp. 1918–1925.

[4] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *European conference on computer vision*. Springer, 2006, pp. 404–417.

[5] L. Clement, J. Kelly, and T. D. Barfoot, "Monocular visual teach and repeat aided by local ground planarity," in *Field and Service Robotics*. Springer, 2016, pp. 547–561.

[6] I. Melekhov, J. Ylioinas, J. Kannala, and E. Rahtu, "Relative camera pose estimation using convolutional neural networks," in *International Conference on Advanced Concepts for Intelligent Vision Systems*. Springer, 2017, pp. 675–687.

[7] T. Sattler, W. Maddern, C. Toft, A. Torii, L. Hammarstrand, E. Stenborg, D. Safari, M. Okutomi, M. Pollefeys, J. Sivic *et al.*, "Benchmarking 6dof outdoor visual localization in changing conditions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8601–8610.

[8] M. Dymczyk, E. Stumm, J. Nieto, R. Siegwart, and I. Gilitschenski, "Will it last? learning stable features for long-term visual localization," in *2016 Fourth International Conference on 3D Vision (3DV)*, Oct 2016, pp. 572–581.

[9] N. Sünderhauf, S. Shirazi, A. Jacobson, F. Dayoub, E. Pepperell, B. Upcroft, and M. Milford, "Place recognition with convnet landmarks: Viewpoint-robust, condition-robust, training-free," in *Robotics: Science and Systems*, 2015.

[10] P.-E. Sarlin, C. Cadena, R. Siegwart, and M. Dymczyk, "From coarse to fine: Robust hierarchical localization at large scale," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 716–12 725.

[11] N. Piasco, D. Sidibé, V. Gouet-Brunet, and C. Demonceaux, "Learning scene geometry for visual localization in challenging conditions," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 9094–9100.

[12] M. Dusmanu, I. Rocco, T. Pajdla, M. Pollefeys, J. Sivic, A. Torii, and T. Sattler, "D2-net: A trainable cnn for joint description and detection of local features," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8092–8101.

[13] Z. Luo, T. Shen, L. Zhou, J. Zhang, Y. Yao, S. Li, T. Fang, and L. Quan, "Contextdesc: Local descriptor augmentation with cross-modality context," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2527–2536.

[14] J. Revaud, P. Weinzaepfel, C. De Souza, N. Pion, G. Csurka, Y. Cabon, and M. Humenberger, "R2d2: Repeatable and reliable detector and descriptor," *arXiv preprint arXiv:1906.06195*, 2019.

[15] J. L. Schönberger, M. Pollefeys, A. Geiger, and T. Sattler, "Semantic visual localization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6896–6906.

[16] M. Larsson, E. Stenborg, C. Toft, L. Hammarstrand, T. Sattler, and F. Kahl, "Fine-grained segmentation networks: Self-supervised segmentation for improved long-term visual localization," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 31–41.

[17] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 5297–5307.

[18] L. Clement and J. Kelly, "How to train a cat: Learning canonical appearance transformations for direct visual localization under illumination change," *IEEE Robotics and Automation Letters*, 2017.

[19] R. Li, S. Wang, and D. Gu, "Ongoing evolution of visual slam from geometry to deep learning: Challenges and opportunities," *Cognitive Computation*, vol. 10, no. 6, pp. 875–889, Dec 2018. [Online]. Available: https://doi.org/10.1007/s12559-018-9591-8

[20] 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*, 2015, pp. 2938–2946.

[21] A. Kendall and R. Cipolla, "Modelling uncertainty in deep learning for camera relocalization," in *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2016-June. Institute of Electrical and Electronics Engineers Inc., jun 2016, pp. 4762–4769.

[22] ——, "Geometric loss functions for camera pose regression with deep learning," in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-January. Institute of Electrical and Electronics Engineers Inc., nov 2017, pp. 6555–6564.

