HyperPose: Hypernetwork-Infused Camera Pose Localization and an Extended Cambridge Landmarks Dataset
Supplementary Material

Ron Ferens Yosi Keller
Faculty of Engineering, Bar Ilan University, Ramat-Gan, Israel
{ronferens, yosi.keller}@gmail.com

1. Extended Cambridge Landmarks Dataset

1.1. Data Generation

The Extended Cambridge Landmarks (ECL) dataset augments the Cambridge Landmarks [3] benchmark with synthetic seasonal and lighting variations using InstructPix2Pix [2]. By applying an adjusted guidance scale and customized prompts, the generated query images extend the original test scenes with three distinct variations: *Summer*, *Winter*, and *Evening*. Each variation introduces expected environmental modifications; for instance, in the *Winter* setting, roads and pavements are partially covered with snow, while in the *Evening* setting, streetlights are illuminated, and the sky appears darker. Visual samples of the ECL dataset can be found in the supplementary materials. Figure 1 shows samples from each scene in the extended dataset. Quantitative pose regression analysis between these more challenging conditions and unmodified sequences helps scientifically assess model adaptability.

1.2. Results

To assess the robustness improvement from integrating a hypernetwork into the Baseline APR, we compare model accuracy on the ECL dataset. Consistent with Table 2 in the main paper, Table 1 reports results on the Extended Cambridge Landmarks (ECL) dataset **without retraining or fine-tuning** (models originally trained on the Cambridge Landmarks dataset). Notably, the Baseline APR w/ hypernet results in the original column of Table 1 align with those in Table 2 of the main paper. As shown, hypernetwork-enhanced models consistently outperform the original, exhibiting lower deviation in pose estimation error compared to the Baseline APR.

2. Oxford RobotCar Dataset

2.1. Datasets Split

This expansive outdoor corpus captures over 100 traversals of a consistent route, that exhibits substantial appearance variations. These are induced by changes in weather, traffic patterns, pedestrian activity, and longitudinal alterations such as construction.

Our experiments utilize two subsets from the Oxford RobotCar Dataset denoted LOOP and FULL, following the training and evaluation protocols introduced in [1, 9]. The LOOP scene encompasses 1120 meters total, while the FULL scene spans a total distance of 9562 meters. Table 2 details the training and testing sequences of the Oxford RobotCar dataset and their attributes.

2.2. Results Trajectories

Figure 2 depicts the predicted trajectories of the AtLoc architecture [9], MapNet [1] and the suggested method evaluated in the Oxford RobotCar scenes LOOP1 (top), LOOP2 (middle), and FULL1 (bottom).

3. Training Details

We adopt the three-phase training procedure introduced in [6]. In the first phase, all network components are trained simultaneously. In the second phase, only the translation-related layers are fine-tuned, including the hypernetwork and regression layers, while other parts of the network remain frozen. In the final phase, the orientation-related components are fine-tuned to improve performance without compromising the localization objectives. Specifically, during the first optimization phase, we set the internal loss parameters S_x and S_q to -3 and -6.5, respectively, for the Cambridge outdoor dataset and 0.0 and 0.0 for the 7 Scenes indoor dataset. In the second and third phases, we set the optimized parameters to 1.0 and 0.0.

We used the Adam optimizer with $\beta_1 = 0.9, \beta_2 = 0.999$, and $\epsilon = 10^{-10}$. The initial values of the loss parameters are set based on the characteristics of each dataset. The batch size was 8, and the initial learning rate was $\lambda = 10^{-4}$. For single-scene models, the learning rate is reduced by 25% every 20 epochs for indoor localization and every 200 epochs for outdoor localization, with maxima of 100 and

