MCAPR: Multi-modality Cross Attention for Camera Absolute Pose Regression

Qiqi Shu, Zhaoliang luan, Stefan Poslad, Marie-Luce Bourguet, and Meng Xu(✉)

Queen Mary University of London, London E1 4NS, UK
meng.xu@qmul.ac.uk

Abstract. Absolute camera pose regression typically estimates the position and orientation of a camera solely based on the captured image, trained with a convolutional backbone with multilayer perceptron heads for a single reference scene only. Recently, leading pose regression results have been achieved on multiple datasets by extending this approach to learn multiple scenes by leveraging data from different modalities, especially by fusing RGB and point cloud data. In this work, we propose to use cross-attention Transformers to learn multi-scene absolute camera pose regression, where cross-attention modules are used to aggregate activation maps with self-attention from different data modalities and convert latent features and scene index into candidate pose predictions. This mechanism allows our model to focus more on the general localization features. We evaluate our approach on the popular indoor benchmark dataset 7-scenes and compare it against both state-of-the-art (SOTA) single-scene and multiple absolute pose regression models. Our main result is the rotation accuracy is improved using our method more than slightly improved position accuracy compared to existing multi-scene methods.

Keywords: Absolute Pose Regression · Transformer · Multi-modality

1 Introduction

There are key indoor vision-based applications such as automating care and health of an increasing elderly, physically less-able population at home. Computer vision here needs to handle multiple rather than single views of the same scene, e.g., because objects may be placed in different locations and orientations after and when they are used by humans. For these, determining the position and orientation of the camera using multimodal sensors is critical. Current camera localization techniques involve various precision, runtime, and memory usage trade-offs. The most accurate visual localization method [15,20] involves hierarchical localization pipelines that generally employ a coarse-to-fine processing stream. In general, to estimate the camera's six degrees of freedom (6DoF) accurately, it is necessary to map the 2D-2D correspondences to 2D-3D, the pose

L. Iliadis et al. (Eds.): ICANN 2023, LNCS 14255, pp. 434–445, 2023.
https://doi.org/10.1007/978-3-031-44210-0_35

of the camera is then determined via the perspective n-point (PnP) algorithm and Random Sample Consensus (RANSAC). Other approaches based on visual absolute pose regressors (APRs) have been proposed, which can estimate camera pose with only the query image through a single forward pass, significantly reducing latency, albeit with significantly lower accuracy. Furthermore, because of its small memory footprint, APRs can be deployed as stand-alone applications on edge devices with limited computing resources. [23] gives an exhaustive survey of visual camera pose estimation.

Blanton et al. [9] proposed a method that extends APR from a single scene to multiple scenes. This method is similar to existing APRs that utilize a CNN backbone to generate the image's global latent descriptor. This approach trains a set of fully connected layers (FC), one for each scene, indexed by the predicted scene identifier. While providing a new generic framework that can optimize a single model for multi-scenes, this method falls short in matching the precision of SOTA APRs. Shavit et al. [17] proposed a new multi-scene absolute pose regression formula named MS-tran based on the recent successful applications of self-attention transformer in computer vision tasks such as image recognition [5] and object detection [17]. MS-tran showed the effectiveness of encoder self-attention in aggregating informative latent features for a specific task. Furthermore, this method was shown to generate multiple predictions that can correspond to independent queries based on the self-attention mechanism.

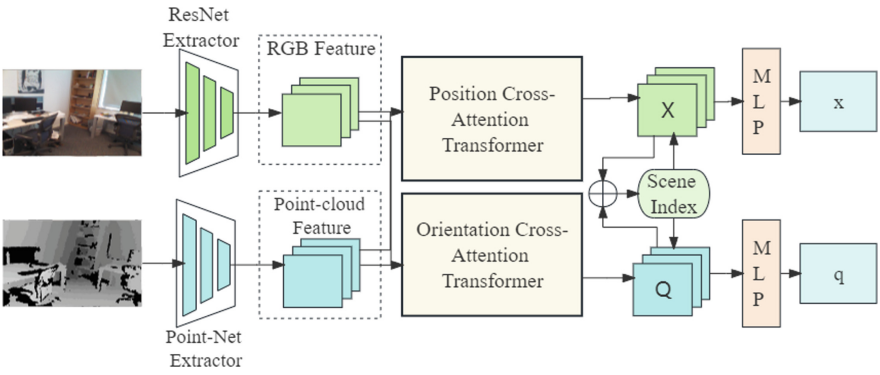

Fig. 1. Overview of the multi-modality cross attention model architecture.

