EURASIP Journal on Advances in Signal Processing

RESEARCH Open Access

Accurate visual localization with semantic masking and attention

Tunan Li[1,2], Zhaohuan Zhan[1] and Guang Tan[1,2*]

*Correspondence: tanguang@mail.sysu.edu.cn

[1] School of Intelligent Systems Engineering, Shenzhen Campus of Sun Yat-sen University, Shenzhen, China
Full list of author information is available at the end of the article

Abstract

Visual localization is the task of accurate camera pose estimation within a scene and is a crucial technique for computer vision and robotics. Among the various approaches, relative pose estimation has gained increasing interest because it can generalize to new scenes. This approach learns to regress relative pose between image pairs. However, unreliable regions that contain objects such as the sky, persons, or moving cars are often present in real images, causing noise and interference to localization. In this paper, we propose a novel relative pose estimation pipeline to address the problem. The pipeline features a semantic masking module and an attention module. The two modules help suppress interfering information from unreliable regions, while at the same time emphasizing important features with an attention mechanism. Experiment results show that our framework outperforms alternative methods in the accuracy of camera pose prediction in all scenes.

Keywords: Relative pose estimation, Semantic segmentation, Attention module, Visual localization

1 Introduction

1.1 Background and significance

Visual localization is the technique of estimating a camera pose, that is, position and orientation in a scene. As a critical task in computer vision and robotics, visual localization has been widely studied in various domains, including Structure from Motion (SfM), Simultaneous Localization and Mapping, and Augmented Reality. It can provide location service in areas (e.g., indoor) where other localization techniques such as the Global Positioning System (GPS) fail to work, without worrying about losing signals from external infrastructure.

Classic visual localization methods follow a 3D structure-based approach [1–4]. They first find matches between local features [5–8] extracted from a query image and 3D point clouds of the scene obtained from SfM. The resulting correspondences are then used to estimate the camera pose by applying an n-point-pose solver. In recent years, learning-based algorithms have become popular with the success of deep convolution neural networks (CNNs). The CNN can generate a representation (i.e., a vector) of a scene based on a learned network model. Based on this, the absolute pose estimation (APE) approach [9–12] attempts to learn the entire localization pipeline. Given a set of training images and their poses, this approach trains a model that directly estimates the camera pose from a query image. Both 3D structure-based and APE approaches are scene specific: They cannot generalize to unseen scenes and need to reconstruct the 3D point clouds or re-train a model for new scenes.

In terms of model construction cost, a more adaptive localization pipeline is relative pose estimation (RPE) [13–16] that works in two stages. Given a query image, an image retrieval routine first returns several closest reference images from the database, whose absolute poses are known. The images are represented by compact descriptors, so the retrieving algorithm is almost unlimited by the scale of scenes. Subsequently, RPE calculates the relative pose of the query image to the reference images. Different from APE, RPE focuses on the view difference of two images. As such, the neural network model does not need to encode the known poses of the reference images and is therefore able to generalize to unseen scenes.

We consider the RPE approach, with a more fine-grained treatment of different types of information in an image. The design is motivated by the observation that not all contents in an image are beneficial to localization. The contents of an image can be roughly classified into two categories: *reliable* contents from static objects and *unreliable* contents from texture-less or moving objects. The RPE method estimates relative pose by correlating features that correspond to the same content in an image pair. Without a notion of reliability, traditional RPE methods may use unreliable contents that cause noise and interference to the localization algorithm.

We propose a novel relative pose estimation pipeline to address this problem. The pipeline features two modules, a semantic masking module and an attention module, as shown in Fig. 1. The former is a network named *BiasNet* that attaches a learned bias value to each image pixel. Generated by a semantic segmentation of an image, the bias values reflect the reliability of each pixel to localization. The second module, named *AttNet*, further refines the features of an image using an attention mechanism,

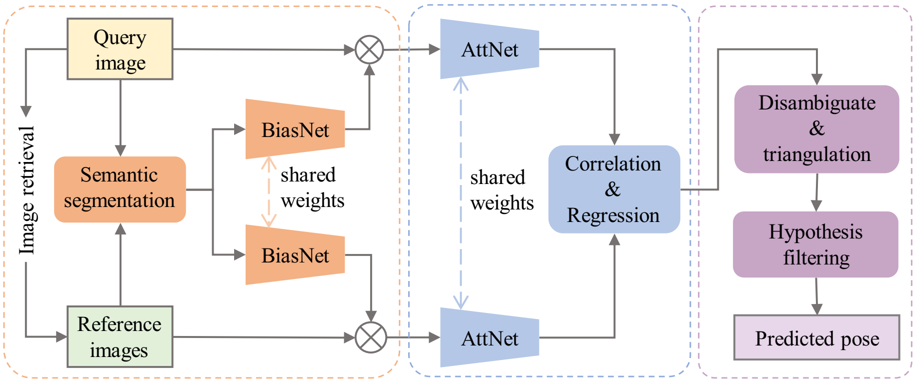

Fig. 1 The proposed relative pose estimate pipeline, with three steps: preparation (orange rectangle), pose regression (blue rectangle), and pose hypothesis filtering (purple rectangle). We use a siamese network architecture to fuse semantic information and extract features, then estimate relative pose by correlating features, and finally predict absolute pose via hypothesis filtering

aiming to emphasize features that are important to localization. Experimental results show the efficacy of our method compared with baseline methods.

To summarize, we make the following contributions:

- A new visual localization method based on relative pose estimation. Our work is among the first to consider the fine-grained treatment of different types of information in an image.
- We design a semantic masking module that suppresses potentially interfering objects in an image. It exploits the region boundaries generated by semantic segmentation, but otherwise does not need prior knowledge of the region category information;
- We propose an attention module that refines the image features. It adaptively emphasizes features that are important to localization;
- Experiments to show that our method outperforms alternative learning-based approaches, with reductions of position error by 12.5% and orientation error by 16.7%, compared to the baseline.

The rest of this paper is organized as follows. Section 1.2 discusses related work; Sect. 2 describes the localization pipeline; Sect. 3 shows our experimental details; the experimental results and ablation study are discussed in Sect. 4; finally, Sect. 5 concludes the paper.

1.2 Related work

Visual localization approaches can be classified into direct and indirect approaches. The former follows a one-stage strategy without the reference image, while indirect localization estimates the pose of the query image relative to the reference images after performing the image retrieval step.

1.2.1 Direct localization

3D structure-based localization. 3D structure-based localization [1, 2, 4, 17] methods depend on local feature techniques [5–8], including handcrafted methods and deep learning methods. These methods detect points of interest in an image and then produce descriptors for each point based on surrounding pixels. The points are matched by comparing the Euclidean distance between descriptors.

3D structure-based localization estimates pose based on 2D-3D matches between feature points in the query image and 3D points in the scene with an SfM model. This approach needs to build a 3D model that requires densely sampled reference images. As a result, it has a scalability problem when the scenes grow in size, due to memory consumption and increasing ambiguity in matching.

Image retrieval. Image retrieval [18] is typically used for place recognition, based on compact, lightweight descriptors using techniques like VLAD, DenseVLAD, NetVLAD [19–21]. It is mainly used for rough localization [20, 22, 23] by approximating the query image pose by the poses of the most similar images. Image retrieval is normally used as the first step in indirect localization approaches, including ours.

