Computer Systems Science & Engineering
DOI: 10.32604/csse.2023.037761
Article

Tech Science Press

An Efficient Indoor Localization Based on Deep Attention Learning Model

Amr Abozeid[1,*], Ahmed I. Taloba[1,2], Rasha M. Abd El-Aziz[1,3], Alhanoof Faiz Alwaghid[1], Mostafa Salem[3] and Ahmed Elhadad[1,4]

[1]Department of Computer Science, College of Science and Arts, Jouf University, Qurayyat, Saudi Arabia
[2]Information System Department, Faculty of Computers and Information, Assiut University, Egypt
[3]Computer Science Department, Faculty of Computers and Information, Assiut University, Assiut, Egypt
[4]Department of Computer Science, Faculty of Computers and Information, South Valley University, Qena, Egypt
*Corresponding Author: Amr Abozeid. Email: aaabozezd@ju.edu.sa

Received: 16 November 2022; Accepted: 06 January 2023

Abstract: Indoor localization methods can help many sectors, such as healthcare centers, smart homes, museums, warehouses, and retail malls, improve their service areas. As a result, it is crucial to look for low-cost methods that can provide exact localization in indoor locations. In this context, image-based localization methods can play an important role in estimating both the position and the orientation of cameras regarding an object. Image-based localization faces many issues, such as image scale and rotation variance. Also, image-based localization's accuracy and speed (latency) are two critical factors. This paper proposes an efficient 6-DoF deep-learning model for image-based localization. This model incorporates the channel attention module and the Scale Pyramid Module (SPM). It not only enhances accuracy but also ensures the model's real-time performance. In complex scenes, a channel attention module is employed to distinguish between the textures of the foregrounds and backgrounds. Our model adapted an SPM, a feature pyramid module for dealing with image scale and rotation variance issues. Furthermore, the proposed model employs two regressions (two fully connected layers), one for position and the other for orientation, which increases outcome accuracy. Experiments on standard indoor and outdoor datasets show that the proposed model has a significantly lower Mean Squared Error (MSE) for both position and orientation. On the indoor 7-Scenes dataset, the MSE for the position is reduced to 0.19 m and 6.25° for the orientation. Furthermore, on the outdoor Cambridge landmarks dataset, the MSE for the position is reduced to 0.63 m and 2.03° for the orientation. According to the findings, the proposed approach is superior and more successful than the baseline methods.

Keywords: Image-based localization; computer vision; deep learning; attention module; VGG-16

1 Introduction

Because of their widespread applications, location-based computing and services have received more attention as the Internet of Things (IoT) has improved. As a result, Information about the target location is critical in localization systems [1,2]. Localization methods are used in developing existing systems using various technologies and methods based on the application. For example, satellite systems with Google Maps that support global coverage, such as Global Positioning System (GPS), have been used to estimate outdoor positioning, tracking, and navigation [3].

Indoor positioning methods can improve service areas provided by healthcare centers, smart homes, museums, warehouses, and shopping malls. As a result, it is appealing to seek a low-cost design capable of providing precise localization in indoor locations. Indoor localization, on the other hand, presents more difficulties than outside localization. Because of the multipath effect, reflecting, fading, deep shadowing effect, and the degradation of delay caused by pervasive hindrances and interactive interference, the pattern of signals in indoor surroundings is more complicated than in outside situations. Therefore, researchers are becoming more interested in indoor-localization methods, which are based on static/mobile cameras, Wi-Fi, Inertial Measurement Units (IMU), and other sensor components [4]. Vision-based localization is growing as cameras become more inexpensive and integrated with smart devices. It is becoming increasingly common in surveillance, navigation, robotics, self-driving, and Augmented Reality (AR) [5,6].

In vision-based methods, the camera Six Degrees of Freedom (6-DoF) poses are evaluated by matching the closest image in a reference database with known ground truth poses. To find matches between images, the global descriptors are searched. A descriptor of an image can be either a handcrafted feature (e.g., Scale Invariant Feature Transform (SIFT) [7,8], Speeded Up Robust Features (SURF) [9], or Oriented FAST and Rotated BRIEF (ORB) [10]) or a learned feature (e.g., SuperPoint [11]). Although feature-based methods are powerful in many situations, it still faces challenges when there are less texture, repetitive structures, and insufficient matching features [12,13].

Deep learning techniques have recently been demonstrated to be effective in solving a variety of computer vision problems [14–17]. Convolutional Neural Networks (CNN) and Fully Convolutional Neural Networks (FCNs) were particularly successful in image segmentation, classification, and recognition. As a result, CNN was used to solve a localization problem. As a result, the localization problem was categorized as a regression problem, like PoseNet [18]. Since then, many improvements have been proposed in terms of incorporating new deep-learning models and architectures.

We present a powerful smart deep-learning model for image-based localization in this paper. Three stages make up the suggested model. The input image is first processed by a truncated VGG-16 [19], which serves as a feature extractor. Next, a channel attention module is employed to draw attention to crucial details while masking distracting ones. A Scale Pyramid Module (SPM) with various dilation rates makes up the next stage. This module records the objects' multiscale information. The final part is the regressor module, consisting of three connective 1×1 convolution layers and two parallel Fully Connected (FC) layers, to regress both location and orientation separately. The result is more accurate when two regressions are used, one for location and the other for orientation. The proposed model is tested on the RGB-D Microsoft 7-Scenes and Cambridge landmarks datasets. The outcomes of these experiments on standard indoor and outdoor datasets demonstrate that the proposed model has a significantly lower MSE for both position and orientation. On the indoor 7-Scenes dataset, the MSE is significantly reduced to 0.19 m for position and 6.25° for orientation. As well, on the Cambridge landmarks dataset, the MSE is significantly reduced to 0.63 m for position and 2.03° for orientation.

We can infer from the findings that the proposed model outperforms the comparable models for image-based localization tasks on both indoor and outdoor datasets.

Our contribution to this paper is to propose a specially designed deep-learning model to handle some image-based localization challenges. The suggested model employs an adapted SPM, which is a feature pyramid module, to address the issues of image scale and rotation variance. The channel attention module is used to collect multiscale features while eliminating unimportant ones. In addition, the proposed model includes two regressions: one for position and one for orientation, which improves result accuracy.