In this work, drawing inspiration from the work of MS-tran, we apply the transformer module to the task of multi-scene absolute pose regression. Specifically, we fuse visual and depth information and transform the encoded scene index into a latent pose representation, as illustrated in Fig. 1. Our approach differs from general APR methods which learn using a single global image encoding to estimate the camera's 6DoF pose [21]. Instead, we propose a multi-modality sensor fusion approach based on cross-attention, as the data varies greatly from different modalities, therefore, we utilize two separate extractors for RGB and depth feature extraction. As camera pose estimation involves two distinct sub-tasks (position

and orientation estimation). We use two separate cross-attention transformer modules, each dedicated to one of the sub-tasks. The outputs of the transformer are used for scene classification and selecting embedding respectively, then we regress the position and orientation feature vector. By providing an adaptive analysis of the point cloud and image feature, the proposed architecture based on dual sensor fusion and a cross-attention mechanism aims for a robust and accurate model when applied to a contemporary indoor scenes benchmark.

In summary, our main contributions are as follows:

- We propose a novel architecture for fusing multi-modality data from different sensors and performing multi-scene APR using cross-attention transformers.
- We demonstrate through experimentation that self-attention and cross-attention can aggregate position and rotation image features.
- The rotation accuracy is improved using our method more than slightly improved position accuracy compared to existing both single and multi-scene APR methods.

2 Related Work

2.1 Image Pose Retrieval

3D or Structure-Based Localization methods are recognized to have the best precision. Hierarchical pose estimation pipelines have introduced a two-stage approach to this process. Firstly, a CNN trained for image retrieval (IR) is used to encode each query image, and then 2D image features are matched with 3D coordinates to estimate the camera pose using PnP-RANSAC. The obtained matches are subsequently utilized to estimate camera pose. DSAC++ [2] and DSAC [3], leverage CNN architectures to estimate 3D coordinates from 2D positions directly from the image, requiring only one image during querying. Like absolute pose regression methods, in order to achieve state-of-the-art accuracy, it is necessary to train a distinct model for each scene.

Relative Pose Regression (RPR). Algorithms based on RPR estimate the relative pose of an input query image and a group of reference images by utilizing known positions. The process of calculating relative pose entails estimating the relative position and orientation between the reference images and the query image [1,10]. The closest neighbouring set of images is determined using image retrieval (IR) methods, and the relative translation and rotation between the query image and each retrieved image is estimated. The estimated relative pose and known position of the reference images are utilized to determine the absolute pose. Similar to 3-D and structure-based localization techniques, the RPR algorithm is a multi-step process that necessitates access to a database of annotated images during querying.

Absolute Pose Regression (APR). PoseNet by Kendall et al. [9] is the first APR method that uses a convolutional neural network backbone and multi-layer perceptron head to regress the camera's position and orientation.

This method is simple and can run in 5ms. Subsequently, several APR methods have achieved good results by modifying PoseNet's backbone and MLP architecture [11,12,16,20–22], as well as alternative loss functions and optimization strategies [15]. However, since indoor environments exhibit dynamic scenes characterized by different lighting conditions and viewpoints, relying solely on visual information from RGB cameras may limit the network's adaptability and reduce the accuracy.