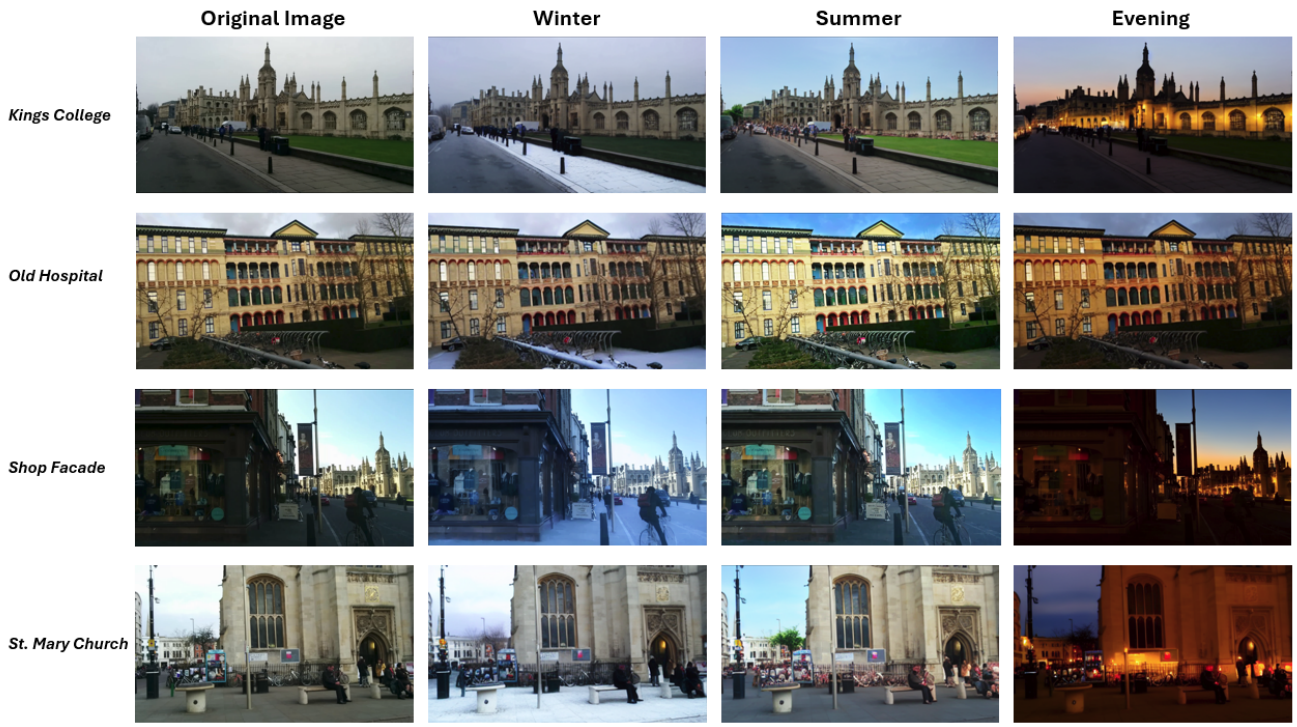


Figure 1. Samples from the proposed Extended Cambridge Landmarks (ECL) Dataset.

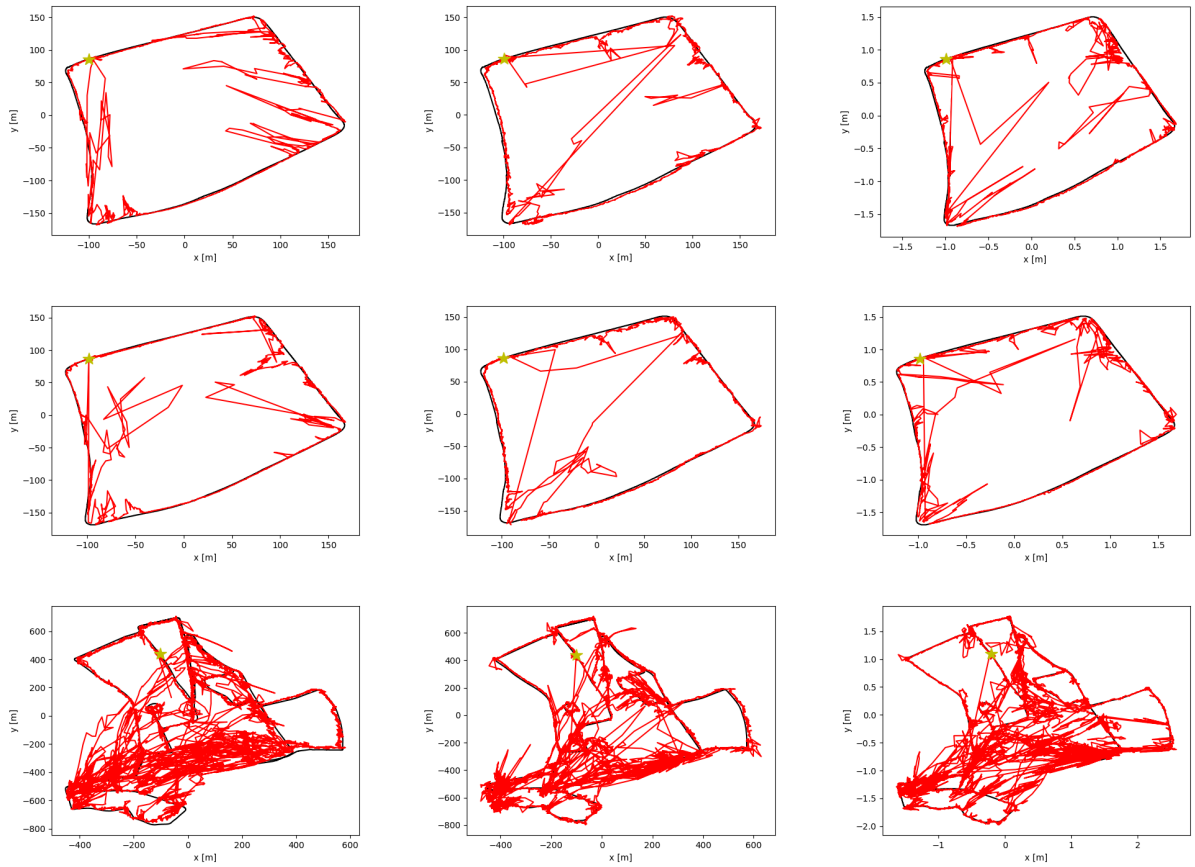
Table 1. **Comparative analysis of the Baseline APR architecture with and without the proposed hypernetwork using the Extended Cambridge Landmarks (ECL) dataset:** We report the median position/orientation error in meters/degrees. **Bold** marks the best performance.