This article is formatted as follows: Section 2 discusses the related work. The proposed model is discussed in Section 3. Section 4 describes the suggested model experiments and outcomes. Finally, the conclusion is presented in Section 5.

2 Related Works

The localization problem refers to the challenge of determining position as well as the orientation of the viewpoint (camera viewpoint). Techniques for localization can be categorized, as shown in Fig. 1, into sensor-based, vision-based, and hybrid. We will give a brief overview of vision-based techniques in this section.

In feature-based techniques, handcrafted features are taken from the original image to find the best localization that matches the stored features. Image descriptors are used to match similar images. Handcrafted (extracted) features (like SIFT [7,8], SURF [9]. ORB [10], etc.) or learned features (such as SuperPoint [11], ASLFeat [20], or SeqNet [21]) can be used as descriptors. Clustering algorithms such as [22,23] can play an essential role in improving the accuracy of feature-based techniques. Although feature-based techniques are efficient and robust, they face difficulties working on images with little or repetitive textures [12,13].

In feature-based methods, the image's handcrafted features are extracted and used to find the best pose that matches the features that have been stored. Deep learning methods directly learn specific representations (encodings) of images at various granularities. The representation for localization problems could be unidentified features, depth, or movement between two images. Typically, the localization problem has been solved using the top three Deep Neural Network (DNN) architectures [24].

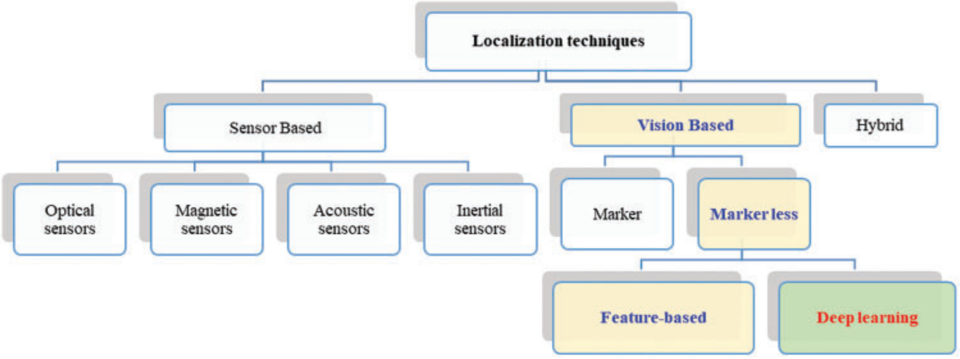


Figure 1: A general classification of localization techniques

As a foundation for building a pose deep learning model, Kendall et al. [18] recommended using the adapted version of the popular DNN design for classification, GoogLeNet [25]. There are a total of 22 layers in the GoogleNet, made up of six inception modules and two intermediate classifiers. Each inception module includes a pile of 3 × 3 filters, 5 × 5 filters, and a pooling layer, all of which are essential for creating robust models.

PoseNet's accuracy is lower than that of some conventional techniques. The regressor layer's input feature vector's (2048D) large size is to blame for this. Furthermore, when utilized with test images that differ from the training images, the overfitting problem worsens as the size of the feature vector increases. Another significant issue left unresolved by the original PoseNet model is the generalization to other datasets.

PoseNet's success in designing the issue as a regression task led to the development of numerous deep-learning models as improvements to the original PoseNet. To alleviate the overfitting issue by reducing the feature vector's size, Walch et al. recommended adding the Long Short-Term Memory (LSTM) modules [26]. Four LSTM units are adopted in this architecture to reduce the feature vector's size further. The LSTM is an example of a Recurrent Neural Network (RNN), which has hidden layers that either gather or omit pertinent contextual features. Recently, some computer vision problems have been solved by combining CNN and LSTM [26].

Another architecture built on the encoder-decoder ResNet34 model is called Hourglass-Pose [27]. The ResNet is made up of four residual modules, each of which has layers for batch normalization, convolution, and activation. Between the opposite decoder and encoder blocks, there are direct (skip) connections. These connections help maintain low-level details that aid in the solution of the vanishing gradient issue.

The VGG-16 model [28] was used in SVS-Pose [29] instead of the GoogleNet model. The SVS-Pose uses a 3 × 3 filter throughout the network. Following the convolutional layers, the VGG-16 has three FC layers. To guess the camera position and the orientation separately, the VGG-16 model was split into two branches after the first FC. Two additional FC layers are added at the end to estimate the position and orientation. BranchNet [30] divides the PoseNet architecture into a shared encoder and a single shared localizer after the fifth inception module. Two branches that assess position and orientation independently are formed from the reset layers by duplicating them.

The MDPoseNet outputs multiple estimated results rather than one estimated pose for each input image. The core concept behind MD-PoseNet is that rather than returning the best camera pose, the network instead outputs the distribution of all possible camera poses [30]. Some models have defined localization as a time-based problem to estimate temporal localization, in contrast to single-image localization. In VidLoc [31], bidirectional recurrent neural networks (BLSTM) are used to localize brief video clips. A network was suggested by VLocNet [32] and VLocNet++ [33] to collectively find both pose and visual odometry. By utilizing Kalman filtering [34], the KFNet method [35] improved the temporal localization.

According to deep learning research, adding more layers makes a model more accurate [36–38]. However, issues such as vanishing/exploding features may arise in the deeper model, which would be detrimental to the training outcomes. An efficient deep learning model called Depth-DensePose was proposed in [36] for 6-DoF camera-based localization. The Depth-DensePose combines the advantages of the DenseNet model and adapted depth-wise separable convolution to create a powerful and deeper model.

3 Methodology

Fig. 2 depicts the proposed model's architecture. There are three stages to it. An adopted VGG-16 [19] followed by a channel attention module is utilized to extract features from a source image. Then, an SPM is used in the middle, followed by 3×3 convolutions using various dilation factors to handle scale variation. The dilations are 2, 4, 8, and 12. Finally, the back end was equipped with two FC layers and 1×1 convolution layers, followed by two parallel FC layers to regress location and orientation independently. The 1×1 convolution layers are used to reduce the feature depth.