2.2 Transformer

Initially developed for machine translation, the Transformer architecture has emerged as the SOTA approach in natural language processing. More recently, there have been efforts to introduce Transformer-like networks to various computer vision tasks, including object detection, image recognition, segmentation, and visual question answering (VQA) [18,19,24] . The Vision Transformer (ViT) [5] directly inputs image patches into a Transformer used for image classification, eliminating the need for any convolutional operations. Our work builds upon these studies, demonstrating the effectiveness of Transformers module in fusing multi-modality information for precise pose estimation. Specifically, to the best of our knowledge, we are the first to propose a multi-modality cross-attention Transformer that adjusts the weights of the multi-heads attention based on the input image and point cloud.

3 Method

Generating stronger representations by matching point clouds and images has been a long-standing challenge due to the inherent differences in features between RGB images and point clouds. In this research, a novel multi-modality cross-attention network is proposed that models both intra- and inter-modal relationships between image regions and point cloud data in a single deep model, allowing for image-point cloud matching and information fusion. To ensure robust cross-modality matching and fusing, two effective attention modules, namely self-attention and cross-attention modules, are designed, each of which plays a critical role in modelling the intra- and inter-modal relationships.

Self-attention and cross-attention modules are illustrated in the green solid line block and the black dashed line block in Fig. 2, respectively. Given a pair of RGB and depth images, we first extract image features by using the Resnet-50 model, and simultaneously use a pre-trained Point-net to extract point cloud features, while the point cloud was generated by depth image with intrinsic parameters. Based on these granular representations of image and point clouds, we model the intra-modal relationships using the self-attention module and model the intra-modal relationships between image and point clouds using the cross-attention module. To handle multiple different scenes, we also apply separate position and orientation transformers modules to encode the localization parameters for each scene. The architecture is based on the DETR method [4] with

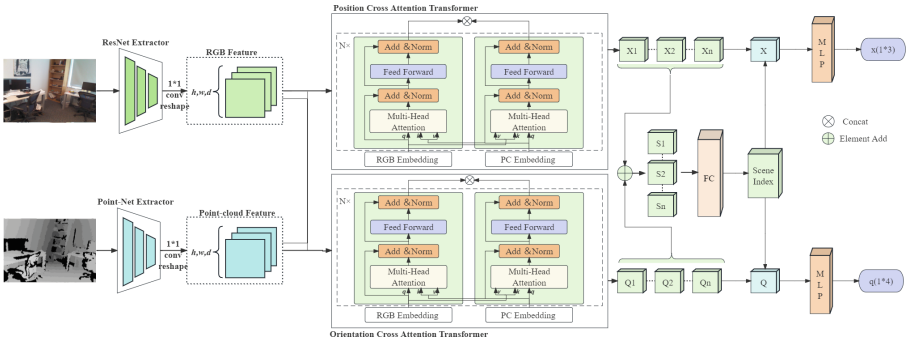

Fig. 2. Multi-modal cross-attention mechanism for absolute pose regression in multiple scenes. The proposed approach consists of two separate cross-attention transformer modules, each of which processes information from RGB and depth modalities, respectively, and outputs features for position and orientation. Afterwards, the scene encoding is integrated with the features from both modalities and softmax processing is applied to obtain a single scene index. Finally, the index and the encoded features are used to regress the position x and orientation q, respectively.

MS-tran [17]. The cross-attention Transformer architecture allows for learning multiple scenes simultaneously while attending to local information in the image and point cloud content. To obtain accurate camera poses, we classify scenes by concatenating $\{X_i\}_0^m$ and $\{Q_i\}_0^m$, then the index of the final detected scene X_i, Q_i is obtained by Soft-Max process. Finally, the final predicted x and q are obtained by separately encoding the indexed X and Q using MLP. It is worth mentioning that in the training phase, we use hard negative mining to construct the triplet loss to optimize the parameters of the model.

