

# OFVL-MS: Once for Visual Localization across Multiple Indoor Scenes

Tao Xie<sup>1</sup>, Kun Dai<sup>1</sup>, Siyi Lu<sup>1,2</sup>, Ke Wang<sup>1,\*</sup>, Zhiqiang Jiang<sup>1</sup>, Jinghan Gao<sup>1</sup>,  
 Dedong Liu<sup>1</sup>, Jie Xu<sup>1</sup>, Lijun Zhao<sup>1,3,\*</sup>, Ruifeng Li<sup>1,\*</sup>

<sup>1</sup>Harbin Institute of Technology <sup>2</sup>China Coal Science and Technology Intelligent Storage Technology Co., Ltd.

<sup>3</sup>Harbin Institute of Technology, Zhengzhou Research Institute

{xietao1997, wangke, jeff\_xu, zhaolj, lrf100}@hit.edu.cn

{20s108237, 19s108222, 20s108237, 22s108222, 22s108237}@stu.hit.edu.cn

## Abstract

In this work, we seek to predict camera poses across scenes with a multi-task learning manner, where we view the localization of each scene as a new task. We propose OFVL-MS, a unified framework that dispenses with the traditional practice of training a model for each individual scene and relieves gradient conflict induced by optimizing multiple scenes collectively, enabling efficient storage yet precise visual localization for all scenes. Technically, in the forward pass of OFVL-MS, we design a layer-adaptive sharing policy with a learnable score for each layer to automatically determine whether the layer is shared or not. Such sharing policy empowers us to acquire task-shared parameters for a reduction of storage cost and task-specific parameters for learning scene-related features to alleviate gradient conflict. In the backward pass of OFVL-MS, we introduce a gradient normalization algorithm that homogenizes the gradient magnitude of the task-shared parameters so that all tasks converge at the same pace. Furthermore, a sparse penalty loss is applied on the learnable scores to facilitate parameter sharing for all tasks without performance degradation. We conduct comprehensive experiments on multiple benchmarks and our new released indoor dataset LIVL, showing that OFVL-MS families significantly outperform the state-of-the-arts with fewer parameters. We also verify that OFVL-MS can generalize to a new scene with much few parameters while gaining superior localization performance. The dataset and evaluation code is available at <https://github.com/mooncake199809/UFVL-Net>.

## 1. Introduction

Visual localization, a challenging task that aims to forecast 6-DOF camera pose on a provided RGB image, is an integral part of several computer vision tasks, such as simul-

taneous localization and mapping [51, 32, 5] and structure-from-motion [11, 31].

Typically, classical structure-based visual localization frameworks [36, 34, 59, 58] construct 2D keypoints and 3D scene coordinates associations by matching local descriptors, and afterwards use a RANSAC-based PnP algorithm [15, 25] to retrieve camera pose. Recently, with the advancements of deep learning [57, 56, 41, 49, 48], scene coordinate regression (SCoRe) based methods [26, 13, 61, 53, 16, 10], which trains a convolutional neural network (CNN) to regress the 3D scene coordinate corresponding to each pixel in the input image and calculates camera pose with PnP algorithm [25], establish state-of-the-art localization performance in small static scenes. Compared with structure-based methods, these methods require no database of images or local descriptors and can benefit from high-precision sensors. While SCoRe based methods achieve impressive results, they come with some drawbacks. Scene coordinate regression is scene-specific and required to be trained for new scenes, resulting in a linear increase in total model size with the number of scenes. After witnessing the success of SCoRe-based methods, a naive problem arise: could a single SCoRe-based model predict 3D coordinates for multiple scenes concurrently and generalize to a new scene? Solving this problem is a key step towards truly SCoRe-based model deployment on autonomous robots.

A naive solution to this problem is that using a shared backbone to extract features from multiple scenes and then leveraging different regression heads to regress scene coordinates for each scene. Nevertheless, jointly optimizing cross-scene localization with a fully shared backbone exists an insurmountable obstacle, i.e., gradient conflict induced by competition among different tasks for shared parameters, resulting in inferior performance compared with learning tasks separately [27, 7, 14, 17]. Towards this end, we propose OFVL-MS, a unified SCoRe-based framework that optimizes visual localization of multiple scenes collectively. OFVL-MS is a multi-task learning (MTL) [12, 23, 29, 8,

\*Corresponding author.

[57, 52, 33, 47] framework where localization of each scene is treated as an individual task. OFVL-MS offers benefits in terms of model complexity and learning efficiency since substantial parameters of the network are shared among multiple scenes, which renders the model more pragmatic to be deployed on robotics. Technically, OFVL-MS eliminates gradient conflict from forward and backward pass.

In the forward pass, we design a layer-adaptive sharing policy to automatically determine whether each active layer of the backbone is shared or not, from which we derive task-shared parameters for efficient storage and task-specific parameters for mitigating gradient conflict. The central idea of the layer-adaptive sharing policy is to transform the layer selection of the backbone into a learnable problem, so that deciding which layers of the backbone to be shared or not can be done during training by solving a joint optimization problem. In the backward pass, inspired by gradient homogenization algorithms in classical multi-task learning [21, 28], we introduce a gradient normalization algorithm that homogenizes the gradient magnitude of the task-shared parameters across scenes to ensure all tasks converge at a similar but optimal pace, further relieving gradient conflict. We also apply a penalty loss on the active layers to prompt all tasks to share as many parameters as possible while improving the performance of some tasks that benefit from the shared parameters, as illustrated in Sec. 4.4 and Sec. 4.7. Experiments show that OFVL-MS achieves excellent localization performance on several benchmarks, including 7-Scenes dataset[39], 12-Scenes datasets [45] and our **released large indoor dataset LIVL** in terms of median positional and rotational errors, etc. We also demonstrate that OFVL-MS can generalize to a new scene with much few parameters while maintaining exceptional performance.

To summarize, the contributions of this work are as follows: (1) We propose OFVL-MS, a unified visual localization framework that optimizes localization tasks of different scenes collectively in a multi-task learning manner. (2) We propose a layer-adaptive sharing policy for OFVL-MS to automatically determine, rather than manually, whether each active layer of backbone is shared or not. A penalty loss is also applied to promote layer sharing across scenes. (3) We introduce a gradient normalization algorithm to homogenize gradient magnitudes of the task-shared parameters, enabling all tasks to converge at same pace. (4) We publish a **new large indoor dataset LIVL** that provides a new test benchmark for visual localization. (5) We demonstrate that OFVL-MS can generalize to a new scene with much fewer parameters while retaining superior localization performance.

## 2. Related Work

**Structured-based Visual Localization.** The structure-based methodologies [36, 34, 59, 58] utilize local descrip-

tors to establish 2D pixel positions and 3D scene coordinate matches for a given query image, afterwards using a PnP algorithm to recover camera pose. However, as opposed to directly matching within an exhaustive 3D map as in [36], current state-of-the-art methods [34, 59, 58] employ image retrieval [2] to narrow down the searching space and utilize advanced feature matching techniques such as Patch2pix [62], SuperGlue [35], LoFTR [42], MatchFormer [50], OAMatcher [9], and DeepMatcher [54] to generate precise 2D-2D correspondences, which are subsequently elevated to 2D-3D matches. The structured-based methods demonstrate state-of-the-art performance in large-scale scenes thanks to expeditious image retrieval techniques and feature matching algorithms, while they are limited in small-scale static scenes such as indoor scenes [26, 20]. Moreover, in lifelong localization scenarios, the size of the image and feature database increases over time due to the continuous addition of new data. As a result, the memory requirements for on-device localization in VR/AR systems may exceed the available limits.