Scene	Method	Flavor			
		original	**winter**	**summer**	**evening**
K. College	Baseline APR	0.89, 2.29	0.94, 2.14	1.03, 2.12	1.15, 2.59
	Baseline APR w/ hypernet	**0.61, 1.84**	**0.67, 2.03**	**0.69, 1.93**	**0.70, 2.52**
Old Hospital	Baseline APR	1.49, 3.30	1.84, 3.72	1.53, 3.77	2.03, 3.43
	Baseline APR w/ hypernet	**1.44, 3.03**	**1.39, 3.15**	**1.39, 3.36**	**1.77, 3.40**
Shop Facade	Baseline APR	0.74, 4.79	0.74, 5.06	0.76, 4.90	1.04, 4.99
	Baseline APR w/ hypernet	**0.70, 3.62**	**0.71, 4.26**	**0.75, 4.08**	**0.88, 4.90**
St. Mary	Baseline APR	1.40, 4.95	1.50, 5.26	1.52, 5.45	1.63, 5.55
	Baseline APR w/ hypernet	**1.37, 4.85**	**1.37, 5.07**	**1.48, 5.35**	**1.53, 5.34**

600 epochs, respectively. For multi-scene models, due to the large number of samples comprising all dataset scenes, we apply a 25% learning rate reduction every 10 epochs for indoor localization and every 200 epochs for outdoor localization, with maxima of 30 and 600 epochs, respectively. Additionally, a weight decay of 10^{-4} and a dropout of $p = 0.1$ are applied to the Transformers during training.

To improve the model's generalization, we applied augmentation in line with [3]. During training, the image is resized so that its smaller edge is resized to 256 pixels and a random 224×224 crop is taken. For the Cambridge dataset, we also apply random adjustments to the brightness, contrast, and saturation of the image. During testing, the image is rescaled so that the smaller edge is resized to 256 pixels, and a center crop is taken without any further augmentation.

The models were trained on a single NVIDIA $2080Ti$ GPU and the PyTorch framework [5]. The training time for a single-scene model ranges from 0.5 to 4 hours, depending on the size of the selected scene. Multi-scene model training requires approximately 12 hours for the indoor dataset



(g) MapNet (h) AtLoc (i) AtLoc w/ hypernetwork

Figure 2. The LOOP1 (top), LOOP2 (middle) and FULL1 (bottom) trajectories of the Oxford RobotCar dataset. The ground truth trajectories are illustrated by black lines. Red lines show the corresponding predictions, where stars denote the starting points in each trajectory.

Table 2. The Oxford RoborCar testing and training sequences.