3.1 Overall Architecture

Our model architecture is illustrated in Fig. 2. Given a pair of RGB images and a depth map as inputs, we first convert the depth map into a corresponding point cloud by calculating the camera intrinsic and extrinsic matrices. In the feature extraction module, we employ a ResNet model to extract features $F_r \in \mathbb{R}^{D_m \times H_q \times W_q}$ of salient regions $I_r \in \mathbb{R}^{3 \times H \times W}$ in the images. Meanwhile, we use PointNet as the point cloud feature encoder to obtain point cloud features $F_p \in \mathbb{R}^{D_m \times H_q \times W_q}$ from point cloud $P_i \in \mathbb{R}^{3 \times N_i}$. We denote $\phi(F_r) \in \mathbb{R}^{D_m \times H_t \times W_t}$ and $\phi(F_p) \in \mathbb{R}^{D_m \times H_q \times W_q}$ as the feature maps extracted from RGB and point cloud. After that, we use a 1×1 convolution to compress the number of channels of $\phi(F_r), \phi(F_p)$ from 1024 to $D_m = 512$. The two features are then further aggregated using the CAT module, which incorporates a cross-attention mechanism, as defined in the following formula::

$$(F_x, F_q) = \mathrm{CAT}(\phi(F_r), \phi(F_p)), \tag{1}$$

where F_x and F_q represent the output embedding from the position transformer and orientation transformer.

Then, we independently feed the image region and point cloud feature region into different cross-attention Transformer units to model the intra-modality relationships. Subsequently, the global representation can be obtained by aggregating these fragment features. To perform index-based classification, the F_x features and the F_q features are first subjected to global average pooling (GAP) before being concatenated and used as input to the classifier ϕ_c. The classification results $P(scene_i), i = 1, 2, ..., n$ can be formulated as:

$$P\left(scene_i\right) = \Phi_c\left(\text{Concat}\left(\text{GAP}\left(\psi\left(F_i, p_i\right)\right)\right)\right) \tag{2}$$

Subsequently, the output vectors corresponding to the highest probability at the scene index are selected. These output vectors, represented as (X_i, Q_i), are then fed into their respective MLP heads with one hidden layer and Gaussian Error Linear Unit (GELU) non-linearity to perform regression on the target vectors, namely x or q.

3.2 Cross-Attention and Self-attention

The self-attention mechanism is a process that involves mapping a set of key-value pairs and a query to produce an output. The output is calculated as the weighted sum of the values, where the weight matrix is determined based on the query and its corresponding key. The multi-head self-attention is accomplished by applying different learnable linear projections to the query (Q), key (K), and value (V) h times.

Specifically, given a set of sequences$\{F_i\}_1^n$, we first compute the input Q_F, K_F, and V_F, respectively. Then, the weight matrix can be obtained through "Scaled Dot-Product Attention". The weighted sum of the values is then computed as follows:

$$\text{Attention}\left(\mathbf{Q}_F, \mathbf{K}_F, \mathbf{V}_F\right) = \text{softmax}\left(\frac{\mathbf{Q}_F \mathbf{K}_F^T}{\sqrt{d_k}}\right)\mathbf{V}_F \tag{3}$$

The Head is then calculated by the following equation:

$$\text{head}_{\,i} = \text{Attention}\left(\mathbf{F}\mathbf{W}_i^Q, \mathbf{F}\mathbf{W}_i^K, \mathbf{F}\mathbf{W}_i^V\right), \tag{4}$$

Afterwards, the values of all the heads are computed and concatenated together as follow:

$$\text{MultiHead}\left(\mathbf{F}\right) = \text{concat}\left(\text{head}_{\,1}, \ldots, \text{head}_{\,h}\right)\mathbf{W}^O, \tag{5}$$

After the MHA operation, there is a Feed-forward Network (FFN) module composed of two linear transformation layer with ReLU activation, defined as:

$$\text{FFN}(x) = \max\left(0, xW_i + b_i\right)W_j + b_j \tag{6}$$

where $W_i, W_j$and $b_i, b_j$are the weight matrices and basis vectors respectively. In order to fuse the image feature and point cloud feature, in one stream of CAT, we let $Q = F_i$and $K = V = F_p$ in equation (3), and obtain the aggregated target feature. This procedure can be summarized as:

$$\widetilde{F}_t = \text{Norm}\left(F_t + P_t + \text{MultiHead}\left(F_t + P_t, \quad F_q + P_q, F_q\right)\right) \tag{7}$$

$$Y_t = \text{Norm}\left(\widetilde{F}_t + FFN\left(\widetilde{F}_t\right)\right) \tag{8}$$

where P_t, P_q, are the spatial position encodings corresponding to Ft and Fq, respectively. In another stream, we set $Q = Fq$ and$K = V = Ft$ and generate Y_q, then aggregate the query feature. The entire computation described above represents one layer of the proposed cross-attention Transformer. The output of one layer serves as the input of the next layer in the network.