**Learning-based Visual Localization.** Current learning-based visual localization approaches can be classified into absolute pose regression (APR) [24, 55, 22], relative pose regression (RPR) [1, 11, 44], and scene coordinate regression (SCoRe) [26, 13, 61, 53, 16]. The APR methods directly forecast the camera pose via a provided RGB image in an end-to-end way. However, such methods can not realize accurate visual localization as they are essentially analogous to approximate pose estimation via image retrieval [46]. The RPR methods utilize a neural network to identify the relative pose among the requested image and the most identical image retrieved from the database, which, however, is time-consuming and restricts their practical application. The SCoRe approaches directly forecast the 3D scene coordinates, walked by the RANSAC-based PnP algorithm. The SCoRe approaches directly forecast the scene coordinates, succeeded by the PnP algorithm to compute camera pose. While these methods can be optimized end-to-end and achieve impressive results, they suffer from some drawbacks. Pose regression and scene coordinate regression are both scene-specific and must be retrained for new scenes, culminating in a linear increase in total model size with the number of scenes.

**Gradient Homogenization over Multi-task Learning (MTL).** During the training process of multi-task learning (MTL), the gradient magnitudes and directions of different tasks interact complicatedly together via backpropagation, a phenomenon known as task interference. Previous methods [37, 28, 6, 21, 30, 40] simplify the matter to two categories of gradient discrepancies (i.e., magnitudes and directions of task gradients) and suggest various techniques to reconcile this difference. For gradient magnitudes, Sener et al. [37] characterize multi-task learning as multi-objective

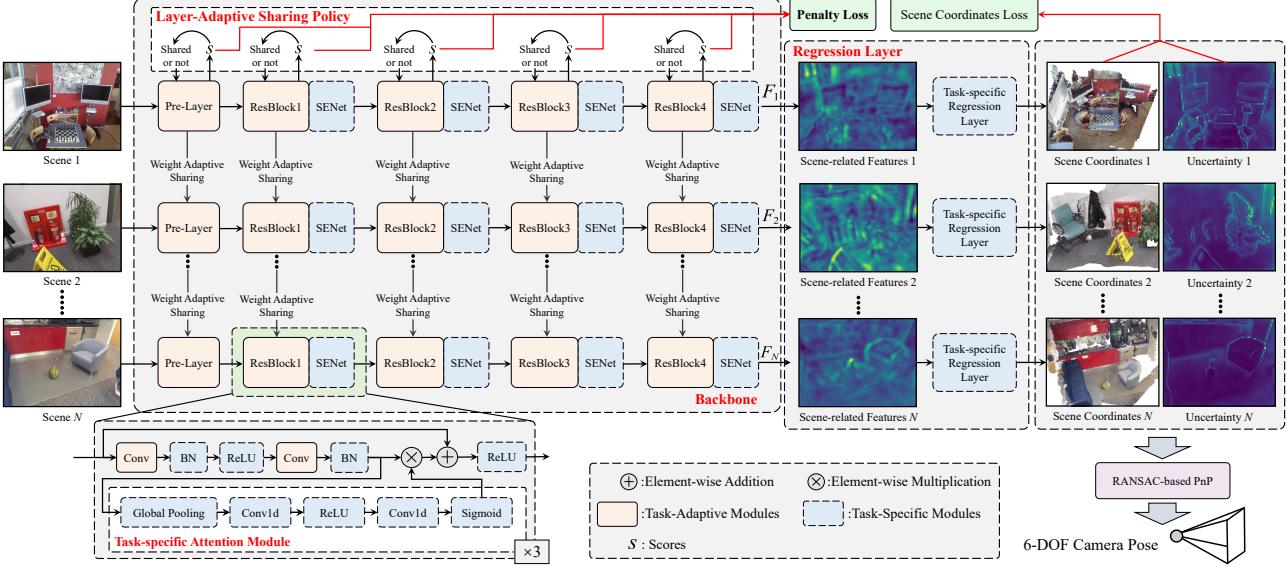


Figure 1. **Overall of OFVL-MS (using ResNet34 [18] as backbone).** OFVL-MS jointly optimizes visual localization across scenes and consists of two components, that is, backbone and regression layer. The layer-adaptive sharing policy and task-specific attention module are utilized to generate more scene-related features, which are fed into regression layers to predict scene coordinates with uncertainty. Besides, the penalty loss is proposed to facilitate OFVL-MS to share parameters as many as possible, realizing efficient storage deployment.

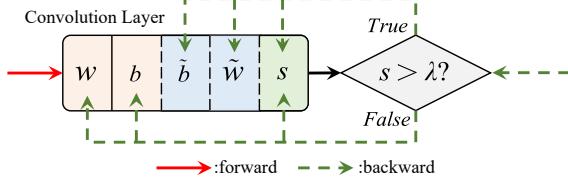


Figure 2. **Layer-adaptive Sharing Policy.** The scores  $s$  are utilized to determine which parameters ( $(w, b, s)$  or  $(\tilde{w}, \tilde{b}, s)$ ) to be optimized in current iteration.

optimization and provide an upper bound for the multi-objective loss. Javaloy et al. [21] homogenize the gradient magnitudes through normalizing and scaling, ensuring training convergence. For gradient direction, Sinha et al. [40] and Maninis et al. [30] propose to enable task gradients statistically to be indistinguishable through adversarial training.

### 3. Method

Given a RGB image, the task of visual localization seeks to estimate the rigid transformation  $T \in SE(3)$  from camera coordinate system to world coordinate system. Such transformation is composed of a 3D rotation matrix  $R \in SO(3)$  and a translation vector  $t \in \mathbb{R}^3$ .

#### 3.1. Overall

We propose OFVL-MS, a unified framework that jointly optimizes localization tasks of different scenes in a multi-task learning manner, where we view the visual localiza-

tion of each scene as a new task. OFVL-MS is a two-stage pipeline with scene coordinates prediction followed by a RANSAC-based PnP algorithm to calculate the camera pose  $T$ . Specifically, OFVL-MS takes  $N$  RGB images  $I_n \in \mathbb{R}^{3 \times H \times W}$ ,  $n \in \{1, 2, \dots, N\}$  from different scenes as input and predicts dense 3D scene coordinates  $\hat{D}_n = \{\hat{d}_{n,i} = (\hat{x}_{n,i}, \hat{y}_{n,i}, \hat{z}_{n,i}) | i = 1, 2, 3, \dots, Q\}$  with 1D uncertainty  $\hat{U}_n = \{\hat{u}_{n,i} | i = 1, 2, 3, \dots, Q\}$ , where  $Q$  is the predicted 3D scene coordinate numbers. Thus, we derive  $Q$  correspondences between 2D pixel coordinates and 3D scene coordinates. Finally, OFVL-MS utilizes RANSAC-based PnP algorithm to calculate 6-DOF camera pose  $T_n = [R_n | t_n]$ . In this work, we focus on designing and optimising OFVL-MS, which encourages all tasks to share as many parameters as possible for efficient storage deployment while maintaining superior performance for all tasks.

#### 3.2. Design OFVL-MS

As shown in Fig. 1, OFVL-MS is characterized by two components: backbone and regression layers.