Sequence	Time	Tag	Mode
	2014-06-26-08-53-56	overcast	Training
	2014-06-26-09-24-58	overcast	Training
LOOP1	2014-06-23-15-41-25	sunny	Testing
LOOP2	2014-06-23-15-36-04	sunny	Testing
	2014-11-28-12-07-13	overcast	Training
	2014-12-02-15-30-08	overcast	Training
FULL1	2014-12-09-13-21-02	overcast	Testing
FULL2	2014-12-12-10-45-15	overcast	Testing

and 24 hours for the outdoor dataset. We select the optimal epoch for the final trained model based on an evaluation of the overall performance for both position and orientation estimation errors.

4. Ablation Study

Due to the significant volume of required experiments, we compare the architecture variations after the first training stage (see Section 3), rather than after the full training. Hence, the reported estimation errors may differ from those in Tables 1, 2, 3, 4 and 5 in the main paper, which reflect full training.

MS-HyperPose Inputs. We evaluated the impact of the input configuration on the performance of the MS-HyperPose architecture. As depicted in Fig. 2 in the main paper, the hypernetwork input for each branch consists of a summation of the embedded activation maps from the CNN backbone and the output of the branch's Transformer. Table 3 compares the overall performance of the model with different input configurations using the Cambridge landmarks dataset. We note that incorporating the backbone embedding improves both positional and rotational errors.

Residual hypernetwork output. Table 4 shows the re-

Table 3. Ablations MS-Hyperpose inputs. We report the median position and orientation errors on the Cambridge Landmarks dataset.

Input	Position [meters]	Orientation [degrees]
Transformers tokens	1.34	2.52
Backbone embedding + Transformers tokens	**1.33**	**2.41**

gression results of the absolute pose versus the residual outputs in the hypernetwork branches. This analysis supports using residual position and orientation outputs, as depicted in Fig. 2 in the main paper. In the integrated layer methods, the regressed pose of the hypernetwork weights is added to the main network regression output after each layer. Such that the next layer sums up the previous main network layer and the corresponding hypernetwork outputs. In the residual approach, only the final output of the main network and hypernetwork regression head are summed to obtain the final pose estimate. The residual-based architecture outperforms integrated approaches, achieving lower errors in both position and orientation.

Table 4. Ablation study of the network residual architecture. We report the overall median position and orientation errors on the Cambridge Landmarks dataset.

Hypernetwork infusion method	Position [meters]	Orientation [degrees]
Single integrated hyper-layer	1.40	2.80
Two integrated hyper-layers	1.29	2.76
Residual hyper-layers	1.33	2.41

Hypernetwork embedding dimension. We assess varying the size of the fully-connected layers in both the position and orientation branches of the regression heads in MS-HyperPose. Table 5 presents multi-scene results obtained by altering the layer dimensions. We compare the overall median position and orientation errors across the Cambridge and 7 Scenes datasets to understand the impact of embedding size. Notably, modifying one branch's dimension impacts the correlated task, as well as the other branch. For example, reducing the orientation embedding to $\mathbb{R}^{256}$ increases both position and orientation errors. Increasing the dimensionality to $\mathbb{R}^{512}$ shows a greater improvement in orientation at the expense of the position accuracy. Table 6 indicates this trend also applies to single-scene APR, suggesting a correlation between the tasks warranting further exploration.

Table 5. Ablation study of the embedding dimension regressed by the hypernetworks in the suggested multi-scene MS-HyperPose architecture. We report the median position and orientation errors for the Cambridge and 7 Scenes datasets.

Embedding Dimensions $\theta_h^{Position}/\theta_h^{Orientation}$	Cambridge	7Scenes
256/256	1.40,2.88°	0.18,8.17°
256/512	**1.34,2.52°**	**0.18,7.11°**
512/512	1.37,2.38°	0.21,6.44°

Table 6. Ablation study of the proposed single-scene baseline APR embedding dimension regressed by the hypernetworks. We report the median position and orientation errors on the *Red Kitchen* scene in the 7 Scenes datasets.

Embedding Dimensions $\theta_h^{Position}/\theta_h^{Orientation}$	Position [meters]	Orientation [degrees]
256/256	0.18	9.01
256/512	**0.17**	**8.79**
256/1024	0.18	9.00

Rotation representations. The authors of [10] contend that 3D and 4D rotation representations are suboptimal for network regression, whereas 5D and 6D continuous representations are more appropriate for learning. As shown in Table 7, employing a 4D-based representation in conjunction with the rotational loss in Eq. 4 in the main paper, led to the lowest orientational error. In both the Quaternions and 4D-Norm approaches, the rotation regressor generates four values that estimate the camera's orientation (quaternion), but the latter computes the orientation loss based on rotation matrices instead. The results refer to the overall performance of the MS-HyperPose architecture on the Cambridge Landmarks dataset.