3.3 Multi-scene Camera Pose Loss

The pose estimation error is calculated by comparing the ground truth pose $p_{gt} = (x_{gt}, q_{gt})$, where $x \in R^3$ denotes the position and $q \in R^4$ encodes its direction as a quaternion, with the predicted pose $p =< x, q >$. The position error and orientation error denoted as L_x and L_q are computed as the Euclidean distance between the predicted value and ground truth. We train our model to minimize both L_x and L_q

$$\begin{aligned} L_x &= \left\|\mathbf{x}_{gt} - \mathbf{x}\right\|_2 \\ L_q &= \left\|\mathbf{q}_{gt} - \frac{\mathbf{q}}{\|\mathbf{q}\|}\right\| \|_2 \end{aligned} \tag{9}$$

We referred to the method of [8] to balance the various loss functions defined for individual objectives using different weighting schemes. The loss equation is as follows.

$$L_p = L_x \exp\left(-\alpha\right) + \alpha + L_q \exp\left(-\beta\right) + \beta, \tag{10}$$

where α, β are the learned parameters. To further improve the performance of the model, we incorporated a negative log-likelihood (NLL) loss term based on the ground truth scene index s_{gt}. Like the loss function in [17], this loss term is calculated using the estimated pose p and the predicted log probability distribution s of the scene. The overall loss function is:

$$L_{\text{multi-scene}} = L_{\mathbf{p}} + NLL\left(\mathbf{s}, \mathbf{s_{gt}}\right) \tag{11}$$

3.4 Implementation Details

The implementation of our model uses PyTorch [13]. As an encoder for RGB images, we use a pre-trained e-ResNet-50 [7], which decreases the size of the feature maps and increases the number of feature channels. And we have also

used Point-Net [14] as a point cloud encoder to extract sufficient geometry information. We denote F_{rgb} and F_{pc} as RGB and point cloud feature input:$F_{rgb} \in \mathbb{R}^{3\times H_q \times W_q}$ and$F_{pc} \in \mathbb{R}^{3\times H_q \times W_q}$. We set $C = 512$ for the input dimension of the components of the cross-attention transformer. All cross-attention transformers consist of 5 layers with GELU non-linearity. For each layer, we use 4 MHA heads and a 2-layer MLP with a hidden dimension of 256. Then we regress the position and orientation vectors from the latter two MLP heads in the dimension of 1024 with only one hidden layer.

4 Experiments and Results

4.1 Dataset

Our method is evaluated on the 7Scenes datasets [6], which are commonly used for the evaluation of indoor pose regression methods. The 7Scenes dataset consists of seven small-scale scenes of indoor office environments.

4.2 Training Details

To minimize the loss function in Eq. 11, we optimize our model using the Adam optimization algorithm with the following hyperparameters: $\alpha = 0.3$, $\beta = 0.9$. The batch size is 4, and the learning rate is set to $\lambda = 10^{-6}$. We reduced the learning rate by %25 every 150 epochs, with a maximum of 900 epochs. The output of the decoder is selected during training using the ground-truth scene index, whereas the estimated scene index is used only for NLL loss evaluation. The scene index is unknown during the inference process, and we rely on the model's prediction by choosing the index with the highest (log) likelihood. All of the experiments reported were carried out on a 40Gb Nvidia-A100.

Table 1. Comparison to state-of-the-art methods: 7-Scenes. We take the median errors of position and orientation for each method.