**Backbone.** The backbone first utilizes a pre-layer with stride of 2 to map the input image to a higher dimension and lower resolution, and then leverages four ResBlocks [18] with stride of  $(1, 1, 2, 2)$  and several attention modules to extract features. The backbone concludes a set of task-shared parameters  $\phi^{sh}$  for  $N$  tasks and task-specific parameters  $\phi_n^{sp}$  for task  $n$  to transform each input  $I_n$  into an intermediate representation  $F_n = f(I_n; \phi^{sh}, \phi_n^{sp}) \in \mathbb{R}^{C_o \times H_o \times W_o}$ , where  $C_o$  is the dimension of  $F_n$ ,  $H_0 = H/8$ ,

$$W_0 = W/8.$$

**Regression Layer.** Additionally, each task  $n$  has a regression layer  $h$ , with exclusive parameters  $\theta_n$ , which takes  $F_n$  as input and predicts 3D scene coordinate  $\hat{D}_n$  as well as 1D uncertainty  $\hat{U}_n$  for task  $n$ .

In this work, instead of altering the architecture of the network or adding a fixed set of parameters, we seek a framework that enables all tasks to share as many parameters as feasible while retaining excellent performance, i.e., proposed task-adaptive sharing policy and gradient balance algorithm. We assume the layers with learnable parameters in the backbone except for the attention modules to be active layers, such as convolution and normalization layer, while other layers, such as ReLU layer and Sigmoid layer, are considered as inactive layers.

**Layer-adaptive Sharing Policy.** Theoretically, when manually determining whether  $K$  active layers are shared or not, a combinatorial search over  $2^K$  possible networks is required. Thus, in lieu of hand-crafted weight or layer sharing schemes, inspired by TAPS [47], we relax the combinatorial issue into a learnable one and introduce a layer-adaptive sharing policy that automatically determines whether each layer of the active layers is shared or not for diverse scenes. Using a single weight and bias for each active layer, however, does not enable different tasks to share or monopolize the parameters dynamically at various iterations during training, hence limiting the adaptivity of OFVL-MS for the scenes.

To tackle this issue, as shown in Fig. 2, taking a convolution layer as example, we cast the initial weight  $w \in \mathbb{R}^{C_{out} \times C_{in} \times k \times k}$  of the convolution kernel as task-shared parameters and define two additional parameters: task-specific weight  $\tilde{w} \in \mathbb{R}^{C_{out} \times C_{in} \times k \times k}$ , and a learnable score  $s \in \mathbb{R}^1$ , where  $C_{out}$ ,  $C_{in}$  and  $k$  mean output channels, input channels, and kernel size, respectively. In forward pass, we define an indicator function for the score to judge whether the parameters of convolution layer are shared or not in current iteration, formulated as:

$$\Theta(s) = \begin{cases} 0 & \text{if } s \geq \lambda \\ 1 & \text{otherwise,} \end{cases} \quad (1)$$

where  $\lambda$  is a preset threshold. The task-adaptive weight  $\bar{w}$  used for current iteration is formulated as:

$$\bar{w} = \Theta(s)w + (1 - \Theta(s))\tilde{w}. \quad (2)$$

If the score  $s$  is larger than the preset threshold  $\lambda$ , the task-specific parameters  $\tilde{w}$  will be activated and optimized, and vice versa. We apply above procedure on all active layers to enable different tasks to share or monopolize the parameters dynamically at various iterations. Besides, concluding additional parameters  $\tilde{w}$  into each layer does not

result in a large increase in memory cost since only the selected parameters  $\bar{w}$  and  $s$  are optimized at each iteration and all other parameters are kept offline.

Compared with TAPS, our proposed sharing policy delivers following merits: (1) we introduce individual task-shared weight  $w$  and task-specific weight  $\tilde{w}$  for each active layer rather than a coupled weight in TAPS, enabling the memory footprint to be agnostic to the number of tasks; (2) once the training for multi-task is done, the new added task can share task-shared parameters or task-specific parameters with any task in our setting, allowing for more flexible parameter sharing and real multi-task learning.

Notably, we set the learnable score  $s$  to be task-shared so that ensuring the parameters of all scenes can be integrated into a collective model. Moreover, we calculate the summation of the absolute values of all scores as penalty loss to enable all tasks to share parameters as many as possible, achieving efficient storage deployment. Since the indicator function  $\Theta(\cdot)$  is not differentiable, we need to modify its gradient during backward pass, which will be presented in Appendix 2.1. Notably, as illustrated in Sec. 4.5, learning task-specific batch normalization can significantly improve the localization performance while adding small parameters, so we set the parameters of normalization layers in active layers as task-specific.

**Task-specific Attention Module.** We further embed an attention module into the backbone, empowering OFVL-MS to learn more scene-related features. The attention module learns a soft attention mask to the features, that can automatically determine the importance of features for each task along the channel dimension, enabling self-supervised learning of more scene-related features. In this work, we adopt SENet [19] as attention module and integrate it into the BasicBlock of each ResBlock. Each task  $n$  has task-specific attention modules with exclusive parameters.

### 3.3. Optimize OFVL-MS

Since each task has its own dataset domain, we need to utilize multiple GPUs to optimize these tasks. For the sake of description, we assume that a single GPU is used to train each scene.

**Loss.** The goal of OFVL-MS is to ensure precise visual localization for all scenes while enabling different tasks to share as many parameters as possible. Therefore, we cast the training process of OFVL-MS as a joint optimization problem for predicted scene coordinates and scores in Eq. (1). For the  $n$ -th scene, the loss  $L_n$  involves two terms: the scene coordinates loss  $L_n^{sc}$  and the penalty loss  $L_n^{pe}$ .

$$L_n = L_n^{sc} + \beta L_n^{pe}, \quad (3)$$

where  $\beta$  denotes weight coefficient used to reconcile  $L_n^{sc}$  and  $L_n^{pe}$ .

*Scene coordinates loss.* We employ the loss function proposed by KFNet [61] to maximize the logarithmic likelihood for the probability density function of the predicted scene coordinates. Specifically, the loss function of the  $n$ -th scene is formatted as:

$$L_n^{sc} = \frac{1}{Q} \sum_{i=1}^Q (3\log \hat{u}_{n,i} + \frac{\|d_{n,i} - \hat{d}_{n,i}\|_2^2}{2\hat{u}_{n,i}^2}), \quad (4)$$

where  $Q$  equals to  $H/8 \times W/8$ ;  $\hat{u}_{n,i}$  is the  $i$ -th predicted uncertainty;  $d_{n,i}$  is the  $i$ -th ground truth scene coordinate;  $\hat{d}_{n,i}$  is the  $i$ -th predicted scene coordinate.

*Penalty loss on the learnable scores.* The penalty loss  $L_n^{pe}$  motivates all tasks to share the parameters of active layers as many as possible. Such loss is denoted by calculating the summation of the absolute values of scores  $s_n$  for the  $n$ -th scene:

$$L_n^{pe} = \frac{1}{\|S_n\|} \sum_{s_n \in S_n} |s_n|, \quad (5)$$

where  $S_n$  means the collection of the scores;  $\|S_n\|$  denotes the number of scores. It is worth noting that the scores  $s_n$  of all scenes are identical since they are set as task-shared.

**Backward Pass and Gradient Normalization Algorithm.** For convenient portrayal, we denote the task-shared and task-specific parameters of OFVL-MS for  $n$ -th scene as  $\chi_n^{sh} = \{\phi^{sh}\}$  and  $\chi_n^{sp} = \{\phi_n^{sp}, \theta_n\}$ .