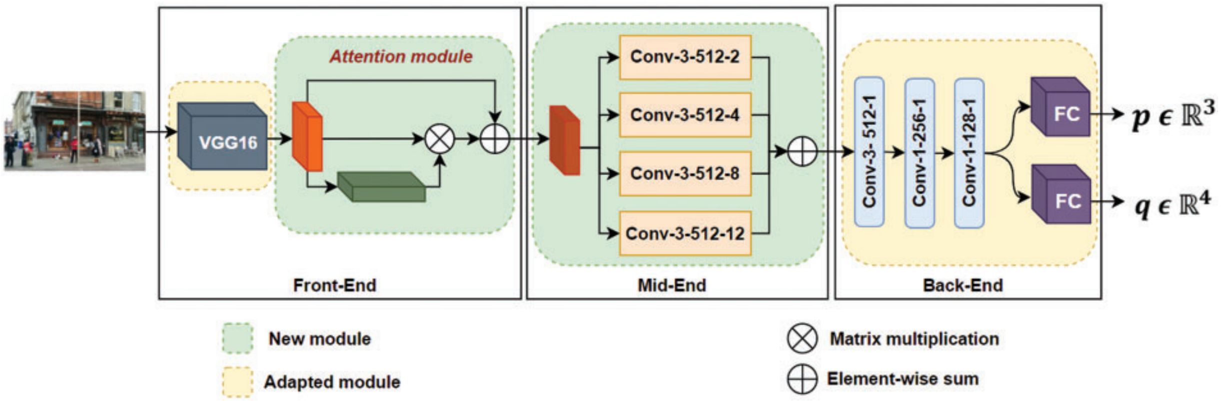


Figure 2: Our model architectural design. Convolutional layer parameters are designated as "Conv-(size of the kernel)-(filters number)-(dilation rate)" in the mid-end module and "Conv-(size of the kernel)-(filters number)-(stride)" in the back-end module

3.1 Front-End: Feature Extraction with Attention Module

We begin by feeding a given localization image of any size into a feature extractor. The feature extractor module comprises the first ten convolutional layers of the pre-trained VGG-16 [19]. Due to the VGG-16's excellent generalization capabilities, it is frequently used as the base for many deep-learning models. As a result, we also adopted a pre-trained VGG-16 on the ImageNet dataset [39] as the base for building the proposed model. Initially, VGG-16 was created and used to solve the image classification problem [39]. Although it has been shown that VGG-16 can be used to accomplish localization tasks [40], other designs should be considered to represent dense scenes more accurately. To achieve the objective of extracting more powerful semantic and contextual features, we incorporate the channel attention module. Recent advancements in the channel attention module [41,42], are the driving force behind this.

Channel Attention: The textures of the foregrounds and backgrounds in the highly-dense scenes are challenging to differentiate. An adopted channel attention module is used to solve this issue. Furthermore, the channel attention module is used to direct attention to important information while obscuring irrelevant ones. Fig. 3 shows the channel attention module architecture. In more detail, given an input feature vector $F \in \mathbb{R}^{C\times H\times\times W}$, C denotes the channel's number, and W and H represent the feature map's width and height. Two feature maps, C_1 and C_2, are produced after the execution of one 1×1convolution layer and subsequent transposing operations. The channel attention map is then created by applying a matrix multiplication and Softmax layer to C_1 and C_2. As in Eq. (1), a channel attention vector C_a with a dimension of C×C is produced [41,42].

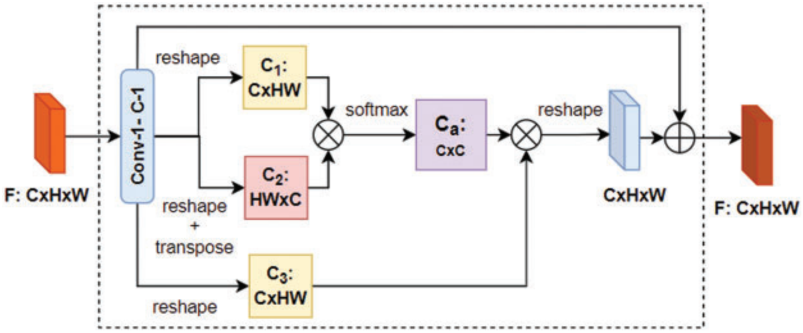


Figure 3: Channel attention module architecture

$$C_a^{ji} = \frac{e^{\left(C_1^i \cdot C_2^j\right)}}{\sum_{i=1}^{C} e^{\left(C_1^i \cdot C_2^j\right)}} \tag{1}$$

where C_a^{ji} denotes the impact of the ith channel on the jth channel. The final feature vector with a dimension of C × H × W is computed as in Eq. (2) [41,42].

$$C_{final}^{j} = \alpha \sum_{i=1}^{C} \left(C_a^{ji} . C_3^{i}\right) + F^{j} \tag{2}$$

where α represents a learned parameter discovered by performing a 1×1 convolution.

3.2 Mid-End: Scale Pyramid Module (SPM)

Multiple max-pooling layers significantly reduce the feature vector size in the front-end stage. The output feature map's size is reduced to 1/64 of its original image size. Two drawbacks will result from this. First, the pooling step renders the features insensitive to local translations, which is beneficial for classification but harmful for image-based localization, making it difficult to produce accurate pose values. Second, the model becomes blind to tiny objects as the feature map's spatial resolution decreases because the information about small objects becomes less valuable.

We use an SPM constructed with four concurrent dilated convolution layers to handle these issues, as inspired by [43]. The dilated convolution operation is a convolution with holes. The idea of extending receptive fields without sacrificing feature map spatial resolution was presented in [44] for a segmentation task. It is an excellent option for this task because it requires no additional parameters or calculations.

The SPM comprises four layers, each of which has the same channels but a different dilation rate to capture features at various scales. As a result, four dilate convolutions with rates of 2, 4, 8, and 12 are used, as suggested in [45]. By doing this, we build a pyramid with various visual fields that can preserve the spatial resolution of feature maps while remaining scale-invariant.

3.3 Back-End: A Regressor Module

The regressor module (back-end) consists of three connective 1 × 1 convolution layers, then two parallel FC layers to regress both locations ($\boldsymbol{p} \in \mathbb{R}^3$) and orientation ($\boldsymbol{q} \in \mathbb{R}^4$). The 1 × 1 convolution is applied to reduce the final feature map to 128 significantly. Moreover, using two fully connected layers as a regression, one for location and the other for orientation, improves the outcomes' accuracy.