Absolute pose estimation. The APE methods [9–12] directly regress the absolute pose via training an end-to-end CNN model. They learn the complete localization pipeline, requiring images and their camera poses as training data. PoseNet [9] is the first end-to-end network based on the GoogLeNet to regress the camera pose; follow-up approaches improve the accuracy with enhanced backbones [10, 11]. The APE methods are significantly less accurate than structure-based methods. It is shown that APR is more closely related to image retrieval approaches [23]. Indeed, APE tries to represent the scenes implicitly by the weights of networks. Thus, the APE models are scene specific and do not generalize to unseen scenes, similar to the 3D structure-based methods.

1.2.2 Indirect localization

A two-stage localization pipeline first performs an image retrieval step and then estimates the pose of the query image relative to the reference images. The reference images have known absolute poses and often have overlapping views with the query image. From the absolute pose of the reference image and relative pose between the query and reference images, indirect localization obtains the absolute pose of the query image, that is, accomplishes visual localization.

Indirect feature-based localization. Indirect feature-based localization relies on local features, as with the 3D structure-based methods, except they do not need to build the 3D points model. By descriptors matching between the query image and similar reference images, they calculate the essential matrix by triangulation. Sarlin et al. [24] use a Graph Neural Network to implement the matching step.

Relative pose estimation. This approach learns to predict relative pose between image pairs with an end-to-end trainable neural network model. RPE is also used as visual odometry in some work [25, 26]. The RPE model does not encode images' absolute pose information, so need not modification for new scenes [27]. RPE can be less accurate than the feature-based approach, so many methods [13–16] have been proposed to improve it. Ding et al. [28] fuses depth information to improve accuracy, but obtaining depth information needs a specialized sensor which may be unavailable on many devices. We observe that semantic segmentation can generate additional information about specific regions' usefulness to localization, without using extra sensors. We therefore propose a new RPE method to take advantage of this observation.

2 Methods

In this section, we describe our localization pipeline. As shown in Fig. 1, the pipeline proceeds in three phases: preparation, pose regression, and pose hypothesis filtering. It first performs image retrieval and semantic segmentation and then uses a pose regression routine to predict a relative pose between a pair of images. Last, it recovers the absolute pose from the N relative pose estimates, with a pose hypothesis filtering routine. For ease of illustration, we show example images in Fig. 2a.

Problem Description. Visual localization is defined as estimating the absolute pose of a given query image $\mathcal{I}_q$. $(R_q, \mathbf{t}_q)$ is composed of a rotation matrix $R_q \in \mathbb{R}^{3\times 3}$ and a translation $\mathbf{t}_q \in \mathbb{R}^3$. Relative pose estimation is the problem of estimating the relative pose $(R_{r\rightarrow q}, \mathbf{t}_{r\rightarrow q})$ between one or multiple reference images $\mathcal{I}_r$ and the query image $\mathcal{I}_q$. With

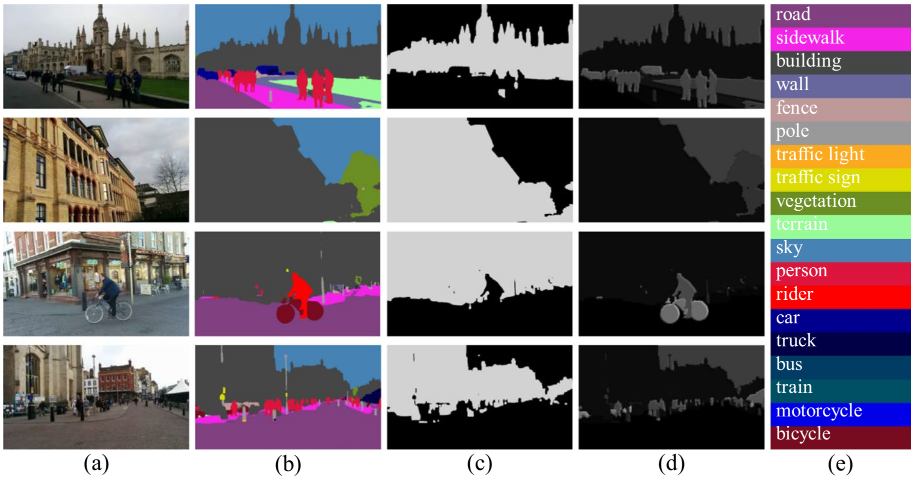

Fig. 2 Examples of each scene. **a** Query images. **b** Semantic segmentation results. **c** Visualization of prior knowledge-based masking. **d** Visualization of the feature maps $\mathcal{S}_i^3$ (second channel) generated by BiasNet. **e** Colormap of semantic categories for the Cityscapes dataset

the relative pose, we can then obtain the absolute pose of the query image based on the known absolute poses of reference images.

2.1 Preparation

Image retrieval. We collect five reference images via 4096-dimensional DenseVLAD descriptors, following Zhou et al. [16]. To ensure larger triangulation angles while still keeping enough visual overlap for successful relative pose estimation, starting with the top retrieved image, we iteratively select the next image that has a distance within $[a, b]$ 4 meters to all previously selected images. We use the same values as Zhou et al. [16], i.e., $a = 3, b = 50$.

Semantic segmentation. We perform segmentation to extract region information from the query image and the reference images in the database, using the PSPNet by Zhao et al.[29]. PSPNet provides a powerful framework for pixel-level semantic segmentation with a pyramid pooling module. For each RGB image $\mathcal{I}_i \in \mathbb{R}^{H \times W \times 3}$ that has $H \times W$ pixels, we can execute the PSPNet and then obtain a semantic map $\mathcal{S}_i^0 \in \mathbb{R}^{H \times W}$. Here $\mathcal{S}_i^0(x, y) \in C = \{c_1, \ldots, c_m\}$, where C is the set of semantic classes. Figure 2b, e show the semantic segmentation results of example images and the colormap of semantic categories of Cityscapes [9].

To preserve the boundaries of some critical semantic categories, we perform a dilation process on a semantic map, inspired by the morphology operation. Specifically, for each pixel $\mathcal{S}(x, y)$ in the semantic map, if $\mathcal{S}(x, y) \in C_{\text{dilation}}$, then we use a 7×7 kernel with values $\mathcal{S}(x, y)$ to replace the 7×7 pixels patch whose center is (x, y).

Bias network We propose a small CNN-based bias network which decodes the semantic map to a bias feature map. The architecture of BiasNet is shown in Fig. 3a. BiasNet consists of two 3×3 convolutional layers and one 1×1 convolutional layer:

$$\mathcal{S}_i^k = Relu(f^k(\mathcal{S}_i^{k-1})), \tag{1}$$

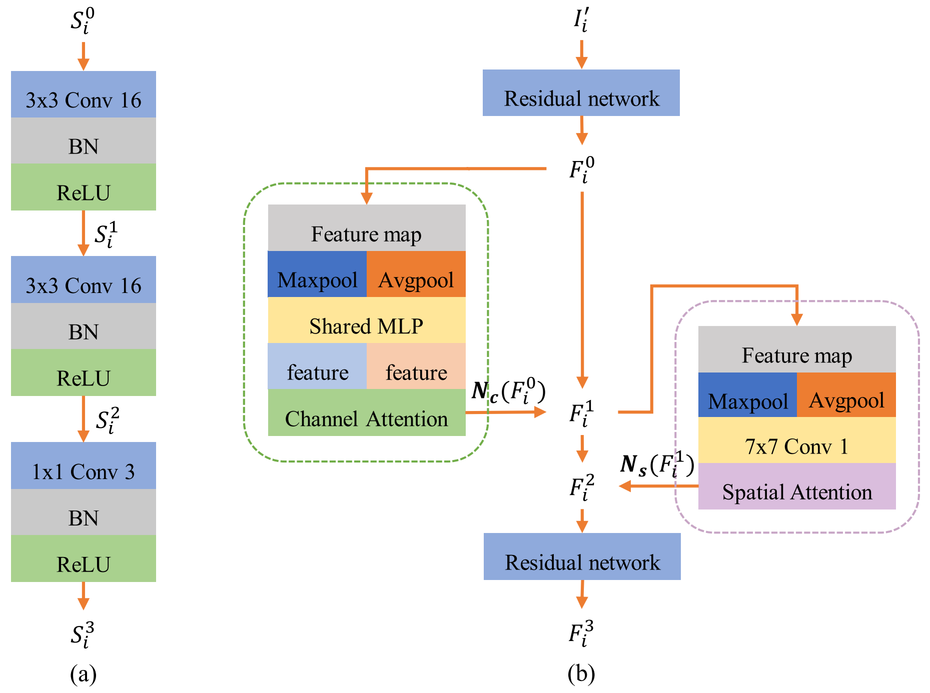

Fig. 3 The architectures of the proposed modules. **a** The bias network, composed of three convolutional layers. **b** The attention network, composed of a residual network with one Convolutional Block Attention Module

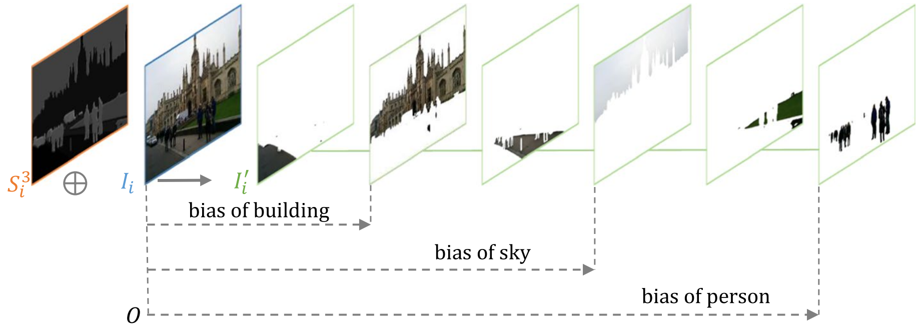

Fig. 4 Element-wise addition of $\mathcal{I}_i$ and $\mathcal{S}_i^3$. Various bias values are added to different categories. Note that $\mathcal{I}_i$ has actually been normalized, for ease of presentation, we display the unprocessed image

where $k = 1, 2, 3$, f^1,f^2,f^3 represent convolutional operations with filter sizes $3 \times 3, 3 \times 3, 1 \times 1$, $\mathcal{S}_i^0$ is the semantic map, $\mathcal{S}_i^3$ is the final layer output, and $\mathcal{S}_i^3 \in \mathbb{R}^{H \times W \times 3}$. Figure 2d shows the visualization of $\mathcal{S}_i^3$. Then we perform an element-wise addition, which is more computation efficient than concatenation, between $\mathcal{I}_i$ and $\mathcal{S}_i^3$, that is,

$$\mathcal{I}_i' = \mathcal{I}_i \oplus \mathcal{S}_i^3, \tag{2}$$

where $\oplus$ denotes element-wise addition, $\mathcal{I}_i' \in \mathbb{R}^{H \times W \times 3}$, and $\mathcal{I}_i'$ is then fed into the following network. Figure 4 illustrates the process of the element-wise addition, bias values reflect the reliability of different categories.

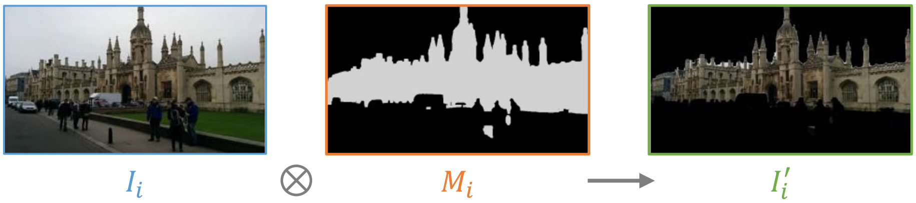

Fig. 5 Element-wise multiplication of $\mathcal{I}_i$ and $\mathcal{M}_i$

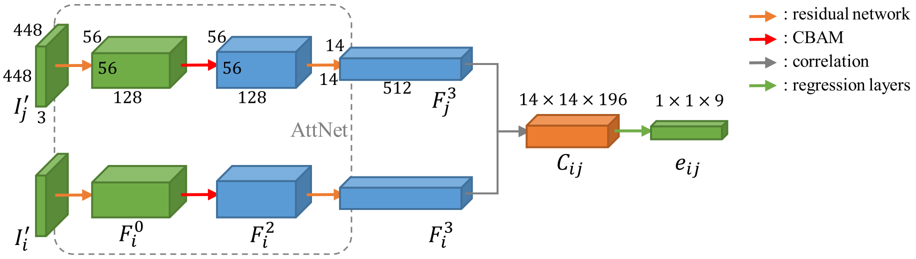

Fig. 6 Illustration of pose regression

We compare BiasNet with a *Prior Knowledge-based Mask* method inspired by [30], or *Prior Mask* for short. The method masks certain image regions and reserves the others according to semantic categories:

$$\mathcal{M}_i(x,y) = \begin{cases} 1, & \text{if } \mathcal{S}_i^0(x,y) \in C_{\text{mask}} \\ 0, & \text{if } \mathcal{S}_i^0(x,y) \notin C_{\text{mask}} \end{cases}, \tag{3}$$

where C_{mask} is the set of reserved categories that can be manually set. For example, the "building" category is normally considered reliable [8] for localization and so is set in C_{mask}. Then we multiply the reference image $\mathcal{I}_i'$ with the mask $\mathcal{M}_i \in \mathbb{R}^{H \times W}$:

$$\mathcal{I}_i' = \mathcal{I}_i \otimes \mathcal{M}_i, \tag{4}$$

where $\otimes$ denotes element-wise multiplication, $\mathcal{I}_i'$ is the counterpart of BiasNet results in Eq. 2. Figure 2c shows the visualization of $\mathcal{M}_i$, and Fig. 5 illustrates how the element-wise multiplication implements Prior Mask.

2.2 Pose regression

We use a siamese neural network [31] to estimate the relative pose between the query and reference images. The backbone consists of three modules: feature extraction module, feature correlation module, and pose regression module. Figure 6 shows an example of inputting an image pair with 448 × 448 pixels.