Method	Chess	Fire	Heads	Office	Pumpkin	Kitchen	stairs
MSPN[3]	0.09/4.76	0.29/10.5	0.16/13.1	0.16/6.8	0.19/5.5	0.21/6.61	0.31/11.63
MS-tran	0.11/4.66	0.24/9.6	0.14/12.19	0.17/5.66	0.18/4.44	0.17/5.94	0.26/8.45
MCAPR(ours)	0.10/4.55	0.22/8.4	0.14/11.6	0.15/5.7	0.17/4.5	0.16/5.88	0.23/8.03

Table 2. Current APR results for the 7Scenes dataset. We take the average median errors of position and orientation of each method.

MethodAverage	position [meters]	Orientation [degrees]
Single-sceneAPRs		
PoseNet	0.44	10.4
BayesianPN	0.47	9.81
LSTM-PN	0.31	9.86
GPoseNet	0.31	9.95
PoseNet-Learnable	0.24	7.87
GeoPoseNet	0.23	8.12
MapNet	0.21	7.78
IRPNet	0.23	8.49
AttLoc	0.20	7.56
Multi-sceneAPRs		
MSPN	0.20	8.41
MS-tran	0.18	7.28
MCAPR(Ours)	0.16	7.03

4.3 Comparison with Other APRs

Our approach aims to provide a multi-modality APR paradigm that improves upon the accuracy achieved by existing APR methods. Therefore, we compare our method against MSPN and MS-tran, which are the only 2 multi-scene APR method we are aware of, as well as leading single-scene APR methods. Table 1 shows the results obtained by our approach (MCAPR) and MS-tran, as well as MSPN, on the 7Scenes dataset. It is shown that our method(MCAPR) outperforms MSPN and MS-tran in the 7Scenes dataset mostly, reducing both position and orientation errors. We further compare our results with other single-scene APR solutions. Table 2 shows the performance of different APRs, MSPN, MS-tran, and our method . We report the average position and orientation errors of all scenes. It's clear that our method achieves the smallest position and orientation errors. In the scene index classification part, our model achieves 97.5% mean cross-scene prediction accuracy, which is crucial for latter pose regression.

4.4 Ablation Study

In this section, we mainly explore the effect of different data modalities and the Cross-Attention Transformer (CAT) module with different hyper-parameters since this module is the core component of our model.

Table 3. Ablation study of different CAT layers. We take the median errors of position and orientation.

Layers	position [meters]	Orientation [degrees]
3	0.19	7.41
4	0.15	7.33
5	0.16	7.03
6	0.18	8.36

Table 4. Ablation study of different Transformer Dimensions. We take the median errors of position and orientation.

Transformer Dimensions	position [meters]	Orientation [degrees]
64	0.17	8.02
128	0.17	7.76
256	0.16	7.43
512	0.16	7.03

Transformer Architecture. We investigate the performance of a Cross-Attention Transformer with different feature dimensions and numbers of layers. As illustrated in Table 3, we test the results of the CAT module with 3 to 6 layers. The CAT that contains 4 layers achieves the best performance on position error, while on orientation the best result is obtained with 5 layers. It can be seen that only increasing the number of layers does not always improve performance, which may cause overfitting in the orientation perception. It should be noted that even using only 3 layers, our model still has better performance than many other proposed methods. In the remaining experiments, the number of CAT layers is set to 5 as we found it has the best accuracy. We also found that the performance of the model improves with the Transformer dimension, suggesting that larger models may achieve further improvement in localization accuracy (Table 4). In conclusion, regardless of the number of layers and feature dim, our model maintains SOTA position and orientation accuracy compared to other APR solutions (Table 2).

Table 5. Ablation study of different modality. We take the median errors of position and orientation.

modality	position [meters]	Orientation [degrees]
RGB	0.18	7.9
Point cloud	0.35	14.33
RGB with Point cloud	0.16	7.03

Different Modality. Table 3 shows the result with different data modalities. We investigate the performance of this cross-attention module with different data modalities. More specifically, we trained and tested our model with multi-modality(image and point cloud), image only(point cloud was filled with zero), and point cloud only(image was filled with zero) in the same network architecture. The result shows that using only point cloud is not sufficient for the camera pose location task, and only using the RGB information can also outperform many other solutions in Table 5. But above all, fusing RGB and point cloud data together through a cross-attention transformer achieves the best performance.