4 Results and Discussion

The proposed model implementation, tests, and results are discussed in this section.

4.1 Implementation and Loss Function

PyTorch was used to implement the suggested model [46]. Facebook's AI Research team created the open-source PyTorch library for computer vision and machine learning applications. The NVIDIA RTX 2060 graphics card and device used for implementation and training have 6 GB of memory and 16 GB of RAM.

The first ten pre-trained convolution layers are adopted from the VGG-16 model. The proposed model is trained from the beginning to end. During the training phase, we optimize the model using Stochastic Gradient Descent (SGD) with a learning rate equal to le-5. As in [18], we apply random data augmentation to obtain additional training examples. To prevent running out of memory, we resize all the images to 244 × 244 pixels before doing the data augmentation. Also, the training batch size is set at 32.

With the following objective loss function, we train the model on Euclidean loss using random gradient descent to regress the pose. As in [18], each image pose ($P = [p, q]$) is constructed by the camera position ($\boldsymbol{p} \, \boldsymbol{\epsilon} \, \mathbb{R}^3$) and the orientation ($\boldsymbol{q} \, \boldsymbol{\epsilon} \, \mathbb{R}^4$). Given a training image set Ii with its ground-truth poses Pi. The objective loss function can be formulated as in Eq. (3):

$$L_i = \left\| P_i - \hat{P}_i \right\|_2 + \beta . \left\| q_i - \frac{\hat{q}_i}{\left\| \hat{q}_i \right\|} \right\|_2 \tag{3}$$

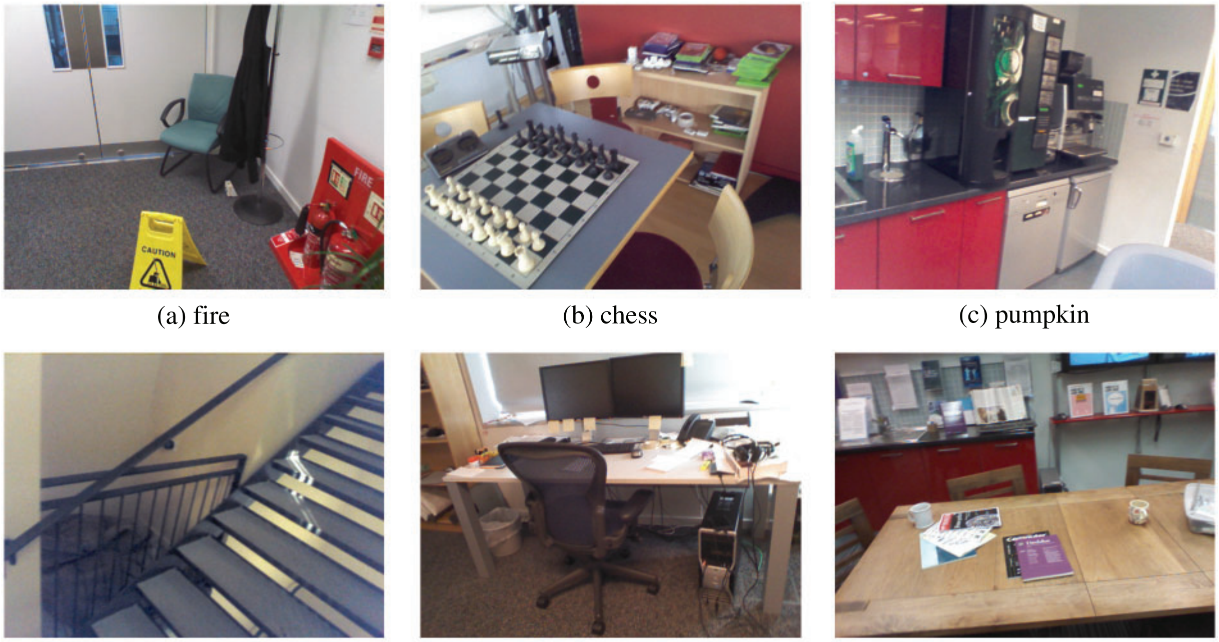
where β denotes a scaling term selected to maintain roughly equal expected values for position and orientation errors. The $(\hat{p}, \hat{q})$ and (p, q) are estimated and ground-truth position-orientation pairs.

4.2 Dataset

Deep learning approaches necessitate a huge amount of training data with ground truth labels, necessitating additional effort and work. Following PoseNet's and its successor models' success, the results mainly were tested and reported on the RGB-D 7-Scenes dataset [47] and the Cambridge dataset [18,48]. The proposed model was evaluated on the Microsoft RGB-D 7-Scenes and the Cambridge landmarks datasets to assess its effectiveness in indoor and outdoor environments.

As shown in Fig. 4, the Microsoft 7-Scenes dataset [47] was created for indoor camera localization and object tracking. It consists of RGB-D images annotated with 6-DoF poses from seven indoor areas. The data was acquired using a Kinect camera with a resolution of 640 × 480 pixels, and ground truth poses were produced using KinectFusion [49]. Each scene comprises between 500 and 1000 frames. Working with this dataset faces challenges because of the texture-free surfaces, motion blur, reflections, and repeating structures.