Table 7. Ablation of the 3D rotation encoding. We report the median orientation errors on the Cambridge Landmarks dataset.

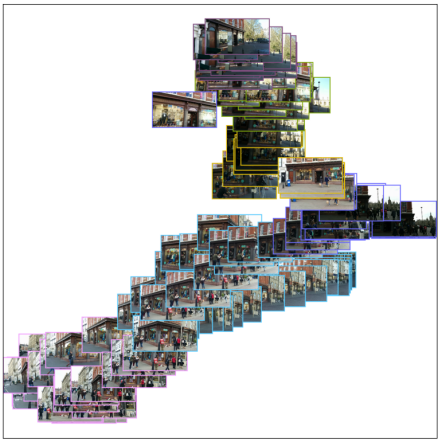
Rotation representation	Position [meters]	Orientation [degrees]
Quaternions	**1.33**	**2.41**
6D [10]	1.40	2.52
9D [10]	1.45	2.62

5. Hypernetwork Output Weights

Given an input query image I_q, hypernetworks generate weights for the position and orientation regression heads, allowing adaptation of the network output based on visual features in the image. The image is likely to exhibit a range of attributes, such as varying illumination conditions, different viewpoints, dynamic objects, and changing backgrounds. Thus, it is expected that the generated weights will be adapted to accommodate these fluctuations so that images from the same scene (location) will be jointly clustered despite the appearance variations, as shown in Fig. 3. Figure 3a illustrates the clustering of the Shop Facade scene test set from the Cambridge Landmarks dataset, based on the ground-truth translation data. The K-Means algorithm, with ($K = 6$), was applied, and images were color-coded according to their assigned clusters. The clusters were formed based on spatial proximity within the scene's coordinate system. Figures 3b and 3c depict the K-Means clustering ($K = 6$) of the 2D t-SNE [8] projections of the backbone's embeddings generated by the Baseline APR and HyperPose networks. Notably, the embeddings produced by HyperPose provide a more coherent visual representation of the scene compared to those from the Baseline APR. As shown in Figure 3b, the Baseline APR embeddings are significantly influenced by variations in lighting conditions, leading to query images from different locations being projected close to each other. In contrast, HyperPose demonstrates improved robustness to lighting variations, preserving a more accurate spatial representation of the scene.

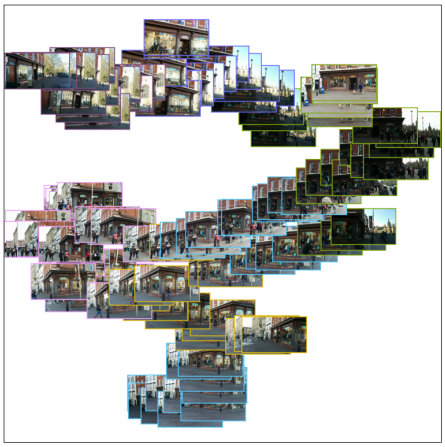
6. Model Robustness

Figure 4a illustrates the improvement in translation error between AtLoc and its hypernetwork-enhanced variation across varying environmental conditions, presented on the RobotCar's LOOP scene. While both models achieve similar performance under simple settings (1,3), the hypernetwork-based model demonstrates improved accuracy under more difficult conditions of extreme lighting (4,5), shifted viewpoints (2,5), and unmapped moving objects (2,4). Figure 5 compares the translation error between the Baseline and hypernetwork-enhanced models on a sequence from the Kings College scene in Cambridge Landmarks. This example demonstrates the improved robustness to occlusion of the hypernetwork-based approach. Specifically, when a passing car temporarily obscures the camera's view, the Baseline error spikes over 10x while the hypernetwork-based model maintains reasonable accuracy. Across the datasets, similar analysis consistently reveals the improved resilience conferred by conditional parameter modulation.