Feature extraction. Different from [13, 16] that extract features by ResNet34 [41] , we propose an attention extraction network, AttNet, to extract more distinctive features. As shown in Fig. 3b, AttNet consists of a residual network and Convolutional Block

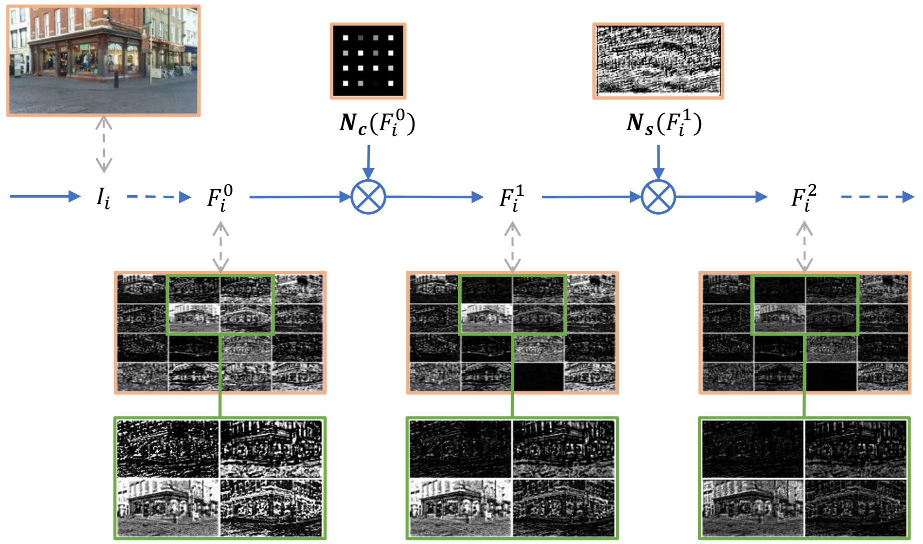

Fig. 7 Visualization of the feature maps of AttNet. Here $\mathcal{N}_c(\mathcal{F}_i^0)$, $\mathcal{N}_s(\mathcal{F}_i^1)$ are feature maps of channel attention and spatial attention, respectively. Note we only show 16 channels out of 64 channels

Attention Module (CBAM) proposed by [32]. The CBAM has shown improvements in classification and object detection, and our method confirms its effectiveness in feature extraction of visual localization tasks. The input is $\mathcal{I}_i'$, as obtained in Sect. 2.1:

$$\mathcal{F}_i^0 = ResNet(\mathcal{I}_i{}'), \tag{5}$$

where $\mathcal{F}_i^0 \in \mathbb{R}^{h \times w \times c}$, for the 448 × 448 pixels image, $h = w = 56, c = 128$ as shown in Fig. 6, ResNet corresponds to the residual network. CBAM sequentially infers attention maps in two separate dimensions, channel and spatial, and then the attention maps are multiplied with the input feature map for feature refinement:

$$\mathcal{N}_c(\mathcal{F}) = \sigma(MLP(AvgPool(\mathcal{F})) + MLP(MaxPool(\mathcal{F}))), \tag{6}$$

$$\mathcal{N}_s(\mathcal{F}) = \sigma(f^{7\times7}([AvgPool(\mathcal{F}); MaxPool(\mathcal{F})])), \tag{7}$$

$$\mathcal{F}_i^1 = \mathcal{N}_c(\mathcal{F}_i^0) \otimes \mathcal{F}_i^0, \quad \mathcal{F}_i^2 = \mathcal{N}_s(\mathcal{F}_i^1) \otimes \mathcal{F}_i^1. \tag{8}$$

Then the remaining network:

$$\mathcal{F}_i^3 = ResNet(\mathcal{F}_i^2), \tag{9}$$

where $\mathcal{F}_i^3$ is the extracted feature of $\mathcal{I}_i$, and $\mathcal{F}_i^3 \in \mathbb{R}^{h' \times w' \times c'}$ for the 448 × 448 pixels image, $h' = w' = 14, c = 512$. Figure 3b illustrates the process of AttNet, and Fig. 7 illustrates how attention works.

Feature correlation. After the feature extraction, we obtain two feature maps with $h' \times w' \times c'$ dimensions from a query image and a reference image, that is,

$\mathcal{F}_i^{(3)}, \mathcal{F}_j^{(3)} \in \mathbb{R}^{h'\times w'\times c'}$. We use a matrix dot product operation between the feature maps to associate two images as [33], which produces a correlation feature map with $h'\times w'\times(h'\times w')$ dimensions that includes information of feature matching, that is,

$$\mathcal{C}_{ij}(x_i, y_i, c) = \mathcal{F}_i^{(3)}(x_i, y_i)^T \mathcal{F}_j^{(3)}(x_j, y_j), \tag{10}$$

where $c = w'\times x_j + y_j$, $x_i, x_j \in \{0, \ldots, h'-1\}$, and $y_i, y_j \in \{0, \ldots, w'-1\}$.

Pose regression. The correlation feature map is fed to regression layers, which consist of two convolution layers and one fully connected layer:

$$\mathbf{e}_{ij} = FC(Relu(f^{1\times 1}(Relu(f^{3\times 3}(\mathcal{C}_{ij}))))), \tag{11}$$

where $\mathbf{e}_{ij} \in \mathbb{R}^9$ as shown in Fig. 6. During training, we use mean-square loss function to minimize the Euclidean distance between the predicted essential matrix E and the ground truth essential matrix E^*. Note that the essential matrix is a 3×3 matrix. The loss function is:

$$\mathcal{L}_{ess}(E^*, E) = \left\|\mathbf{e}^* - \mathbf{e}\right\|_2. \tag{12}$$

Here, $\mathcal{L}_{ess}$ is the loss, $\mathbf{e}_{ij} \in \mathbb{R}^9$ is the vectorized $E \in \mathbb{R}^{3\times 3}$. Given the relative pose label $(R^*, \mathbf{t}^*)$, the ground truth essential matrix is $E^* = [t^*]_\times R^*$, where $[t^*]_\times$ is the skew-symmetric matrix of the normalized translation t^*.

2.3 Pose hypothesis filtering

The final part of the proposed pipeline is a RANSAC algorithm that estimates the absolute pose of the query image. Consider N reference images $\mathcal{I}_r = \{\mathcal{I}_{r1}, ..., \mathcal{I}_{rk}, ..., \mathcal{I}_{rN}\}$, we estimate N essential matrices E that encode the relative pose between the considered image pairs. There are four possible relative poses $(R, \mathbf{t})$, $(R, -\mathbf{t})$, $(R', \mathbf{t})$, $(R', -\mathbf{t})$ corresponding to a certain E, where R and R' are related by a 180° rotation. We can disambiguate the two rotations by comparing the angle difference between the possible absolute rotations of multiple image pairs. The signs of any direction would not change the position of a point triangulated from multiple directions. Therefore, the absolute pose of the query image can be uniquely determined by at least two image pairs.

We use the RANSAC algorithm to filter hypothesis following [13, 16]. Consider a query image $\mathcal{I}_q$ and an iteration $p^s = \{\mathcal{I}_{ri}, \mathcal{I}_{rj}\}$ selected from the reference images set $\mathcal{I}_r$, which contains two image pairs $((\mathcal{I}_{ri}, \mathcal{I}_q), (\mathcal{I}_{rj}, \mathcal{I}_q))$, where $p^s \subset \mathcal{I}_r$ and $s = 1, 2, ..., \binom{N}{2}$, $\binom{\cdot}{\cdot}$ means combination, and N represents the number of reference images. We obtain four absolute rotations $R_i R_{\mathcal{I}_{ri}}, R_i' R_{\mathcal{I}_{ri}}, R_j R_{\mathcal{I}_{rj}}, R_j' R_{\mathcal{I}_{rj}}$. Two of them will be identical in theory, while the two with the smallest angle difference will be considered true in practice. Suppose R_i and R_j are consistent by relative rotation, the position of the query image can be determined by triangulation from two rays $\mathbf{c}_{\mathcal{I}_{ri}} + \lambda_i R^T_{\mathcal{I}_{ri}} R_i \mathbf{t}_i$ and $\mathbf{c}_{\mathcal{I}_{rj}} + \lambda_j R^T_{\mathcal{I}_{rj}} R_j \mathbf{t}_j$, where $\mathbf{c}_{\mathcal{I}}$ denotes the camera center of $\mathcal{I}$ computed by $\mathbf{c}_{\mathcal{I}} = -R_{\mathcal{I}}^{T}\mathbf{t}_{\mathcal{I}}$, and $\lambda_i, \lambda_j \in \mathbb{R}$ define the positions of the points along the rays. Notice that we use quaternion to represent the rotation and $N = 5$ in practice.