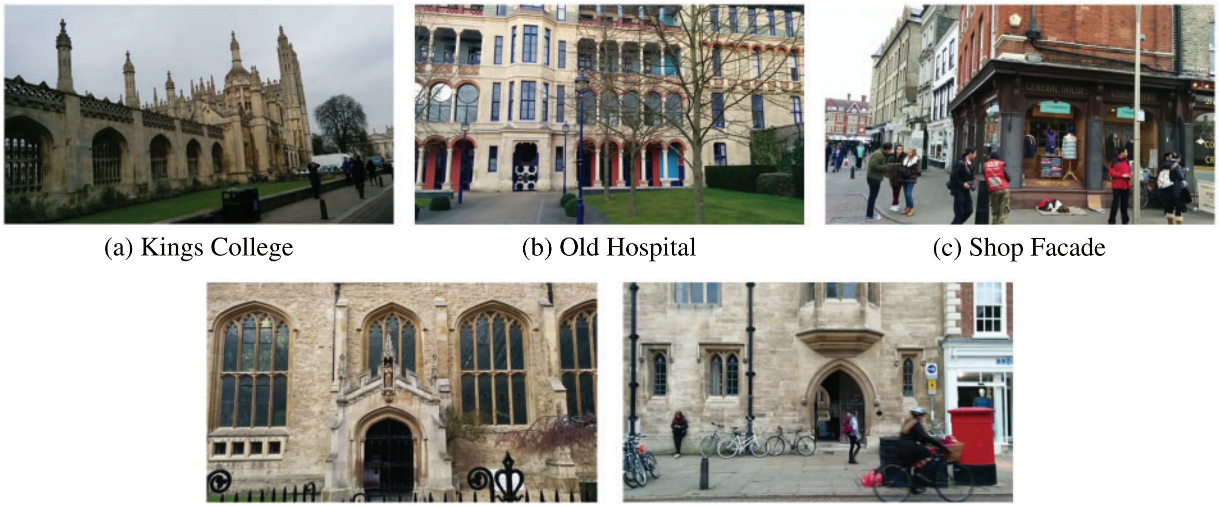
The Cambridge landmarks dataset is a large outdoor localization dataset containing six scenes focused on Cambridge University (see Fig. 5). For each scene, a set of images for training and testing is prepared. The approximately 12,000 images in the Cambridge dataset were created using a phone camera, and a full 6-DOF camera pose has been labeled. Each image has a resolution of 1920 × 1080 pixels. The Visual SfM (Structure from Motion) generates the dataset labels [50]. The datasets were subjected to various random scaling and transformations, resulting in a decrease in complexity and an increase in dataset size. Each input image was resized to 256 × 341 pixels and cropped randomly to 224 × 224 pixels.



(a) fire (b) chess (c) pumpkin

(d) stairs (e) office (f) red kitchen

Figure 4: Samples from the RGB-D 7-Scenes indoor dataset [47]



(a) Kings College (b) Old Hospital (c) Shop Facade

(d) St. Mary’s Church (e) Street

Figure 5: Samples from the Cambridge outdoor dataset [18,48]

4.3 Results on Indoor and Outdoor Datasets

In this section, we evaluate our model on the RGB-D Microsoft 7-Scenes and the Cambridge landmarks datasets. In these experiments, the Mean Squared Error (MSE) metric was used for both position (in meters) and rotation (in degrees). A general formulation of the MSE is shown in Eq. (4).

Where n represents the number of testing images. The $\hat{Y}$ and Y represent the estimated and the ground-truth values (position or orientation), respectively.

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2 \tag{4}$$

The proposed model was evaluated and compared to related works, such as PoseNet [18], LSTM-Pose [26], DensePoseNet [51], SVS-Pose [29], VLocNet [32], and Depth-DensePose [36]. Table 1 shows our model's quantitative findings and comparisons with similar state-of-the-art methods on the 7-Scenes dataset. We train our model across all the 7-scenes. For each scene, our model significantly outperforms the related approaches. Furthermore, the findings suggest that our method performs effectively in various settings.

Table 1: Comparing the proposed model (MSE) to other related models on the RGB-D 7-Scenes dataset

	Chess		Fire		Office		Heads		Pumpkin		Kitchen		Stairs		Average	
	P	R	P	R	P	R	P	R	P	R	P	R	P	R	P	R
PoseNet	0.32	8.12	0.47	14.4	0.48	7.68	0.29	12	0.47	8.42	0.59	8.64	0.47	13.8	**0.44**	**10.44**
SVS-PoseNet	0.37	7.24	0.43	13.7	8.04	0.3	0.31	12	0.61	7.08	0.58	7.54	0.48	13.1	**0.47**	**9.81**
BranchNet	0.18	5.17	0.34	8.99	0.3	7.05	0.2	14.15	0.27	5.1	0.33	7.4	0.38	10.26	**0.29**	**8.3**
LSTM-Pose	0.24	5.77	0.34	11.9	0.3	8.08	0.21	13.7	0.33	7	0.37	8.83	0.4	13.7	**0.31**	**9.85**
Hourglass-Pose	0.15	6.53	0.27	10.84	0.21	8.48	0.19	11.63	0.25	7.01	0.27	10.84	0.29	12.46	**0.23**	**9.68**
Our model	**0.13**	**4.36**	**0.22**	**8.04**	**0.15**	**0.23**	**0.16**	**10.54**	**0.22**	**4.6**	**0.25**	**6.6**	**0.21**	**9.36**	**0.19**	**6.25**

Specifically, compared to PoseNet [18], the baseline model, our model significantly improves localization prediction performance, lowering MSE for position from 0.32 m to 0.13 m and orientation from 8.12° to 4.36°. Fig. 6 depicts the MSE distributions of position and orientation errors across scenes, demonstrating that our proposed model outperforms the others in terms of overall accuracy. Furthermore, the proposed model can provide high-quality results even when tested on a challenging dataset.

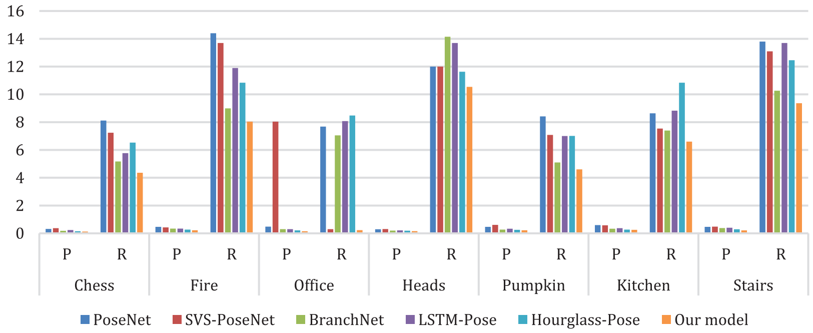


Figure 6: Comparison of the proposed model to related methods on RGB-D 7-Scenes

In Table 2, we compare our experimental results with the results of related models on the outdoor Cambridge dataset. These experiments have several conclusions that can be drawn. First off, our model is better than the related work and achieves a lower MSE. In general, our model reduces the overall average MSE for both position and orientation to 0.63 m and 2.03o, respectively. Secondly, the proposed model outperforms VLocNet [32], which was unable to handle a sizable dataset. The outcomes in Fig. 7 demonstrate the value of the suggested architecture in resolving the image-based localization issue.