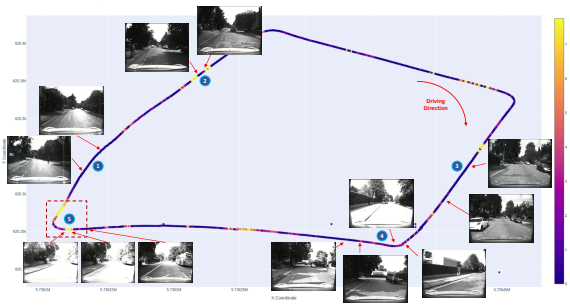
(a) Clustering by translation

(b) Clustering by Baseline's embeddings

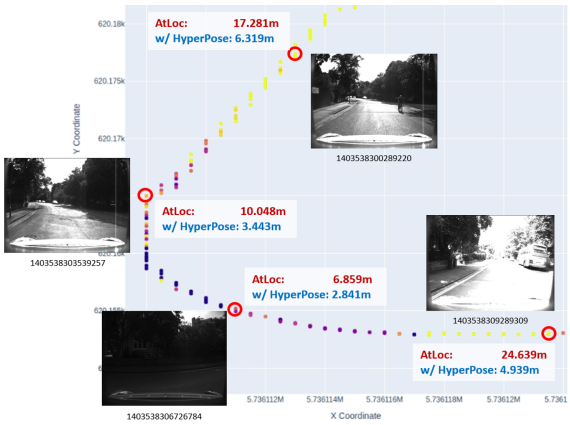


(c) Clustering by Hyperpose's embeddings

Figure 3. The clusters of the Shop Facade test set images, that are part of the Cambridge Landmarks dataset. The clusters are computed using K-Means with $K = 4$. (a) Clustering of the corresponding camera positions (X, Y). (b) Clustering of the 2D t-SNE projections of the regression weights computed by the hypernetwork.



(a) Positional clusters.



(b) Hypernetworks weights clusters.

Figure 4. Relative translation error of AtLoc [Red] and AtLoc with hypernet [Blue] on the RobotCar's 'loop' scene. Brighter colors encode higher differences and more significant accuracy improvements. These are shown near the corners, where the appearance changes significantly between frames of nearby locations.

7. Model Extension and Transfer Learning

In the context of multi-scene pose regression, it is important to consider scenarios where the model must adapt to an additional scene. We assess the advantage of incorporating a hypernetwork into a multi-scene APR by analyzing its impact on the transfer-learning required to infer the camera pose in the new scene. In real-world applications, obtaining precise ground-truth mapping for a new scene is a complex, resource-intensive, and costly task. Therefore, we evaluate the model's performance under conditions where only a limited amount of data is available for fine-tuning. Figure 6 compares the translation and orientation errors of a selected multi-scene APR model with and without hypernetwork. Specifically, the comparison focuses on the models' accuracy after 2-5 epochs of fine-tuning using 5%, 10%, 25%, 50%, and 75% of a new scene's training data. In the initial phase, we train a multi-scene model utilizing the 7-Scene dataset while omitting the "Heads" scene. In the subsequent phase, we implement a transfer learning procedure, adapting the model with only a subset of the training data from the "Heads" scene. As shown in the figure, the hypernetwork-based model consistently is more accurate compared to the original model across all data limitations. Moreover, the hypernetwork-based model achieves competitive pose error using only 25% of the available training data.

8. Limitations and Future Work

HyperPose improves accuracy for both novel and existing architectures (e.g. [7], [9]), with minimal impact on inference time (See ***Results*** section in the main paper). However, the increase in model size is attributed to the high output dimensionality of the hypernetworks. Specifically, this expansion arises from the substantial number of parameters required by the regression layers within the hypernetwork.

A linear layer necessitates $N_p = (C_{in} + 1)C_{out}$ parameters. To generate the weights of such layer, a corresponding hypernet outputs requires $(C_{in} + 1)N_p$ parameters. For example, in the case of MS-HyperPose, a layer that receives and outputs R^{256} requires $N_p = 65792$ parameters. Thus, the layer's size is $((256+1)\cdot 65792)\cdot 4bytes = 67,634,176bytes = 66MB$. Therefore, the total memory size of all six hypernet layers is $66+34+0.5+135+270+2.1 = 507.6MB$

However, the model sizes using HyperPose are reasonable for the current architectures. Specifically, even this enlarged multi-scene variant is close to that of the moderately-sized VGG19 [4].

The observed accuracy improvement isn't solely related to the increased model size, as shown in the ablation study (Supplementary materials). Varying the hypernetwork's embedding size and the parameters for 3D rotation produced larger models, yet meticulous architecture selection consistently influenced accuracy across experiments. In our early experiments, we considered a Transformer-Encoder over the current MLP. Despite a higher parameter count, it achieved an average performance of $0.87m, 3.43°$ on Cambridge and $0.17m, 9.09°$ on 7Scenes. This highlights that while enlarging models positively impacts translation accuracy, it can compromise orientation accuracy.