For each iteration p^s, we estimate an absolute pose hypothesis $(R_{\mathcal{I}_q}, \mathbf{t}_{\mathcal{I}_q})$ of the query image as shown above. This gives us $\begin{pmatrix} 5 \\ 2 \end{pmatrix}$ hypotheses for the query image. Next, we determine which image pairs are inliers to each pose hypothesis. For a pair $(\mathcal{I}_{rk}, \mathcal{I}_q)$, we first determine the rotation R_k that minimizes the angle between the $R_k R_{\mathcal{I}_{rk}}$ and the hypothesis rotation $R_{\mathcal{I}_q}$. Subsequently, we perform the consistency measure:

$$\alpha = arccos(\frac{\mathbf{t}_k^T \mathbf{t}_{\text{hypo}}}{\|\mathbf{t}_k\|_2 \left\|\mathbf{t}_{\text{hypo}}\right\|_2}), \tag{13}$$

where $\mathbf{t}_{\text{hypo}}$ is the relative translation between $\mathcal{I}_q$ to $\mathcal{I}_{rk}$ calculated by $\mathbf{t}_{\text{hypo}} = R_{\mathcal{I}_{rk}} (\mathbf{c}_{\mathcal{I}_q} - \mathbf{c}_{\mathcal{I}_{rk}})$. If the angle is below a given threshold $\alpha_{\max}$, the pair $(\mathcal{I}_{rk}, \mathcal{I}_q)$ is considered an inlier. RANSAC finally returns the pose with the maximum number of inliers. Furthermore, if the number of inliers is not enough, we use the top-retrieved image pose as the predicted pose of the query image.

The proposed method is summarized in Algorithm 1.

Algorithm 1 The proposed relative pose estimation algorithm.

```
Input:  $\mathcal{I}_q$, $\mathcal{I}_{db}$, $(R_{db}, \mathbf{t}_{db})$;
Output:  $(R_q, t_q)$;
Retrieve five most similar reference images $\mathcal{I}_r$ for $\mathcal{I}_q$ from the database $\mathcal{I}_{db}$.
Put $\mathcal{I}_q$ and each $\mathcal{I}_{rk}$ into PSPNet to obtain the semantic maps $\mathcal{S}_q$, $\mathcal{S}_{rk}$, where k=1,2,3,4,5.
Make pairs of $\mathcal{I}_q$, $\mathcal{I}_{rk}$, $\mathcal{S}_q$, $\mathcal{S}_{rk}$, and feed them into BiasAttNet.
Obtain five essential matrices $E_k$ between image pairs.
Perform pose hypothesis filtering with RANSAC:
for Each iteration $p^s = \{\mathcal{I}_{ri}, \mathcal{I}_{rj}\}$, where $s = 1, 2, ..., \binom{5}{2}$ do
    Estimate an absolute pose $(R_{\mathcal{I}_q}, \mathbf{t}_{\mathcal{I}_q})$ for $\mathcal{I}_q$ via disambiguation and triangulation.
    Determine the number of inlier image pairs through consistency check.
end for
if There is an iteration with no less than two inliers then
    Return the pose with the maximum number of inliers.
else
    Use the top-retrieved image pose $(R_{\mathcal{I}_{r1}}, \mathbf{t}_{\mathcal{I}_{r1}})$ as the predicted pose.
end if
return  $(R_q, t_q)$
```

3 Experiments

In this section, we describe the evaluation protocol and implementation details of the proposed method.

Datasets. We evaluate our method on the Cambridge Landmarks [9] dataset, including King's College, Old Hospital, Shop Facade, and St Mary's Church scenes. Table 1 and Fig. 2a show the spatial extent and example images of four outdoor scenes. We adopt the PSPNet [29] as our semantic segmentation model which is pre-trained on the Cityscapes [34] outdoor dataset.

Table 1 Dataset statistics and percentage of building category pixels

	Kings' College	Old Hospital	Shop Facade	St M. Church
#Training images	1220	895	231	1487
#Test images	343	182	103	530
Spatial extent	140 × 40m	50 × 40m	35 × 25m	80 × 60m
%Building pixels	56.30%	73.70%	74.90%	66.60%

The building category occupies most part of the scene images

Implementation details. For semantic map dilation and the Prior Mask method, we set $C_{\text{dilation}} = C_{\text{mask}} = \{c_{\text{building}}\}$, since the building category is the most stable and reliable basis for localization. Table 1 shows the percentage of the dilated building category pixels in the whole image for different scenes. It can be seen that the building category occupies most part of the scene images. The masks obtained after dilation and reservation are shown in Fig. 2c.

We use the same training pairs as Zhou et al. [16] for a fair comparison. AttNet is initialized with ResNet34 weights pre-trained on ImageNet [35]; the CBAM, BiasNet, and regression network layers are initialized with Kaiming initialization [36]. The BiasNet is trained with the backbone together. All trained images are first rescaled to 853 × 480 pixels, and then randomly cropped for training and center cropped for testing, both to 448 × 448 pixels, followed by normalization. For ease of presentation, we used 853 × 480 resolution images in all figures. All models are trained with the AdamOptimizer [37] with learning rate $1e^{-4}$ and weight decay $1e^{-6}$ in a batch size of 16 for at most 60 epochs. We verify our model every six epochs, sort the results according to the pass rate of the test images within the error range of 5m, 10°, and use the best one as the final result. The code is implemented using Pytorch [38], and all the experiments are conducted on an NVIDIA 2080Ti GPU.

4 Results and discussion

In this section, we discuss the performance results. We first show that the proposed approach outperforms the baselines, and then evaluate the impact of each module by ablation study; finally we analyze the working principle of each module.

4.1 Comparison with other approaches

We compare our approach with various methods, including 3D structure-based (3D) method [17], image retrieval (IR) [20, 23], absolute pose estimation (APE) [9–12], indirect feature-based localization (IFB), and relative pose estimation (RPE) [16]. We use SIFT and RANSAC implementation provided by OpenCV [39] as the indirect feature-based method. In particular, we use a recent RPE method EssNet as a baseline for comparison. Following the same convention of prior work [9–12, 16, 17, 20, 23], we compute the median absolute position error in meters and the median absolute rotation error in degrees for all scenes and methods. Table 2 shows that our approach consistently outperforms all IR and APE methods, as well as EssNet. In particular, it surpasses all pure learning-based approaches. For MapNet [12], our approach performs better overall, except for a slightly larger error in orientation. For visibility reasons, we show the results of several classic methods in Fig. 8.