5 Conclusion

Unlike general methods relying solely on RGB image data for absolute scene pose estimation, this work proposes a novel multi-modal approach using cross-attention transformers for multi-scene absolute pose regression. By encoding both RGB and point cloud data and inputting them into separate transformer cross-attention modules, position and orientation information are obtained. This approach allows for the utilization of multi-modal information and aggregation of non-scene-specific features. Our method achieve state-of-the-art localization accuracy in both single-scene and multi-scene absolute regression methods on indoor datasets.

References

1. Balntas, V., Li, S., Prisacariu, V.: Relocnet: continuous metric learning relocalisation using neural nets. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 751–767 (2018)
2. Brachmann, E., et al.: Dsac-differentiable ransac for camera localization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6684–6692 (2017)
3. Brachmann, E., Rother, C.: Learning less is more-6d camera localization via 3d surface regression. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4654–4662 (2018)
4. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers (2020)
5. Dosovitskiy, A., et al.: An image is worth 16x16 words: transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
6. Glocker, B., Izadi, S., Shotton, J., Criminisi, A.: Real-time rgb-d camera relocalization. In: 2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR), pp. 173–179. IEEE (2013)
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
8. Kendall, A., Cipolla, R.: Geometric loss functions for camera pose regression with deep learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5974–5983 (2017)

9. Kendall, A., Grimes, M., Cipolla, R.: Posenet: A convolutional network for real-time 6-dof camera relocalization. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2938–2946 (2015)
10. Laskar, Z., Melekhov, I., Kalia, S., Kannala, J.: Camera relocalization by computing pairwise relative poses using convolutional neural network. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 929–938 (2017)
11. Melekhov, I., Ylioinas, J., Kannala, J., Rahtu, E.: Image-based localization using hourglass networks. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 879–886 (2017)
12. Naseer, T., Burgard, W.: 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), pp. 1525–1530. IEEE (2017)
13. Paszke, A., et al.: Pytorch: an imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems, vol. 32, pp. 8024–8035. Curran Associates, Inc. (2019). http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf
14. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 652–660 (2017)
15. Sarlin, P.E., Cadena, C., Siegwart, R., Dymczyk, M.: From coarse to fine: robust hierarchical localization at large scale. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12716–12725 (2019)
16. Shavit, Y., Ferens, R.: Do we really need scene-specific pose encoders? In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 3186–3192. IEEE (2021)
17. Shavit, Y., Ferens, R., Keller, Y.: Learning multi-scene absolute pose regression with transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 2733–2742 (2021)
18. Su, W., et al.: Vl-bert: pre-training of generic visual-linguistic representations. arXiv preprint arXiv:1908.08530 (2019)
19. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: International Conference on Machine Learning, pp. 10347–10357. PMLR (2021)
20. Walch, F., Hazirbas, C., Leal-Taixe, L., Sattler, T., Hilsenbeck, S., Cremers, D.: Image-based localization using lstms for structured feature correlation. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 627–637 (2017)
21. Wang, B., Chen, C., Lu, C.X., Zhao, P., Trigoni, N., Markham, A.: Atloc: attention guided camera localization. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 10393–10401 (2020)
22. Wu, J., Ma, L., Hu, X.: Delving deeper into convolutional neural networks for camera relocalization. In: 2017 IEEE International Conference on Robotics and Automation (ICRA), pp. 5644–5651. IEEE (2017)
23. Xu, M., et al.: A critical analysis of image-based camera pose estimation techniques. arXiv preprint arXiv:2201.05816 (2022)
24. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable detr: Deformable transformers for end-to-end object detection. arXiv preprint arXiv:2010.04159 (2020)