[23] T. Naseer and W. Burgard, "Deep regression for monocular camera-based 6-dof global localization in outdoor environments," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 1525–1530.

[24] 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*, 2017, pp. 627–637.

[25] 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*, 2017, pp. 6856–6864.

[26] M. Patel, B. Emery, and Y. Y. Chen, "ContextualNet: Exploiting contextual information using LSTMs to improve image-based localization," in *Proceedings - IEEE International Conference on Robotics and Automation*. Institute of Electrical and Electronics Engineers Inc., sep 2018, pp. 5890–5896.

[27] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[28] Q. Bateux, E. Marchand, J. Leitner, F. Chaumette, and P. Corke, "Training deep neural networks for visual servoing," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1–8.

[29] V. Mohanty, S. Agrawal, S. Datta, A. Ghosh, V. D. Sharma, and D. Chakravarty, "Deepvo: A deep learning approach for monocular visual odometry," *ArXiv*, vol. abs/1611.06069, 2016.

[30] S. Wang, R. Clark, H. Wen, and N. Trigoni, "End-to-end, sequence-to-sequence probabilistic visual odometry through deep neural networks," *The International Journal of Robotics Research*, vol. 37, no. 4-5, pp. 513–542, 2018.

[31] H. Zhan, R. Garg, C. S. Weerasekera, K. Li, H. Agarwal, and I. M. Reid, "Unsupervised Learning of Monocular Depth Estimation and Visual Odometry with Deep Feature Reconstruction," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, dec 2018, pp. 340–349.

[32] G. Iyer, J. Krishna Murthy, G. Gupta, K. Madhava Krishna, and L. Paull, "Geometric consistency for self-supervised end-to-end visual odometry," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, vol. 2018-June. IEEE Computer Society, dec 2018, pp. 380–388.

[33] G. Costante, M. Mancini, P. Valigi, and T. A. Ciarfuglia, "Exploring Representation Learning With CNNs for Frame-to-Frame Ego-Motion Estimation," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 18–25, dec 2015.

[34] V. Peretroukhin and J. Kelly, "Dpc-net: Deep pose correction for visual localization," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2424–2431, July 2018.

[35] V. Peretroukhin, B. Wagstaff, and J. Kelly, "Deep probabilistic regression of elements of so (3) using quaternion averaging and uncertainty injection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 83–86.

[36] Z. Laskar, I. Melekhov, S. Kalia, and J. Kannala, "Camera Relocalization by Computing Pairwise Relative Poses Using Convolutional Neural Network," in *Proceedings - 2017 IEEE International Conference on Computer Vision Workshops, ICCVW 2017*, vol. 2018-January. Institute of Electrical and Electronics Engineers Inc., jan 2018, pp. 920–929.

[37] V. Balntas, S. Li, and V. Prisacariu, "Relocnet: Continuous metric learning relocalisation using neural nets," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 751–767.

[38] S. Saha, G. Varma, and C. V. Jawahar, “Improved visual relocalization by discovering anchor points,” in *BMVC*, 2018.

[39] G. L. Oliveira, N. Radwan, W. Burgard, and T. Brox, “Topometric localization with deep learning,” *ArXiv*, vol. abs/1706.08775, 2017.

[40] A. Valada, N. Radwan, and W. Burgard, “Deep Auxiliary Learning for Visual Localization and Odometry,” in *Proceedings - IEEE International Conference on Robotics and Automation*. Institute of Electrical and Electronics Engineers Inc., sep 2018, pp. 6939–6946.

[41] X. Li, S. Wang, Y. Zhao, J. Verbeek, and J. Kannala, “Hierarchical scene coordinate classification and regression for visual localization,” 2019.

[42] E. Brachmann and C. Rother, “Expert sample consensus applied to camera re-localization,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 7525–7534.

[43] K. He, X. Zhang, S. Ren, and J. Sun, “Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, sep 2015.

[44] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *International Conference on Learning Representations*, 12 2014.