Table 2 Results on the Cambridge Landmarks dataset

		Kings' College	Old Hospital	Shop Facade	St M. Church	Average
3D	*Active Search [17]	0.48m, 0.67°	0.81m, 1.15°	0.17m, 0.65°	0.36m, 1.00°	0.46m, 0.87°
IFB	*SIFT + RANSAC	0.49m, 0.70°	1.04m, 1.29°	0.19m, 0.67°	0.36m, 1.03°	0.52m, 0.92°
IR	DenseVLAD (D.VLAD) [20]	2.80m, 5.72°	4.01m, 7.13°	1.11m, 7.61°	2.31m, 8.00°	2.56m, 7.12°
	D.VLAD + Inter. [23]	1.48m, 4.45°	2.68m, 4.63°	**0.90m**, 4.32°	1.62m, 6.06°	1.67m, 4.87°
APE	PoseNet (PN) [9]	1.92m, 5.40°	2.31m, 5.38°	1.46m, 8.08°	2.65m, 8.48°	2.09m, 6.84°
	Bay. PN [10]	1.74m, 4.06°	2.57m, 5.14°	1.25m, 7.54°	2.11m, 8.38°	1.92m, 6.28°
	LSTM PN [11]	0.99m, 3.65°	1.51m, 4.29°	1.18m, 7.44°	**1.52m**, 6.68°	1.30m, 5.52°
	MapNet [12]	1.07m, **1.89°**	1.94m, 3.91°	1.49m, **4.22°**	2.00m, **4.53°**	1.63m, **3.64°**
RPE	EssNet [16]	0.82m, 2.32°	1.58m, 4.37°	1.29m, 6.32°	1.74m, 6.17°	1.36m, 4.80°
	BiasAttNet (ours)	**0.75m**, 2.26°	**1.44m, 3.23°**	0.95m, 5.24°	1.62m, 5.27°	**1.19m**, 4.00°

We compare our approach against 3D structure-based method (3D), image retrieval (IR), indirect feature-based localization (IFB), absolute and relative pose estimation (APE and RPE) methods. We report the median position error in meters and orientation error in degrees. The best results are highlighted in bold except for methods marked with a *

Compared to EssNet, our approach improves the position accuracy consistently in all scenes, achieving a position error reduction of 12.5% and an orientation error reduction of 16.7% on average. In particular, for the Shop Facade scene, the position error is reduced by as much as 35.8%, and for the Old Hospital scene, the orientation error is reduced by 25.9%. Furthermore, we take the 100 images in each scene that have the largest pixel ratio of obstacles and moving objects. Then we report the median error in these image sets. From Fig. 9, we can see that the error increases because of the increased challenge, but our method remains superior to the baseline.

4.2 Ablation study

By integrating our modules progressively, we show the impact of the individual modules of our pipeline. Table 3 shows the results of the baseline, BiasNet, AttNet, and our approach. For ease of observation, we show results in Fig. 10. As can be seen, each module positively affects the pipeline, but there is some fluctuation with scenes. The pipeline has the best performance with both BiasNet and AttNet, showing stable improvements in all scenes.

Furthermore, we compare our approach to the Prior Mask method described in Sect. 2.1; the results are shown in Table 3. Looking at Tables 1 and 3, we see that with Prior Mask, some image contents are unhelpful or harmful to localization performance. Overall, Prior Mask performs slightly better than the baseline. That is, the building category provides sufficient support for localization. However, there is no accuracy improvement when integrating the attention module into the pipeline with Prior Mask. Rather than using a fixed mask, BiasNet adaptively fuses semantic information and obtains better accuracy. The bias feature maps $\mathcal{B}_i^3$ produced by BiasNet are shown in Fig. 2d. Different from the Prior Mask method, BiasNet learns to return different bias values for different categories, that is, filter image contents with bias values. The semantic fusion provided by BiasNet is at the category level, as with the bias feature map. From the results, we find that the 19 categories are approximately divided into two types: reliable, often unchanging, categories such as buildings and roads, and unreliable, often changing, categories, such as persons, the sky, and vegetation. BiasNet gives each category a different bias value that agrees with humans' common knowledge.

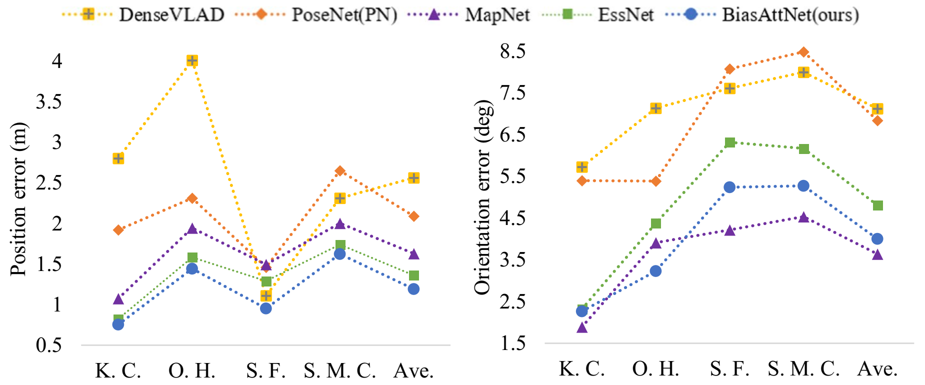

Fig. 8 Position and orientation errors of the various methods on the Cambridge Landmarks dataset. (K.C. stands for King's College, O.H. for Old Hospital, S.F. for Shop Facade, S.M.C. for St Mary's Church, and Ave. for Average.)

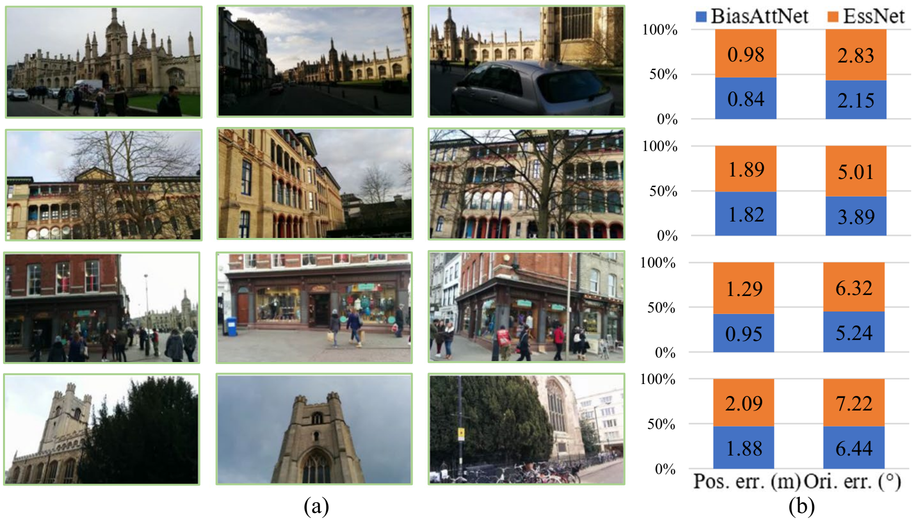

Fig. 9 Performance results of challenging images. **a** Some representative query images that contain interfering objects. **b** Stacked bar charts of position error and orientation error

We also evaluated the performance of the dilation strategy. Due to imperfect accuracy, the segmentation method cannot distinguish precisely the connecting regions between the critical categories and others. This will cause slight damage to the boundaries of critical categories which are useful for localization. The 7 × 7 kernel allows the boundaries of the critical categories to grow 3 pixels outward, so the details of the critical categories can be completely retained. Comparing the overall localization errors of BiasNet and BiasNet without dilation, as Table 3 shows, we see that the dilation strategy can reduce localization error.