For task-specific parameters, we define the gradients of  $\chi_n^{sp}$  for  $n$ -th scene at  $i$ -th iteration as:  $G_{n,i}^{sp} = \nabla_{\chi_{n,i}^{sp}} L_{n,i}$ , where  $L_{n,i}$  means the loss function for the  $n$ -th scene at the  $i$ -th iteration. Subsequently, the task-specific parameters on each GPU will be optimized based on the calculated  $G_{n,i}^{sp}$ . Noting that when optimizing a scene with multiple GPUs, the gradients  $G_{n,i}^{sp}$  on the GPUs would be averaged and then the parameters are updated accordingly. For task-shared parameters, the gradients of  $\chi_n^{sh}$  for  $n$ -th scene at  $i$ -th iteration is also formulated as:  $G_{n,i}^{sh} = \nabla_{\chi_{n,i}^{sh}} L_{n,i}$ .

A straightforward scheme for optimizing the task-shared parameters involves averaging the gradients  $G_{n,i}^{sh}$  across all GPUs and then updating the corresponding weights. While this method streamlines the optimization problem, it may also trigger gradient conflict among tasks, lowering overall performance due to an unequal competition among tasks for the shared parameters, i.e., gradient magnitude disparities. Moreover, OFVL-MS is designed for jointly optimizing multiple indoor scenes, where the varied scene domains will further intensify the gradient conflict. Inspired by [28, 40, 21, 37], we utilize a gradient normalization algorithm to homogenize the gradient magnitude of the task-shared parameters for all scenes, allowing all tasks to converge at same pace and alleviating the gradient conflict. Specifically, OFVL-MS first places gradient norms of task-shared parameters on a common scale  $D$ . Considering the

magnitudes and the change rate of gradient reflect whether the optimization direction in current iteration is dependable or not, we define  $D$  as the linear combination of task-wise gradient magnitudes:

$$D = \sum_{n=1}^N W_{n,i} \|G_{n,i}^{sh}\|_2, \quad (6)$$

where the weight  $W_{n,i}$  is denoted as the relative convergence of each task:

$$W_{n,i} = \frac{\|G_{n,i}^{sh}\|_2 / \|G_{n,i-1}^{sh}\|_2}{\sum_{j=1}^N \|G_{j,i}^{sh}\|_2 / \|G_{j,i-1}^{sh}\|_2}. \quad (7)$$

Then, given the common scale  $D$ , OFVL-MS generates the optimized gradients  $\hat{G}_{n,i}^{sh}$ :

$$\hat{G}_{n,i}^{sh} = D \frac{G_{n,i}^{sh}}{\|G_{n,i}^{sh}\|_2}. \quad (8)$$

Ultimately, we average the gradients  $\hat{G}_{n,i}^{sh}$  on all GPUs to derive  $\hat{G}_i^{sh}$ , ensuring the gradients of the task-shared parameters for all scene are equivalent. The  $\hat{G}_i^{sh}$  is formulated as:

$$\hat{G}_i^{sh} = \frac{1}{N} \sum_{n=1}^N \hat{G}_{n,i}^{sh}. \quad (9)$$

### 3.4. Pose Estimation

We design the regression layer as a fully convolutional structure to predict dense 3D scene coordinates as well as 1D uncertainty, where the uncertainty is utilized to measure the prediction effect by quantifying the noise induced from both data and model. Based on the predicted 2D pixel coordinates-3D scene coordinates correspondences, we apply the RANSAC-based PnP algorithm to minimize reprojection errors and finally derive camera pose  $T$ .

## 4. Experiments

### 4.1. Datasets

**7-Scenes** [39] dataset records 41k RGB-D images and corresponding camera poses of seven different indoor environments using a handheld Kinect camera. **12-Scenes** [45] dataset, whose recorded environment is larger than that of 7-Scenes, records RGB-D images in twelve indoor environments with an iPad color camera.

### 4.2. Experimental Settings

**Implementation Details.** We employ the Adamw solver for optimization with a weight decay of 0.05. The initial learning rate is set to  $1.4 \times 10^{-3}$  for 7-Scenes while  $2.4 \times 10^{-3}$  for 12-Scenes with cosine annealing. Considering the number of images for each scene is distinct, we

Methods	Metrics	Chess	Fire	Heads	Office	Pumpkin	Redkitchen	Stairs	Average
AS [36]	Med. Err. Acc.	0.03, 0.87 —	0.02, 1.01 —	0.01, 0.82 —	0.04, 1.15 —	0.07, 1.69 —	0.05, 1.72 —	0.04, 1.01 —	0.03, 1.18 68.7
InLoc [43]	Med. Err. Acc.	0.03, 1.05 —	0.03, 1.07 —	0.02, 1.16 —	0.03, 1.05 —	0.05, 1.55 —	0.04, 1.31 —	0.09, 2.47 —	0.04, 1.38 66.3
HLoc [34]	Med. Err. Acc.	0.02, 0.85 —	0.02, 0.94 —	0.01, 0.75 —	0.03, 0.92 —	0.05, 1.30 —	0.04, 1.40 —	0.05, 1.47 —	0.03, 1.09 73.1
MS-Transformer [38]	Med. Err. Acc.	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 —	0.18, 7.27 —
DSAC* [3]	Med. Err. Acc.	0.02, 1.10 —	0.02, 1.24 —	0.01, 1.82 —	0.03, 1.15 —	0.04, 1.34 —	0.04, 1.68 —	0.03, 1.16 —	0.02, 1.35 85.2
SCoordNet [61]	Med. Err. Acc.	0.019, 0.63 —	0.023, 0.91 —	0.018, 1.26 —	0.026, 0.73 —	0.039, 1.09 —	0.039, 1.18 —	0.037, 1.06 —	0.029, 0.98 —
HSCNet [26]	Med. Err. Acc.	0.02, 0.7 <b>97.5</b> 96.7	0.02, 0.9 <b>100.0</b> 86.5	0.01, 0.9 —	0.03, 0.8 —	0.04, 1.0 59.9	0.04, 1.2 65.5	0.03, 0.8 <b>87.5</b> 84.8	0.03, 0.9 84.8
FDANet [53]	Med. Err. Acc.	0.018, 0.64 95.70	0.018, 0.73 96.10	0.013, 1.07 99.20	0.026, 0.75 88.08	0.036, 0.91 65.65	0.034, 1.03 78.32	0.041, 1.14 62.80	0.026, 0.89 83.69
VS-Net [20]	Med. Err. Acc.	<b>0.015, 0.5</b> —	0.019, 0.8 —	0.012, 0.7 —	<b>0.021, 0.6</b> —	0.037, 1.0 —	0.036, 1.1 —	0.028, 0.8 —	0.024, 0.8 —
OFVL-MS18	Med. Err. Acc.	0.021, 0.67 96.20	0.018, 0.67 97.55	0.010, 0.56 98.90	0.030, 0.83 81.73	0.033, 0.96 67.15	0.035, 1.02 75.06	0.031, 0.89 79.80	0.025, 0.80 85.19
OFVL-MS34	Med. Err. Acc.	0.019, 0.63 97.40	0.017, 0.65 96.60	<b>0.008, 0.53</b> <b>100.0</b> 85.58	0.027, 0.74 67.50	0.031, 0.93 77.14	0.032, 1.01 87.40	0.027, <b>0.69</b> 87.40	0.023, 0.74 87.37
OFVL-MS50	Med. Err. Acc.	<b>0.015, 0.50</b> 97.10	<b>0.015, 0.59</b> <b>99.40</b>	<b>0.008, 0.56</b> <b>100.0</b> <b>89.53</b>	<b>0.023, 0.63</b> <b>89.53</b>	<b>0.030, 0.86</b> <b>68.80</b>	<b>0.031, 0.99</b> <b>81.48</b>	<b>0.026, 0.76</b> 84.70	0.021, <b>0.69</b> <b>88.72</b>

Table 1. The median positional error (m), rotational error ( $^\circ$ ), and 5cm- $5^\circ$  accuracy (%) of different methods on **7-Scenes dataset**.