Table 2: Comparing the proposed model (MSE) to other related models on the dataset of Cambridge dataset

	K. College		Shop Façade		Old hospital		St M. Church		Street		Average	
	P	R	P	R	P	R	P	R	P	R	P	R
PoseNet	1.92	5.40	1.46	8.08	2.31	5.38	2.65	8.48	3.67	6.50	**2.40**	**6.77**
LSTM-Pose	0.99	3.65	1.18	7.44	1.51	4.29	1.52	6.68	NA	NA	**1.3**	**5.52**
DensePoseNet	1.66	4.86	1.41	7.18	2.57	5.14	2.45	7.96	2.96	6.00	**2.21**	**6.23**
VLocNet	0.84	1.42	0.59	3.53	1.07	2.41	0.63	3.91	NA	NA	**0.78**	**2.82**
SVS-Pose	1.06	2.81	0.63	5.73	1.50	4.03	2.11	8.11	NA	NA	**1.33**	**5.17**
Depth-DensePose	0.69	1.23	0.38	3.04	0.60	0.82	0.78	2.26	1.24	3.34	**0.74**	**2.12**
Our model	**0.58**	**1.13**	**0.29**	**2.87**	**0.53**	**0.75**	**0.68**	**2.2**	**1.11**	**3.21**	**0.63**	**2.03**

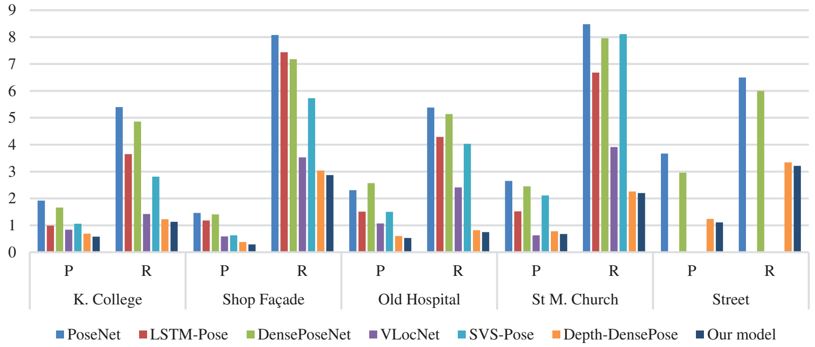


Figure 7: Comparison of our model to other related methods on Cambridge landmarks

Based on the findings of both indoor and outdoor datasets and comparisons with recent related work, we can conclude that our proposed model achieves the lowest location MSE and the lowest orientation MSE in most outdoor and indoor scenes.

4.4 Ablation Study

In this subsection, we perform ablation experiments on four scenes from the 7-Scenes dataset using simplified models to evaluate better the role played by various modules in the proposed model.

1. Baseline: It is made up of the first ten VGG-16 convolution layers and the regressor module.
2. Baseline+Attention: A channel attention module is inserted after the truncated VGG-16 and before the regressor module.
3. Baseline+Attention+SPM: The proposed Model.

Table 3 summarizes the ablation experiment results. The findings show that each component of our model helps to increase accuracy. In particular, the simple baseline model does not provide the best results. The channel attention model has a positive effect on the results because it is used to direct attention to important information while obscuring irrelevant ones. The utilization of SPM enhances accuracy even more by capturing multiscale features. It is evident that by adding both the channel attention module and the SPM into the combined model, the suggested model can achieve higher localization performance and more accurate prediction results.

Table 3: Ablation experiments on four scenes from the 7-Scenes dataset

Methods	Chess		Fire		Office		Heads	
	P	R	P	R	P	R	P	R
Baseline	0.28	7.47	0.42	12.14	0.341	3.87	0.212	11.341
Baseline+Attention	0.14	5.13	0.25	9.78	0.18	0.89	0.17	10.79
Baseline+Attention+SPM	**0.13**	**4.36**	**0.22**	**8.04**	**0.15**	**0.23**	**0.16**	**10.54**

4.5 Efficiency Evaluation

Table 4 allows for numerous conclusions to be formed. Firstly, the proposed model depends on a significantly lower number of parameters than the VGG-16. However, compared to PoseNet, the proposed model relies on more parameters. Secondly, the suggested model consumes 92.53% less memory than the VGG-16 and 35.16% more memory than the PoseNet. Despite having more parameters than PoseNet, it has much-improved accuracy.

Table 4: Evaluation of total parameter number and size

	VGG-16	PoseNet	**Our model**
Total parameters	134,268,738	12,431,173	**19,172,178**
Trainable parameters	134,268,738	12,431,173	**19,172,178**
Parameters size (MB)	512.19	47.42	**73.14**

Empirical studies show that deep-learning performance improves as the number of parameters increases [52]. A low-complexity (fewer parameters) model may be quicker to train, but it may only capture some of the useful information in the data. On the other hand, a complex model can capture more features from the data. However, it will be more challenging to train and may be sensitive to overfitting [53]. Therefore, the proper balance between accuracy and complexity is crucial for a successful deep-learning model. This issue is critical for real-time and mobile applications with limited memory.

5 Conclusion

This paper proposes an efficient image-based localization deep learning model. This model consists of three stages. First, an adapted VGG-16 was used to feature extraction from a source image. Then, a channel attention module is used to draw attention to crucial details and hide distracting ones. Second, the mid-end stage consists of an SPM with diverse dilation rates that store multiscale object features. Third, there are two FC layers in the stage of the regressor. This increases the outcome's accuracy by using one for location and the other for orientation. Two standard indoor/outdoor datasets, the Microsoft 7-Scenes and the Cambridge landmarks were utilized to examine our model. These experiments' findings indicate that the suggested model has a lower MSE for location and orientation. The MSE for location on the 7-Scenes dataset is dramatically decreased to 0.19 m and

6.25° for orientation. Furthermore, using the Cambridge landmarks dataset, the MSE is lowered to 0.63 m for location and 2.03° for orientation. The outcomes indicate that the suggested model outperforms related models on both indoor and outdoor datasets for image-based localization tasks. Additionally, ablation studies were carried out further to evaluate the efficiency of each component of our methodology.