To validate the benefit of semantic information to localization, we replace $\mathcal{I'}_i$ in Sect. 2.1 with $\mathcal{B}_i^3$, that is, using semantic maps as the only input. As Table 3 shows that semantics is effective in improving accuracy. Furthermore, we remove the hypothesis filtering step in our pipeline and use $\{\mathcal{I}_{r1}, \mathcal{I}_{r2}\}$ as the only hypothesis and use it as pose

Table 3 Results of ablation study

	Kings' College	Old Hospital	Shop Facade	St M. Church	Average
EssNet	0.82m, 2.32°	1.58m, 4.37°	1.29m, 6.32°	1.74m, 6.17°	1.36m, 4.80°
BiasNet	0.79m, 2.77°	1.46m, 4.26°	1.27m, 6.63°	**1.51m**, 6.79°	1.26m, 4.99°
BiasNet (without dilation)	0.82m, 2.80°	**1.40m**, 4.00°	1.48m, 9.30°	1.52m, 6.27°	1.31m, 5.59°
AttNet	0.84m, 2.28°	1.60m, 4.41°	1.19m, **4.96°**	1.57m, 5.56°	1.30m, 4.30°
BiasNet + AttNet	0.75m, 2.26°	1.44m, **3.23°**	**0.95m**, 5.24°	1.62m, **5.27°**	**1.19m, 4.00°**
BiasAttNet (without RANSAC)	0.82m, **2.11°**	1.60m, 3.96°	1.00m, 5.39°	1.71m, 5.77°	1.28m, 4.31°
Prior Mask + ResNet	**0.72m**, 2.22°	1.68m, 3.62°	1.17m, 6.57°	1.78m, 6.29°	1.34m, 4.68°
Prior Mask + AttNet	0.84m, 2.25°	1.63m, 3.78°	1.30m, 6.84°	1.55m, 5.38°	1.33m, 4.56°
Semantics only	1.43m, 3.30°	4.32m, 9.87°	3.23m, 16.01°	4.53m, 21.04°	3.38m, 12.56°
SIFT + RANSAC (448 × 448)	0.63m, 0.79°	1.75m, 1.81°	0.35m, 1.47°	0.50m, 1.47°	0.81m, 1.38°
SIFT + RANSAC + Semantics	0.59m, 0.81°	1.58m, 2.10°	0.32m, 1.50°	0.54m, 1.56°	0.76m, 1.49°

We compare the pipeline that has individual modules removed or replaced. The best results are highlighted in bold except for methods marked with a *

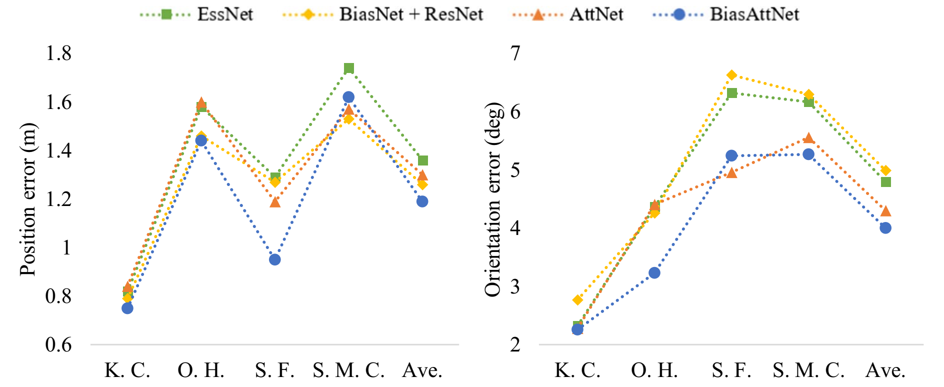

Fig. 10 Position and orientation errors of the baseline and the proposed pipeline. We demonstrate the proposed modules are effective by progressively integrating the modules. (K.C. stands for King's College, O.H. for Old Hospital, S.F. for Shop Facade, S.M.C. for St Mary's Church, and Ave. for Average.)

estimation. The results show the hypothesis filtering makes our approach robust to outliers.

We are also interested in the effect of semantics in feature-based approaches. To this end, we use SIFT and RANSAC to implement the indirect feature-based method and use 448 × 448 pixels center cropped images as our pipeline. We filter matches between different categories in the feature match step as [40]. It almost produces no gain from semantics for feature-based methods, as shown in Table 3.

4.3 Effectiveness of semantic masking and attention

In this subsection, we will show and explain how do our proposed modules work.

How does semantic masking work? The semantic mask is at pixel level and contains different categories of objects. By integrating the semantics map, the extracted features contain this valuable category information. As analyzed in Sect. 4.2, our approach filters

Table 4 Parameters sizes and GFLOPs of the proposed network

Description	BiasNet	AttNet	Regression layers	BiasAttNet
Parameters size	2.60K	21.29M	3.18M	21.61M
GFLOPs	0.52	14.68	0.03	30.45

Table 5 Execution time of our localization approach

		Kings' College	Old Hospital	Shop Facade	St M. Church	Mean execution time
# image pairs		1715	910	515	2650	
BiasAttNet	Total time	102.81 s	67.17 s	33.92 s	182.49 s	67 ms
	Time per pair	60 ms	73 ms	65 ms	68 ms	
SIFT + RANSAC	Total time	352.29 s	310.38 s	109.36 s	537.13 s	226 ms
	Time per pair	205 ms	341 ms	212 ms	202 ms	

some categories and retains others. In the correlation step later on, the same semantic categories produce enhanced responses, and the different categories produce suppressed responses.

How does attention work? We select one image in Shop Facade as an example and visualize the feature maps with torchvision [38], as depicted in Fig. 7. Comparing the feature maps $\mathcal{F}_i^0$ and $\mathcal{F}_i^1$, we see that the brightness of each channel feature maps drops to varying degrees, that is, some channels are suppressed in the feature map. The visualization of the channel attention feature map $\mathcal{N}_c(\mathcal{F}_i^0)$ shows a lowered degree of each channel. From the visualization of spatial attention feature maps $\mathcal{N}_s(\mathcal{F}_i^1)$, we find that the spatial feature map is similar in structure to the input image, and some information is filtered out in the spatial dimension. In RPE localization, the task of the attention network is to retain distinctive features of the image which make it easy to match other features when comparing images. Because the attention module suppresses meaningless feature maps along the channel and spatial dimensions, AttNet can play an essential role in our approach.

4.4 Computational complexity

Table 4 shows the parameters and GFLOPs of our modules. As can be seen, the overall overhead of BiasNet is quite small in terms of both parameters and computation. We report the computation time of test images in Table 5. Our approach takes 67 ms per image pair on an NVIDIA 2080Ti GPU. That makes the localization of a query image, which involves five image pairs, take 334 ms (3 FPS). In comparison, the indirect feature-based localization (SIFT + RANSAC) method takes 1131 ms for a query image on an Intel Core i7-9700 CPU.

5 Conclusion

In this paper, we have proposed a novel relative pose estimation pipeline. We show that different regions in the image play different roles, some positive and others negative, in the localization task. Our approach can filter out interference information brought by the changing objects in the image, at both pixel and feature levels. We design a network module that incorporates semantic information by segmentation, and show its benefit to localization. We also propose an attention feature extraction network that refines reliable information and extracts more distinctive features. We use hypothesis filtering to make our approach robust to outliers, and verify its effect with ablation experiments. The results show our pipeline outperforms all learning-based methods.

Abbreviations

CNN: Convolution neural networks; SfM: Structure from Motion; APE: Absolute pose estimation; RPE: Relative pose estimation; BiasNet: Bias network; AttNet: Attention network; CBAM: Convolutional Block Attention Module; K.C.: King's College; O.H.: Old Hospital; S.F.: Shop Facade; S.M.C.: St Mary's Church; Ave.: Average.

Acknowledgements

The authors would like to thank the anonymous reviewers for their valuable suggestions.