Methods	Med. Err.	Acc.
DSAC* [4]	—	99.1
SCoordNet [61]	—	98.9
HSCNet [26]	0.011, 0.50	99.3
FDANet [53]	0.014, 0.37	99.6
OFVL-MS18	0.013, 0.48	98.7
OFVL-MS34	0.007, 0.25	99.9
OFVL-MS50	0.008, 0.30	99.5

Table 2. The median positional error (m), rotational error ( $^\circ$ ), and 5cm- $5^\circ$  accuracy (%) of different methods on **12-Scenes dataset**.

train OFVL-MS for  $200k$  iterations with batch size of 4. For layer-adaptive sharing policy, we set the threshold  $\lambda = 0.5$  in Eq. (1) to determine whether each active layer of the backbone is shared or not. Besides, we set  $\beta = 0.25$  in Eq. (2) to reconcile scene coordinates loss and penalty loss. More implementation details can be found in Appendix 2.

**Evaluation Metrics.** Following previous works [53, 61, 26], we evaluate our method using the following metrics: (i) the median positional and rotational errors of the predicted pose; (ii) the percentage of images with positional and rotational errors less than 5cm and  $5^\circ$ .

#### 4.3. Comparison with State-of-the-art Methods

We design three versions OFVL-MS18, OFVL-MS34 and OFVL-MS50 of our method by using ResNet18, ResNet34, ResNet50 [18] as backbone respectively, and then compare OFVL-MS families with other state-of-the-arts on 7-Scenes and 12-Scenes datasets, with the results

reported in Table 1 and Table 2.

**Localization on 7-Scenes.** We compare OFVL-MS with representative structure-based methods (AS [36], InLoc [43], HLoc [34]), APR methods (MS-Transformer [38]), and SCoRe-based methods (DSAC\* [4], SCoordNet [61], HSCNet [26], FDANet [53], and VSNet [20]). As shown in Table 1, OFVL-MS surpasses existing methods by non-trivial margins in terms of all evaluation metrics. Specifically, OFVL-MS18/34 outperforms the structure-based method HLoc by 12.09%/14.27% in terms of 5cm- $5^\circ$  accuracy. Besides, compared with SCoRe-based methods HSCNet and FDANet, OFVL-MS18/34 realizes outstanding performance with the improvement of 0.4%/2.57% and 1.51%/3.68%. Compared with the cutting-edge method VS-Net, OFVL-MS18/34 also achieve higher performance. Moreover, OFVL-MS50 yields 0.021m median position error,  $0.69^\circ$  median rotational error and 88.72% 5cm- $5^\circ$  accuracy, establishing a new state-of-the-art for 7-Scenes dataset. Fig. 3 shows the cumulative pose errors distribution of different approaches on 7-Scenes dataset, which further demonstrates the superiority of OFVL-MS families in visual localization.

**Localization on 12-Scenes.** As illustrated in Table 2, we compare OFVL-MS families with state-of-the-arts on 12-Scenes dataset. It can be observed that all methods achieve excellent results since the training trajectories closely resemble the test trajectories. Despite this, OFVL-MS families exhibit exceptional performances, in which OFVL-MS34 realizes the most superior performance with the positional errors of 7mm and localization accuracy of

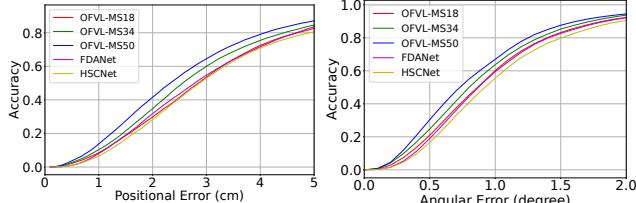


Figure 3. The cumulative pose errors distribution of different methods on 7-Scenes dataset. We aggregate the poses across all scenes and calculate the percentage of poses with the error threshold increasing.

Methods	Total Params (M)	Med. Err.	Acc.
7-Scenes			
DSAC++ [3]	182.384	0.036, 1.10	74.4
HSCNet [26]	288.751	0.030, 0.90	84.8
FDANet [53]	168.758	0.026, 0.89	83.69
OFVL-MS18	<b>48.803</b>	0.025, 0.80	85.19
OFVL-MS34	64.403	0.023, 0.74	87.37
OFVL-MS50	53.015	<b>0.021, 0.69</b>	<b>88.72</b>
12-Scenes			
HSCNet [26]	495.002	0.011, 0.50	99.3
FDANet [53]	289.299	0.014, 0.37	99.6
OFVL-MS18	<b>75.453</b>	0.013, 0.48	98.7
OFVL-MS34	158.693	<b>0.007, 0.25</b>	<b>99.9</b>
OFVL-MS50	126.694	0.008, 0.30	99.5

Table 3. The model size of different methods. OFVL-MS families achieve the best localization accuracy with much less parameters.

99.9%.

**Model Size Comparison.** We compare the storage space occupied by different methods to demonstrate the efficient storage deployment of OFVL-MS families. Previous works typically train a separate model for each scene, resulting in a linear increase in model size with the number of scenes. However, OFVL-MS deposits multiple models with a majority of shared parameters into a single one, realizing efficient storage. As shown in Table 3, OFVL-MS families reduce the model parameters significantly compared with other state-of-the-arts. For 7-Scenes dataset, the parameters size of OFVL-MS50 is only 1/5 of that of HSCNet, but the localization accuracy is improved by 3.92%. For 12-Scenes dataset, OFVL-MS34 achieves the best performance with much fewer parameters (only 1/3 of HSCNet).

#### 4.4. Joint Training vs Separate Training

To further demonstrate the efficiency of jointly optimizing localization tasks across scenes, we train a separate model for each scene. We choose OFVL-MS34 as the benchmark for validation. As shown in Table 4, OFVL-MS34 reduces total model size from 177.779M to 64.403M by sharing parameters for all scenes. Besides, it is astonishing to find OFVL-MS34 achieves competitive performance through joint training, indicating that closely-related tasks have mutual benefits.

Methods	Total Params (M)	Med. Err.	Acc.
Separate Learning	177.779	<b>0.023, 0.74</b>	86.50
Joint Learning	<b>64.403</b>	<b>0.023, 0.74</b>	<b>87.37</b>

Table 4. The comparison between joint and separate training of OFVL-MS34 on 7-Scenes dataset.

Methods	Total Params (M)	Med. Err.	Acc.
EXP1	<b>50.243</b>	0.027, 0.82	79.94
EXP2	64.391	0.024, 0.76	86.10
EXP3 (Ours)	64.403	<b>0.023, 0.74</b>	<b>87.37</b>

Table 5. The comparison between different parameters sharing strategies on 7-Scenes dataset.