However, there are some disadvantages, such as the generalizability to other datasets and the suboptimal performance on less-texture senses of the proposed model. As a result, we intend to conduct additional experiments on diverse datasets to enhance our model to address these issues.

Acknowledgement: This work was funded by the Deanship of Scientific Research at Jouf University under grant No (DSR-2021-02-0379).

Funding Statement: This work was funded by the Deanship of Scientific Research at Jouf University under grant No (DSR-2021-02-0379).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] J. Wichmann, "Indoor positioning systems in hospitals: A scoping review," *Digital Health*, vol. 8, pp. 1–20, 2022.

[2] K. A. Phung, C. Kirbas, L. Dereci and T. V. Nguyen, "Pervasive healthcare internet of things: A survey," *Information-An International Interdisciplinary Journal*, vol. 13, no. 8, pp. 1–17, 2022.

[3] K. G. Tan, K. Z. Aung, M. S. Aung, M. T. Soe, A. Abdaziz *et al.,* "Review of indoor positioning: Radio wave technology," *Applied Sciences*, vol. 11, no. 1, pp. 1–44, 2021.

[4] G. Zhou, S. Xu, S. Zhang, Y. Wang and C. Xiang, "Multi-floor indoor localization based on multi-modal sensors," *Sensors*, vol. 22, no. 11, pp. 1–19, 2022.

[5] M. Shu, G. Chen and Z. Zhang, "Efficient image-based indoor localization with MEMS aid on the mobile device," *ISPRS Journal of Photogrammetry Remote Sensing*, vol. 185, no. 2, pp. 85–110, 2022.

[6] F. S. Daniş, A. T. Naskali, A. T. Cemgil and C. Ersoy, "An indoor localization dataset and data collection framework with high precision position annotation," *Pervasive Mobile Computing*, vol. 81, no. 11, pp. 101554, 2022.

[7] J. Wu, Z. Cui, V. S. Sheng, P. Zhao, D. Su *et al.,* "A comparative study of SIFT and its variants," *Measurement Science Review*, vol. 13, no. 3, pp. 122–131, 2013.

[8] K. ELDahshan, H. Farouk, A. Abozeid and M. H. Eissa, "Global dominant SIFT for video indexing and retrieval," *Journal of Theoretical Applied Information Technology*, vol. 97, no. 19, pp. 5023–5034, 2019.

[9] H. Bay, A. Ess, T. Tuytelaars and L. Van Gool, "Speeded-up robust features (SURF)," *Computer Vision Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[10] M. Bansal, M. Kumar and M. Kumar, "2D object recognition: A comparative analysis of SIFT, SURF and ORB feature descriptors," *Multimedia Tools Applications*, vol. 80, no. 12, pp. 18839–18857, 2021.

[11] Z. Li, J. Cao, Q. Hao, X. Zhao, Y. Ning *et al.,* "DAN-superPoint: Self-supervised feature point detection algorithm with dual attention network," *Sensors*, vol. 22, no. 5, pp. 1940, 2022.

[12] L. Wang, R. Li, J. Sun, H. S. Seah, C. K. Quah *et al.,* "Feature-based and convolutional neural network fusion method for visual relocalization," in *Proc. 15th Int. Conf. on Control, Automation, Robotics and Vision (ICARCV)*, Singapore, pp. 1489–1495, 2018.

[13] P. Roy and C. Chowdhury, "A survey of machine learning techniques for indoor localization and navigation systems," *Journal of Intelligent Robotic Systems*, vol. 101, no. 3, pp. 1–34, 2021.

[14] T. Kattenborn, J. Leitloff, F. Schiefer and S. Hinz, “Review on convolutional neural networks (CNN) in vegetation remote sensing,” *ISPRS Journal of Photogrammetry Remote Sensing*, vol. 173, no. 2, pp. 24–49, 2021.

[15] D. Sarvamangala and R. V. Kulkarni, “Convolutional neural networks in medical image understanding: A survey,” *Journal of Evolutionary Intelligence*, vol. 15, pp. 1–22, 2021.

[16] A. Dhillon and G. K. Verma, “Convolutional neural network: A review of models, methodologies and applications to object detection,” *Progress in Artificial Intelligence Journal*, vol. 9, no. 2, pp. 85–112, 2020.

[17] Z. Li, F. Liu, W. Yang, S. Peng and J. Zhou, “A survey of convolutional neural networks: Analysis, applications, and prospects,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 12, pp. 6999–7019, 2022.

[18] A. Kendall, M. Grimes and R. Cipolla, “PoseNet: A convolutional network for real-time 6-dof camera relocalization,” in *Proc. IEEE Int. Conf. on Computer Vision*, Santiago, Chile, pp. 2938–2946, 2015.

[19] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” arXiv preprint arXiv:1409.1556, 2014.

[20] Z. Luo, L. Zhou, X. Bai, H. Chen, J. Zhang *et al.,* “ASLFeat: Learning local features of accurate shape and localization,” in *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, pp. 6588–6597, 2020.

[21] S. Garg and M. Milford, “SeqNet: Learning descriptors for sequence-based hierarchical place recognition,” *IEEE Robotics Automation Letters*, vol. 6, no. 3, pp. 4305–4312, 2021.

[22] Y. Tang, Z. Pan, W. Pedrycz, F. Ren and X. Song, “Viewpoint-based kernel fuzzy clustering with weight information granules,” *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 6, no. 6, pp. 1–15, 2022.

[23] Y. Tang, L. Zhang, G. Bao, F. Ren and W. Pedrycz, “Symmetric implicational algorithm derived from intuitionistic fuzzy entropy,” *Iranian Journal of Fuzzy Systems*, vol. 19, no. 4, pp. 27–44, 2022.

[24] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan *et al.,* “Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions,” *Journal of Big Data*, vol. 8, no. 1, pp. 1–74, 2021.

[25] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed *et al.,* “Going deeper with convolutions,” in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Boston, MA, USA, pp. 1–9, 2015.