Future work will explore exploiting shared position and orientation weights to reduce the hypernetwork's size.

References

[1] Samarth Brahmbhatt, Jinwei Gu, Kihwan Kim, James Hays, and Jan Kautz. Geometry-aware learning of maps for camera localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1

[2] T. Brooks, A. Holynski, and A. A. Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18392–18402, Los Alamitos, CA, USA, 2023. IEEE Computer Society. 1

Figure 5. Occluded pose regression on Cambridge's Kings College scene. The accuracy improvement in the localization of the middle frame (#66), where a significant occlusion is present, is significant.

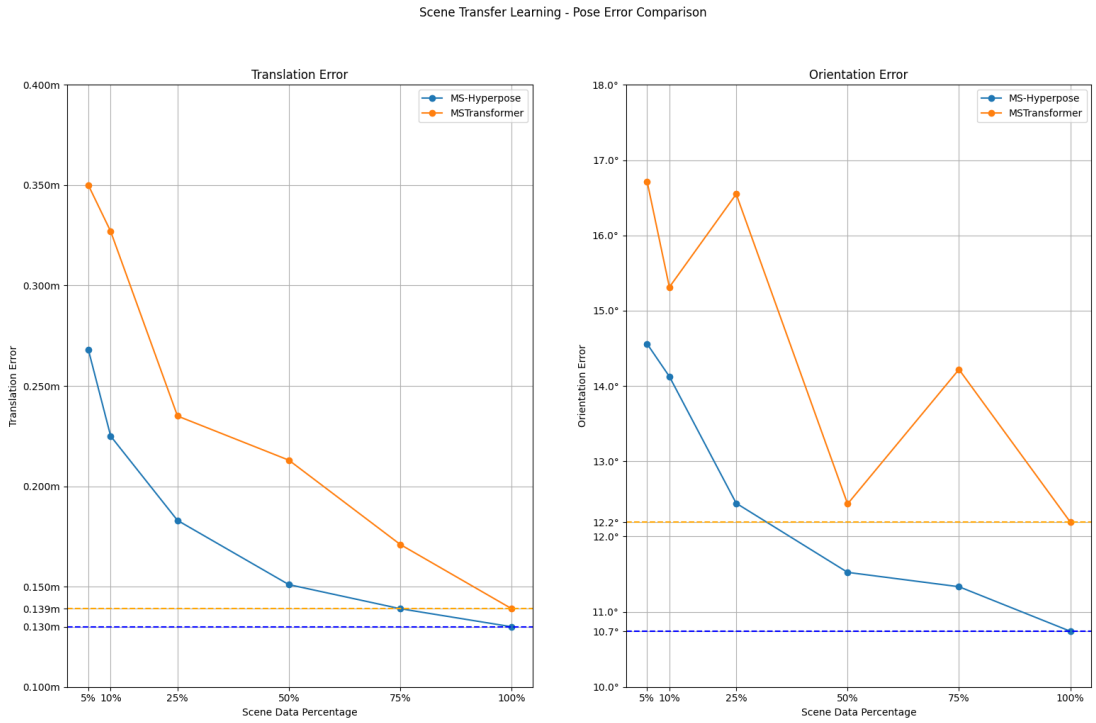


Figure 6. Comparison of multiscene APR with and without the integration of hypernetwork.

[3] 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)*, pages 2938–2946, 2015. 1, 2

[4] S. Liu and W. Deng. Very deep convolutional neural network based image classification using small training sample size. In *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, pages 730–734, 2015. 6

[5] et al. Paszke, Adam. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NIPS)*, pages 8026–8037, 2019. 2

[6] Yoli Shavit and Ron Ferens. Do we really need scene-specific pose encoders? In *Proceedings of the International Conference on Pattern Recognition (ICPR)*, pages 3186–3192. IEEE, 2021. 1

[7] Yoli Shavit, Ron Ferens, and Yosi Keller. Learning multi-scene absolute pose regression with transformers. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021. 6

[8] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9 (11), 2008. 5

[9] Bing Wang, Changhao Chen, Chris Xiaoxuan Lu, Peijun Zhao, Niki Trigoni, and Andrew Markham. Atloc: Attention guided camera localization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10393–10401, 2020. 1, 6

[10] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5745–5753, 2019. 4