Methods	Increased Params (M)	Med. Err.	Acc.
EXP1: 12-Scenes to 7-Scenes			
HSCNet [26]	$41.250 \times 7$	0.030, 0.90	84.80
FDANet [53]	$24.108 \times 7$	0.026, 0.89	83.69
OFVL-MS18 <sup>†</sup>	<b>5.476 × 7</b>	0.029, 0.93	77.59
OFVL-MS34 <sup>†</sup>	$12.117 \times 7$	0.023, 0.75	85.73
OFVL-MS50 <sup>†</sup>	$9.881 \times 7$	<b>0.021, 0.71</b>	<b>86.75</b>
EXP2: 7-Scenes to 12-Scenes			
HSCNet [26]	$41.250 \times 12$	0.011, 0.50	99.3
FDANet [53]	$24.108 \times 12$	0.014, 0.37	<b>99.6</b>
OFVL-MS18 <sup>†</sup>	<b>5.597 × 12</b>	0.009, 0.38	96.7
OFVL-MS34 <sup>†</sup>	$6.501 \times 12$	0.009, 0.31	97.3
OFVL-MS50 <sup>†</sup>	$5.835 \times 12$	<b>0.008, 0.29</b>	98.3

Table 6. Experiments of generalizing to new scenes. <sup>†</sup> indicates using the models trained on 12-Scenes/7-Scenes to conduct the generalization experiments on 7-Scenes/12-Scenes. Increased Params (M) means the extra parameters size that each method requires when generalizing to new scenes.

#### 4.5. Diverse Parameters Sharing Strategies

To verify the effectiveness of the proposed layer-adaptive sharing policy, we apply three different parameter sharing strategies on OFVL-MS34 for 7-Scenes dataset. EXP1: All parameters of active layers are set as task-shared. EXP2: All parameters of active layers (both convolutional and batch normalization layers) are determined whether to be shared by scores. EXP3: All parameters of active layers (only convolutional layers) are determined whether to be shared by scores, and the batch normalization layers are set as task-specific. As shown in Table 5, compared to setting all parameters as task-shared, OFVL-MS34 significantly improves localization performance from 79.94 to 87.37 in terms of 5cm-5° accuracy at the expense of a small increase in model parameters, indicating that using additional task-specific parameters to learn scene-related features is critical to resolve gradient conflict. Besides, the performance of OFVL-MS is further enhanced with BN layers set as task-specific.

#### 4.6. Generalize to New Scenes

In this part, we conduct two experiments to demonstrate that OFVL-MS can generalize to new scenes with much

Methods	TSAM	GNA	PL	Med. Err.	Acc.	Params-t (M)
EXP1	$\times$	✓	✓	0.026, 0.79	84.30	<b>63.297</b>
EXP2	✓	$\times$	✓	0.025, 0.77	84.16	64.403
EXP3	✓	✓	$\times$	<b>0.023, 0.74</b>	86.25	78.059
EXP4	✓	✓	✓	<b>0.023, 0.74</b>	<b>87.37</b>	64.403

Table 7. **Ablation study with various variants of OFVL-MS** on 7-Scenes dataset. TSAM: Task-specific Attention Module, GNA: Gradient Normalization Algorithm, PL: Penalty Loss. Params-t means the total parameters of OFVL-MS34 for the seven scenes.

fewer parameters and thus can scale up gracefully with the number of scenes. We utilize the model trained on 12-Scenes/7-Scenes and conduct the generalization experiments on 7-Scenes/12-Scenes. Specifically, we freeze the task-shared parameters trained on 12-Scenes/7-Scenes, and add task-specific parameters as well as an additional regression layer for each scene of 7-Scenes/12-Scenes to predict the scene coordinates.

As shown in Table 6, despite generalizing to a new scene, OFVL-MS34/50 still outperform HSCNet and FDANet by 0.93%/1.95% and 2.04%/3.06% in terms of 5cm-5° accuracy for EXP1, illustrating that OFVL-MS can avoid catastrophic forgetting and achieve genuine incremental learning. Besides, compared with 41.250/24.108 M increased parameters of HSCNet and FDANet, OFVL-MS18/34/50 only need 5.476/12.117/9.881 M parameters when generalizing to a new scene, realizing efficient storage.

For EXP2, OFVL-MS families yield the lowest localization errors. It is worth noting that the incremental models achieve more precise localization performance in most of scenes except for Floor5b, resulting in the 5cm-5° accuracy declined, which will be presented in Appendix 3. Moreover, OFVL-MS families realize efficient storage deployment with 5.597/6.501/5.835 M additional parameters compared with HSCNet and FDANet.

#### 4.7. Ablation study

To comprehensively confirm the veracity of the modules suggested in this work, various variants of OFVL-MS34 are validated using the 7-Scenes dataset. As shown in Table 7, all of the components contribute to outstanding performance. EXP1: Removing all task-specific attention modules results in a large drop in localization accuracy, demonstrating the strong ability of TSAM to generate more scene-related features, realizing efficient scene parsing. EXP2: Removing gradient normalization algorithm leads to much lower accuracy, validating that homogenizing the gradient magnitude of the task-shared parameters alleviates the gradient conflict significantly. EXP3: Removing penalty loss results in degraded localization accuracy, indicating that promoting the informative parameters sharing across scenes improves localization performance.

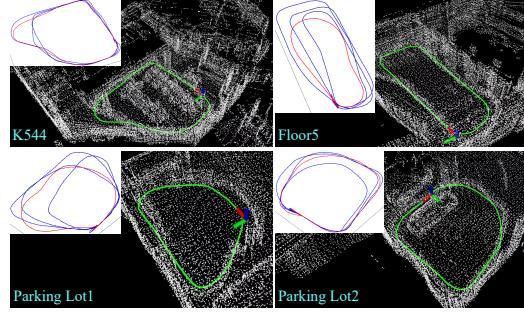


Figure 4. **LIVL dataset**. The blue lines indicate training trajectories whereas the red lines indicate test trajectories.

Methods	Metric	K544	Floor5	Parking lot1	Parking lot2	Average	Params-t (M)
SCoordNet [61]	Med. Err.	0.171, 2.12	0.208, 1.94	0.353, 2.97	0.184, 2.13	0.229, 2.29	93.086
	Acc.	9.86	20.31	11.28	20.82	15.57	
FDANet [53]	Med. Err.	0.143, 1.89	0.167, 1.59	0.291, 2.75	0.138, 1.61	0.185, 1.96	96.432
	Acc.	12.64	23.87	13.31	22.89	18.18	
OFVL-MS18	Med. Err.	0.074, 1.12	0.174, 1.84	0.274, 2.28	0.100, 1.31	0.155, 1.64	<b>37.246</b>
	Acc.	39.21	16.92	24.50	28.34	27.24	
OFVL-MS34	Med. Err.	0.071, 1.01	<b>0.147, 1.48</b>	0.278, <b>2.08</b>	<b>0.095, 1.17</b>	0.147, 1.43	73.184
	Acc.	42.53	25.54	25.72	<b>29.03</b>	30.71	
OFVL-MS50	Med. Err.	<b>0.050, 0.81</b>	0.148, <b>1.37</b>	<b>0.265, 2.48</b>	0.107, <b>1.02</b>	<b>0.142, 1.42</b>	45.678
	Acc.	<b>49.91</b>	<b>30.72</b>	<b>26.17</b>	28.34	<b>33.79</b>	

Table 8. The median positional error (m) and rotational error (°) of OFVL-MS families on **LIVL dataset**. Params-t means that the total parameters of OFVL-MS for the four scenes.

#### 4.8. Camera Localization on LIVL

Despite the existence of publicly available datasets for visual localization, there is no dataset for large-scale indoor scenes. Thus, we introduce the challenging **LIVL** dataset containing RGB-D images tagged with 6-DoF camera poses collected around four scenes. (i) **K544**: spanning about  $12 \times 9\text{m}^2$ . (ii) **Floor5**: spanning about  $12 \times 5\text{m}^2$ . (iii) **Parking lot1** spanning about  $8 \times 6\text{m}^2$ . (iv) **Parking lot2** spanning about  $8 \times 8\text{m}^2$ . Each scene contains three sequences for training and one sequence for test. A massive proportion of motion blur and sparse texture in the scene make visual localization in the four scenes challenging. We give the visualization of **LIVL** dataset in Fig. 4. The dataset was collected using a autonomous platform armed with a RealSense D435 camera and a VLP-16 laser radar. The RGB and depth images are captured at a resolution of  $640 \times 480$  pixels and aligned with point clouds using timestamp. We utilize the LiDAR-based SLAM system A-LOAM [60] to compute the ground truth pose. More details of the dataset can be found in Appendix 4.