[26] F. Walch, C. Hazirbas, L. Leal-Taixe, T. Sattler, S. Hilsenbeck *et al.,* “Image-based localization using lstms for structured feature correlation,” in *Proc. IEEE Int. Conf. on Computer Vision*, Venice, Italy, pp. 627–637, 2017.

[27] I. Melekhov, J. Ylioinas, J. Kannala and E. Rahtu, “Image-based localization using hourglass networks,” in *Proc. IEEE Int. Conf. on Computer Vision Workshops*, Venice, Italy, pp. 879–886, 2017.

[28] X. Zhang, J. Zou, K. He and J. Sun, “Accelerating very deep convolutional networks for classification and detection,” *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 38, no. 10, pp. 1943–1955, 2015.

[29] T. Naseer and W. Burgard, “Deep regression for monocular camera-based 6-dof global localization in outdoor environments,” in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, Vancouver, Canada, pp. 1525–1530, 2017.

[30] J. Wu, L. Ma and X. Hu, “Delving deeper into convolutional neural networks for camera relocalization,” in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, Singapore, pp. 5644–5651, 2017.

[31] R. Clark, S. Wang, A. Markham, N. Trigoni and H. Wen, “Vidloc: A deep spatio-temporal model for 6-dof video-clip relocalization,” in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, pp. 6856–6864, 2017.

[32] A. Valada, N. Radwan and W. Burgard, “Deep auxiliary learning for visual localization and odometry,” in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, Brisbane, Australia, pp. 6939–6946, 2018.

[33] N. Radwan, A. Valada, W. Burgard and A. Letters, “Vlocnet++: Deep multitask learning for semantic visual localization and odometry,” *IEEE Robotics*, vol. 3, no. 4, pp. 4407–4414, 2018.

[34] R. J. Meinhold and N. D. Singpurwalla, “Understanding the kalman filter,” *The American Statistician*, vol. 37, no. 2, pp. 123–127, 1983.

[35] L. Zhou, Z. Luo, T. Shen, J. Zhang, M. Zhen *et al.,* “KFNet: Learning temporal camera relocalization using kalman filtering,” in *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, pp. 4919–4928, 2020.

[36] A. Abozeid, H. Farouk and S. Mashali, “Depth-DensePose: An efficient densely connected deep learning model for camera-based localization,” *International Journal of Electrical Computer Engineering*, vol. 12, no. 3, pp. 2792–2801, 2022.

[37] N. Alalwan, A. Abozeid, A. A. ElHabshy and A. Alzahrani, “Efficient 3D deep learning model for medical image semantic segmentation,” *Alexandria Engineering Journal*, vol. 60, no. 1, pp. 1231–1239, 2021.

[38] J. Dolz, K. Gopinath, J. Yuan, H. Lombaert, C. Desrosiers *et al.,* “HyperDense-Net: A hyper-densely connected CNN for multi-modal image segmentation,” *IEEE Transactions on Medical Imaging*, vol. 38, no. 5, pp. 1116–1126, 2018.

[39] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh *et al.,* “Imagenet large scale visual recognition challenge,” *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[40] I. Ha, H. Kim, S. Park and H. Kim, “Image-based indoor localization using BIM and features of CNN,” in *Proc. Int. Symp. on Automation and Robotics in Construction (ISARC)*, Berlin, Germany, pp. 1–4, 2018.

[41] S. Woo, J. Park, J. -Y. Lee and I. S. Kweon, “Cbam: Convolutional block attention module,” in *Proc. the European Conf. on Computer Vision (ECCV)*, Munich, Germany, pp. 3–19, 2018.

[42] J. Gao, Q. Wang and Y. Yuan, “SCAR: Spatial-/channel-wise attention regression networks for crowd counting,” *Neurocomputing*, vol. 363, no. 3, pp. 1–8, 2019.

[43] L. Liang, H. Zhao, F. Zhou, Q. Zhang, Z. Song *et al.,* “Sc2net: Scale-aware crowd counting network with pyramid dilated convolution,” *Applied Intelligence*, vol. 52, no. 8, pp. 12091–12102, 2022.

[44] F. Yu and V. Koltun, “Multi-scale context aggregation by dilated convolutions,” arXiv preprint arXiv:07122, 2015.

[45] X. Chen, Y. Bin, N. Sang and C. Gao, “Scale pyramid network for crowd counting,” in *Proc. IEEE Winter Conf. on Applications of Computer Vision (WACV)*, Waikoloa Village, HI, USA, pp. 1941–1950, 2019.

[46] Pytorch, “An open source machine learning framework,” 2022. [Online]. Available: https://pytorch.org/.

[47] B. Glocker, S. Izadi, J. Shotton and A. Criminisi, “Real-time RGB-D camera relocalization,” in *Proc. IEEE Int. Symp. on Mixed and Augmented Reality (ISMAR)*, Adelaide, Australia, pp. 173–179, 2013.

[48] A. Kendall, M. Grimes and R. Cipolla, “Cambridge landmarks dataset,” 2015. [Online]. Available: http://mi.eng.cam.ac.uk/projects/relocalisation/.

[49] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim *et al.,* “KinectFusion: Real-time dense surface mapping and tracking,” in *Proc. 10th IEEE Int. Symp. on Mixed and Augmented Reality*, United States, pp. 127–136, 2011.

[50] C. Wu, “VisualSFM: A visual structure from motion system,” 2011. [Online]. Available: http://ccwu.me/vsfm/.

[51] A. Elmoogy, X. Dong, T. Lu, R. Westendorp and K. Reddy, “Linear-PoseNet: A real-time camera pose estimation system using linear regression and principal component analysis,” in *Proc. IEEE 92nd Vehicular Technology Conf. (VTC2020-Fall)*, Virtual Conference, pp. 1–6, 2020.

[52] P. Enkvetchakul and O. Surinta, “Effective data augmentation and training techniques for improving deep learning in plant leaf disease recognition,” *Applied Science Engineering Progress*, vol. 15, no. 3, pp. 3810, 2022.

[53] M. M. Bejani and M. Ghatee, “A systematic review on overfitting control in shallow and deep neural networks,” *Artificial Intelligence Review*, vol. 54, no. 8, pp. 6391–6438, 2021.