Author contributions

TNL developed and implemented the core network of the proposed approach, ZHZ contributed the idea of attention module, GT provided the general idea of the work. All authors read and approved the final manuscript.

Funding

The work was supported in part by Shenzhen Basic Research Fund under grant JCYJ20200109142217397.

Availability of data and materials

[9, 34] provided the Cambridge Landmarks dataset of outdoor scenes and the Cityscapes dataset of semantic segmentation, respectively.

Declarations

Ethics approval and consent to participate

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Consent for publication

Not applicable.

Author details

[1]School of Intelligent Systems Engineering, Shenzhen Campus of Sun Yat-sen University, Shenzhen, China. [2]Guangdong Provincial Key Laboratory of Fire Science and Intelligent Emergency Technology, Guangzhou, China.

Received: 23 November 2021 Accepted: 23 April 2022
Published online: 16 May 2022

References

1. J.L. Schonberger, J.-M. Frahm, Structure-from-motion revisited, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 4104–4113
2. E. Brachmann, C. Rother, Learning less is more-6d camera localization via 3d surface regression, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 4654–4662
3. P.-E. Sarlin, C. Cadena, R. Siegwart, M. Dymczyk, From coarse to fine: robust hierarchical localization at large scale, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2019), pp. 12716–12725
4. J.L. Schönberger, M. Pollefeys, A. Geiger, T. Sattler, Semantic visual localization, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 6896–6906
5. D.G. Lowe, Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
6. K.M. Yi, E. Trulls, V. Lepetit, P. Fua, Lift: learned invariant feature transform, in *European Conference on Computer Vision* (Springer, 2016), pp. 467–483
7. D. DeTone, T. Malisiewicz, A. Rabinovich, Superpoint: self-supervised interest point detection and description, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops* (2018), pp. 224–236
8. J. Revaud, P. Weinzaepfel, C. De Souza, N. Pion, G. Csurka, Y. Cabon, M. Humenberger, R2d2: repeatable and reliable detector and descriptor (2019). arXiv:1906.06195
9. A. Kendall, M. Grimes, R. Cipolla, Posenet: A convolutional network for real-time 6-dof camera relocalization, in *Proceedings of the IEEE International Conference on Computer Vision* (2015), pp. 2938–2946
10. A. Kendall, R. Cipolla, Modelling uncertainty in deep learning for camera relocalization, in *2016 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2016), pp. 4762–4769

11. F. Walch, C. Hazirbas, L. Leal-Taixe, T. Sattler, S. Hilsenbeck, D. Cremers, Image-based localization using lstms for structured feature correlation, in *Proceedings of the IEEE International Conference on Computer Vision* (2017), pp. 627–637
12. S. Brahmbhatt, J. Gu, K. Kim, J. Hays, J. Kautz, Geometry-aware learning of maps for camera localization, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 2616–2625
13. Z. Laskar, I. Melekhov, S. Kalia, J. Kannala, Camera relocalization by computing pairwise relative poses using convolutional neural network, in *Proceedings of the IEEE International Conference on Computer Vision Workshops* (2017), pp. 929–938
14. V. Balntas, S. Li, V. Prisacariu, Relocnet: continuous metric learning relocalisation using neural nets, in *Proceedings of the European Conference on Computer Vision (ECCV)* (2018), pp. 751–767
15. S. Saha, G. Varma, C. Jawahar, Improved visual relocalization by discovering anchor points (2018). arXiv:1811.04370
16. Q. Zhou, T. Sattler, M. Pollefeys, L. Leal-Taixe, To learn or not to learn: Visual localization from essential matrices, in *2020 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2020), pp. 3319–3326
17. T. Sattler, B. Leibe, L. Kobbelt, Efficient & effective prioritized matching for large-scale image-based localization. IEEE Trans. Pattern Anal. Mach. Intell. **39**(9), 1744–1756 (2016)
18. H. Noh, A. Araujo, J. Sim, T. Weyand, B. Han, Large-scale image retrieval with attentive deep local features, in *Proceedings of the IEEE International Conference on Computer Vision* (2017), pp. 3456–3465
19. H. Jégou, M. Douze, C. Schmid, P. Pérez, Aggregating local descriptors into a compact image representation, in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (IEEE, 2010), pp. 3304–3311
20. A. Torii, R. Arandjelovic, J. Sivic, M. Okutomi, T. Pajdla, 24/7 place recognition by view synthesis, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 1808–1817
21. R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, J. Sivic, Netvlad: Cnn architecture for weakly supervised place recognition, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 5297–5307
22. T. Sattler, A. Torii, J. Sivic, M. Pollefeys, H. Taira, M. Okutomi, T. Pajdla, Are large-scale 3d models really necessary for accurate visual localization?, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 1637–1646
23. T. Sattler, Q. Zhou, M. Pollefeys, L. Leal-Taixe, Understanding the limitations of cnn-based absolute camera pose regression, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2019), pp. 3302–3312
24. P.-E. Sarlin, D. DeTone, T. Malisiewicz, A. Rabinovich, Superglue: learning feature matching with graph neural networks, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020), pp. 4938–4947
25. V. Mohanty, S. Agrawal, S. Datta, A. Ghosh, V.D. Sharma, D. Chakravarty, Deepvo: A deep learning approach for monocular visual odometry (2016). arXiv:1611.06069
26. N. Radwan, A. Valada, W. Burgard, Vlocnet++: deep multitask learning for semantic visual localization and odometry. IEEE Robot Autom Lett **3**(4), 4407–4414 (2018)
27. D. Winkelbauer, M. Denninger, R. Triebel, Learning to localize in new environments from synthetic training data (2020). arXiv:2011.04539
28. X. Ding, Y. Wang, L. Tang, Y. Jiao, R. Xiong, Improving the generalization of network based relative pose regression: dimension reduction as a regularizer (2020). arXiv:2010.12796
29. H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 2881–2890
30. X. Bi, Y. Chen, X. Liu, D. Zhang, R. Yan, Z. Chai, H. Zhang, X. Liu, Method towards cvpr 2021 image matching challenge (2021). arXiv:2108.04453
31. J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, R. Shah, Signature verification using a" siamese" time delay neural network, in *Advances in Neural Information Processing Systems*, vol. 6 (1993)
32. S. Woo, J. Park, J.-Y. Lee, I.S. Kweon, Cbam: convolutional block attention module, in *Proceedings of the European Conference on Computer Vision (ECCV)* (2018), pp. 3–19
33. I. Rocco, R. Arandjelovic, J. Sivic, Convolutional neural network architecture for geometric matching, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 6148–6157
34. M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, B. Schiele, The cityscapes dataset for semantic urban scene understanding, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 3213–3223
35. J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: a large-scale hierarchical image database, in *2009 IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2009), pp. 248–255
36. K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, in *Proceedings of the IEEE International Conference on Computer Vision* (2015), pp. 1026–1034
37. D.P. Kingma, J. Ba, Adam: a method for stochastic optimization (2014). arXiv:1412.6980
38. A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, A. Lerer, Automatic differentiation in pytorch (2017)
39. G. Bradski, The opencv library. Dr. Dobb's J. Softw. Tools Profess. Program. **25**(11), 120–123 (2000)
40. X. Yin, L. Ma, P. Sun, X. Tan, A visual fingerprint update algorithm based on crowdsourced localization and deep learning for smart iov. EURASIP J. Adv. Signal Process. **2021**(1), 1–22 (2021)
41. K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 770–778

Publisher's Note