As shown in Table 8, we can observe that OFVL-MS50 realizes the best performance with 0.142m and 1.42° median localization error. Wherein, OFVL-MS50 yields 0.05m and 0.81° localization error in K544 scene that contains discriminative texture. Moreover, Floor5 and Parking lot1 are laborious for OFVL-MS families to localize since there exists repetitive and sparse texture, and illu-

mination disturbance. Besides, we can also observe that 5cm-5° accuracy is inferior due to the large scale of LIVI dataset. Compared with typical SCoRe based methods SCoordNet [61] and FDANet [53], OFVL-MS families outperform them by non-trivial margins in terms of all evaluation metrics while necessitating much fewer total parameters, further indicating that the closely-related tasks benefit from the shared parameters and the efficacy of our OFVL-MS.

## 5. Conclusion

In this work, we introduce OFVL-MS, a unified network that achieves precise visual localization across scenes in a multi-task learning manner. OFVL-MS achieves high performance for all tasks and keeps storage efficient for model deployment through forward pass (layer-adaptive sharing policy) and backward pass (gradient normalization algorithm) of the network. Moreover, a penalty loss is proposed to motivate OFVL-MS to share parameters as many as possible while maintaining precise localization accuracy. We demonstrate that OFVL-MS can generalize to a new scene with small task-specific parameters while realizing superior localization performance. We also publish a **new large indoor dataset LIVL** to provide a new test benchmark for the community.

**Acknowledgement.** This work was supported in part by National Natural Science Foundation of China under Grant 62073101, in part by Science and Technology Innovation Venture Capital Project of Tiandi Technology Co., LTD. (2022-2-TD-QN009), in part by “Ten Thousand Million” Engineering Technology Major Special Support Action Plan of Heilongjiang Province, China (SC2021ZX02A0040), and in part by Self-Planned Task (SKLRS202301A09) of SKLRS (HIT) of China.

## References

- [1] Yehya Abouelnaga, Mai Bui, and Slobodan Ilic. Distillpose: Lightweight camera localization using auxiliary learning. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7919–7924. IEEE, 2021. [2](#)
- [2] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5297–5307, 2016. [2](#)
- [3] Eric Brachmann and Carsten Rother. Learning less is more: 6d camera localization via 3d surface regression. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4654–4662, 2018. [6](#) [7](#)
- [4] Eric Brachmann and Carsten Rother. Visual camera relocalization from rgb and rgb-d images using dsac. *IEEE transactions on pattern analysis and machine intelligence*, 44(9):5847–5865, 2021. [6](#)
- [5] Carlos Campos, Richard Elvira, Juan J Gómez Rodríguez, José MM Montiel, and Juan D Tardós. Orb-slam3: An accurate open-source library for visual, visual-inertial, and multimap slam. *IEEE Transactions on Robotics*, 37(6):1874–1890, 2021. [1](#)
- [6] Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *International conference on machine learning*, pages 794–803. PMLR, 2018. [2](#)
- [7] Zhao Chen, Jiquan Ngiam, Yanping Huang, Thang Luong, Henrik Kretzschmar, Yuning Chai, and Dragomir Anguelov. Just pick a sign: Optimizing deep multitask models with gradient sign dropout. *Advances in Neural Information Processing Systems*, 33:2039–2050, 2020. [1](#)
- [8] Sumanth Chennupati, Ganesh Sistu, Senthil Yogamani, and Samir A Rawashdeh. Multinet++: Multi-stream feature aggregation and geometric loss strategy for multi-task learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. [2](#)
- [9] Kun Dai, Tao Xie, Ke Wang, Zhiqiang Jiang, Ruifeng Li, and Lijun Zhao. Oamatcher: An overlapping areas-based network for accurate local feature matching. *arXiv preprint arXiv:2302.05846*, 2023. [2](#)
- [10] Kun Dai, Tao Xie, Ke Wang, Zhiqiang Jiang, Dedong Liu, Ruifeng Li, and Jiahe Wang. Eaainet: An element-wise attention network with global affinity information for accurate indoor visual localization. *IEEE Robotics and Automation Letters*, 8(6):3166–3173, 2023. [1](#)
- [11] Mingyu Ding, Zhe Wang, Jiankai Sun, Jianping Shi, and Ping Luo. Camnet: Coarse-to-fine retrieval for camera relocalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2871–2880, 2019. [1](#), [2](#)
- [12] Carl Doersch and Andrew Zisserman. Multi-task self-supervised visual learning. In *Proceedings of the IEEE international conference on computer vision*, pages 2051–2060, 2017. [2](#)
- [13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. [1](#), [2](#)
- [14] Chris Fifty, Ehsan Amid, Zhe Zhao, Tianhe Yu, Rohan Anil, and Chelsea Finn. Efficiently identifying task groupings for multi-task learning. *Advances in Neural Information Processing Systems*, 34:27503–27516, 2021. [1](#)
- [15] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. [1](#)
- [16] Peiyu Guan, Zhiqiang Cao, Junzhi Yu, Chao Zhou, and Min Tan. Scene coordinate regression network with global context-guided spatial feature transformation for visual relocalization. *IEEE Robotics and Automation Letters*, 6(3):5737–5744, 2021. [1](#), [2](#)

- [17] Michelle Guo, Albert Haque, De-An Huang, Serena Yeung, and Li Fei-Fei. Dynamic task prioritization for multitask learning. In *Proceedings of the European conference on computer vision (ECCV)*, pages 270–287, 2018. 1
- [18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3, 6
- [19] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 4
- [20] Zhaoyang Huang, Han Zhou, Yijin Li, Bangbang Yang, Yan Xu, Xiaowei Zhou, Hujun Bao, Guofeng Zhang, and Hongsheng Li. Vs-net: Voting with segmentation for visual localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6101–6111, 2021. 2, 6
- [21] Adrián Javaloy and Isabel Valera. Rotograd: Gradient homogenization in multitask learning. *arXiv preprint arXiv:2103.02631*, 2021. 2, 3, 5
- [22] Alex Kendall and Roberto Cipolla. Modelling uncertainty in deep learning for camera relocalization. In *2016 IEEE international conference on Robotics and Automation (ICRA)*, pages 4762–4769. IEEE, 2016. 2
- [23] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491, 2018. 2
- [24] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proceedings of the IEEE international conference on computer vision*, pages 2938–2946, 2015. 2
- [25] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Epnp: An accurate o (n) solution to the pnp problem. *International journal of computer vision*, 81(2):155–166, 2009. 1
- [26] Xiaotian Li, Shuzhe Wang, Yi Zhao, Jakob Verbeek, and Juho Kannala. Hierarchical scene coordinate classification and regression for visual localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11983–11992, 2020. 1, 2, 6, 7
- [27] Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. *Advances in Neural Information Processing Systems*, 34:18878–18890, 2021. 1
- [28] Liyang Liu, Yi Li, Zhanghui Kuang, J Xue, Yimin Chen, Wenming Yang, Qingmin Liao, and Wayne Zhang. Towards impartial multi-task learning. *ICLR*, 2021. 2, 5
- [29] Shikun Liu, Edward Johns, and Andrew J Davison. End-to-end multi-task learning with attention. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1871–1880, 2019. 2
- [30] Kevis-Kokitsi Maninis, Ilija Radosavovic, and Iasonas Kokkinos. Attentive single-tasking of multiple tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1851–1860, 2019. 2, 3
- [31] Dror Moran, Hodaya Koslowsky, Yoni Kasten, Haggai Maron, Meirav Galun, and Ronen Basri. Deep permutation equivariant structure from motion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5976–5986, 2021. 1
- [32] Raul Mur-Artal and Juan D Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgbd cameras. *IEEE transactions on robotics*, 33(5):1255–1262, 2017. 1
- [33] Noha Radwan, Abhinav Valada, and Wolfram Burgard. Vlocnet++: Deep multitask learning for semantic visual localization and odometry. *IEEE Robotics and Automation Letters*, 3(4):4407–4414, 2018. 2
- [34] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12716–12725, 2019. 1, 2, 6
- [35] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4938–4947, 2020. 2
- [36] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Efficient & effective prioritized matching for large-scale image-based localization. *IEEE transactions on pattern analysis and machine intelligence*, 39(9):1744–1756, 2016. 1, 2, 6
- [37] Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. *Advances in neural information processing systems*, 31, 2018. 2, 5
- [38] Yoli Shavit, Ron Ferens, and Yosi Keller. Learning multi-scene absolute pose regression with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2733–2742, 2021. 6
- [39] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgbd images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2930–2937, 2013. 2, 5
- [40] Ayan Sinha, Zhao Chen, Vijay Badrinarayanan, and Andrew Rabinovich. Gradient adversarial training of neural networks. *arXiv preprint arXiv:1806.08028*, 2018. 2, 3, 5
- [41] Ziying Song, Haiyue Wei, Caiyan Jia, Yongchao Xia, Xiaokun Li, and Chao Zhang. Vp-net: Voxels as points for 3d object detection. *IEEE Transactions on Geoscience and Remote Sensing*, 2023. 1
- [42] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8922–8931, 2021. 2
- [43] Hajime Taira, Masatoshi Okutomi, Torsten Sattler, Mircea Cimpoi, Marc Pollefeys, Josef Sivic, Tomas Pajdla, and Akihiko Torii. Inloc: Indoor visual localization with dense matching and view synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7199–7209, 2018. 6

- [44] Mehmet Ozgur Turkoglu, Eric Brachmann, Konrad Schindler, Gabriel J Brostow, and Aron Monszpart. Visual camera re-localization using graph neural networks and relative pose supervision. In *2021 International Conference on 3D Vision (3DV)*, pages 145–155. IEEE, 2021. [2](#)
- [45] Julien Valentin, Angela Dai, Matthias Nießner, Pushmeet Kohli, Philip Torr, Shahram Izadi, and Cem Keskin. Learning to navigate the energy landscape. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 323–332. IEEE, 2016. [2, 5](#)
- [46] Florian Walch, Caner Hazirbas, Laura Leal-Taixe, Torsten Sattler, Sebastian Hilsenbeck, and Daniel Cremers. Image-based localization using lstms for structured feature correlation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 627–637, 2017. [2](#)
- [47] Matthew Wallingford, Hao Li, Alessandro Achille, Avinash Ravichandran, Charless Fowlkes, Rahul Bhotika, and Stefano Soatto. Task adaptive parameter sharing for multi-task learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7561–7570, 2022. [2, 4](#)
- [48] Li Wang, Ziyang Song, Xinyu Zhang, Chenfei Wang, Guoxin Zhang, Lei Zhu, Jun Li, and Huaping Liu. Sat-gcn: Self-attention graph convolutional network-based 3d object detection for autonomous driving. *Knowledge-Based Systems*, 259:110080, 2023. [1](#)
- [49] Li Wang, Xinyu Zhang, Ziyang Song, Jiangfeng Bi, Guoxin Zhang, Haiyue Wei, Liyao Tang, Lei Yang, Jun Li, Caiyan Jia, et al. Multi-modal 3d object detection in autonomous driving: A survey and taxonomy. *IEEE Transactions on Intelligent Vehicles*, 2023. [1](#)
- [50] Qing Wang, Jiaming Zhang, Kailun Yang, Kunyu Peng, and Rainer Stiefelhagen. Matchformer: Interleaving attention in transformers for feature matching. *arXiv preprint arXiv:2203.09645*, 2022. [2](#)
- [51] Shuzhe Wang, Zakaria Laskar, Iaroslav Melekhov, Xiaotian Li, and Juho Kannala. Continual learning for image-based camera localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3252–3262, 2021. [1](#)
- [52] Shiguang Wang, Tao Xie, Jian Cheng, Xingcheng Zhang, and Haijun Liu. Mdl-nas: A joint multi-domain learning framework for vision transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20094–20104, 2023. [2](#)
- [53] Tao Xie, Kun Dai, Ke Wang, Ruifeng Li, Jiahe Wang, Xinyue Tang, and Lijun Zhao. A deep feature aggregation network for accurate indoor camera localization. *IEEE Robotics and Automation Letters*, 7(2):3687–3694, 2022. [1, 2, 6, 7, 8, 9](#)
- [54] Tao Xie, Kun Dai, Ke Wang, Ruifeng Li, and Lijun Zhao. Deepmatcher: A deep transformer-based network for robust and accurate local feature matching. *arXiv preprint arXiv:2301.02993*, 2023. [2](#)
- [55] Tao Xie, Ke Wang, Ruifeng Li, and Xinyue Tang. Visual robot relocalization based on multi-task cnn and image-similarity strategy. *Sensors*, 20(23):6943, 2020. [2](#)
- [56] Tao Xie, Li Wang, Ke Wang, Ruifeng Li, Xinyu Zhang, Haoming Zhang, Linqi Yang, Huaping Liu, and Jun Li. Farpnet: Local-global feature aggregation and relation-aware proposals for 3d object detection. *IEEE Transactions on Multimedia*, 2023. [1](#)
- [57] Tao Xie, Shiguang Wang, Ke Wang, Linqi Yang, Zhiqiang Jiang, Xingcheng Zhang, Kun Dai, Ruifeng Li, and Jian Cheng. Poly-pc: A polyhedral network for multiple point cloud tasks at once. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1233–1243, 2023. [1, 2](#)
- [58] Luwei Yang, Rakesh Shrestha, Wenbo Li, Shuaicheng Liu, Guofeng Zhang, Zhaopeng Cui, and Ping Tan. Scenesqueezers: Learning to compress scene for camera relocalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8259–8268, 2022. [1, 2](#)
- [59] Hailin Yu, Weicai Ye, Youji Feng, Hujun Bao, and Guofeng Zhang. Learning bipartite graph matching for robust visual localization. In *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 146–155. IEEE, 2020. [1, 2](#)
- [60] Ji Zhang and Sanjiv Singh. Visual-lidar odometry and mapping: Low-drift, robust, and fast. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2174–2181. IEEE, 2015. [8](#)
- [61] Lei Zhou, Zixin Luo, Tianwei Shen, Jiahui Zhang, Mingmin Zhen, Yao Yao, Tian Fang, and Long Quan. Kfnet: Learning temporal camera relocalization using kalman filtering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4919–4928, 2020. [1, 2, 5, 6, 8, 9](#)
- [62] Qunjie Zhou, Torsten Sattler, and Laura Leal-Taixe. Patch2pix: Epipolar-guided pixel-level correspondences. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4669–4678, 2021. [2](#)