A Robust CNN Training Approach to Address Hierarchical Localization with Omnidirectional Images

Juan José Cabrera, Sergio Cebollada[a], Luis Payá[b], María Flores[c] and Oscar Reinoso[d]
Department of Systems Engineering and Automation, Miguel Hernández University, Elche, Spain

Keywords: Hierarchical Localization, Omnidirectional Imaging, Deep Learning, Bayesian Optimization.

Abstract: This paper reports and evaluates the training optimization process of a Convolutional Neural Network (CNN) with the aim of addressing the localization of a mobile robot. The proposed method addresses the localization problem by means of a hierarchical approach by using a visual sensor that provides omnidirectional images. In this sense, AlexNet is adapted and re-trained with a twofold purpose. First, the rough localization step consists of a room retrieval task. Second, the fine localization step within the retrieved room is carried out by means of a nearest neighbour search by comparing a holistic descriptor obtained from the CNN with the visual model of the retrieved room. The novelty of the present work lies in the use of a CNN and holistic descriptors obtained from raw omnidirectional images as captured by the vision system, with no panoramic conversion. In addition, this work proposes the use of a data augmentation technique and a Bayesian optimization to address the training process of the CNN. These approaches constitute an efficient and robust solution to the localization problem, as shown in the experimental section even in presence of substantial changes of the lighting conditions.

1 INTRODUCTION

In recent years, the use of omnidirectional cameras along with computer vision techniques has proven to be a robust solution to address the task of localization in mobile autonomous robotics. This kind of images provide a lot of information with a 360 degree field of view around it and the cost of the camera is relatively low compared to other types of sensors. In addition, holistic (or global-appearance) description approaches have been also proven to present a successful solution for extracting the most relevant information from images, as they lead to more direct location algorithms based on a pairwise comparison between descriptors.

As for the mapping task, using hierarchical models with holistic descriptors allows solving the localization task efficiently. This method involves sorting the visual information hierarchically into different layers of information in such a way that the localization can be resolved in two main steps. First, a rough localization to know an area of the environment, and second, a fine localization, which is tackled in that pre-selected area.

Furthermore, using artificial intelligence (AI) techniques to carry out computer vision and robotics problems has emerged during the past few years. This is due to the fact that faster and more efficient hardware devices are available with a relative low cost. Among AI techniques, convolutional neural networks (CNN) are a very popular technique for tackling a variety of problems in mobile robotics. In the light of the training process, this should be robust and varied. This process plays an important role in the success of the desired tasks. Hence, two topics should be especially considered: (1) a large set of training data must be available and (2) training parameters must be cautiously selected.

In light of the above information, the aim of this work is to introduce and evaluate the performance of a CNN training which is used to address the mapping and localization tasks by using convolutional neural networks. The efficiency of these techniques will be assessed through the success for room retrieval and the ability of the proposed approach to robustly estimate the robot's position using the information stored on the map. This evaluation is done by using only the set of images obtained by an omnidirectional vision

[a] https://orcid.org/0000-0003-4047-3841
[b] https://orcid.org/0000-0002-3045-4316
[c] https://orcid.org/0000-0003-1117-0868
[d] https://orcid.org/0000-0002-1065-8944

Cabrera, J., Cebollada, S., Payá, L., Flores, M. and Reinoso, O.
A Robust CNN Training Approach to Address Hierarchical Localization with Omnidirectional Images.
DOI: 10.5220/0010574603010310
In *Proceedings of the 18th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2021)*, pages 301-310
ISBN: 978-989-758-522-7

sensor installed on the mobile robot, which moves in an indoor environment in real operating conditions.

The novelty of the present work is a localization approach based on a CNN which departs from omnidirectional images. Also, this work presents a CNN training optimization process to carry out efficiently the training task. In general, the objective of this work is to re-adapt and use a CNN with a dual purpose: (1) retrieving in which room the robot is currently located (rough localization step) and (2) refining this localization within the recovered room (fine localization step) by means of global-appearance descriptors obtained from intermediate layers of the CNN itself. Our main contributions in this work can be summarized as follows.

- We adapt and train a CNN with the aim of retrieving the room where an omnidirectional image was captured.
- We propose a training process based on a data augmentation approach and a training optimization.
- We study the use of the proposed deep learning technique to address localization in a hierarchical way.

The remainder of the paper is structured as follows. Section 2 presents a review of the related literature. After that, section 3 presents the methods to train the adapted CNN. Section 4 explains the proposed localization method based on the adapted CNN and presents all the experiments which were tackled to test the validity of the proposed methods. Finally, section 5 presents the conclusions and future works.

2 STATE OF THE ART

A variety of problems in computer vision and robotics have been recently solved by using machine learning techniques (Cebollada et al., 2021). For instance, Dymczyk et al. (2018) propose to tackle the localization task by using a classifier that classifies landmark observations. Regarding deep learning, this subfield of machine learning has gained much interest due to the improvements in processing systems. This technique has gained popularity in the field of robotics in the last few years. For example, Shvets et al. (2018) use segmentation to distinguish between different surgical instruments. Levine et al. (2018) propose a convolutional neural network for robotic grasping from monocular images by learning a hand-eye coordination. Regarding the use of CNNs in the field of mobile robotics, many works have proven success by using this tool. For instance, Sinha et al. (2018) propose a robot re-localization in GPS-denied environments by using a CNN to process data from a monocular camera. Chaves et al. (2019) use a CNN to detect objects in images and use this to build a semantic map.

Regarding the use of the visual information, in line with previous works (Cebollada et al., 2019), the present work focuses on addressing the mapping and localization tasks by means of obtaining a unique descriptor per image which contains global information about it. A wide range of works has been proposed in mobile robotics by using holistic descriptors. Originally, global-appearance description is based on handcrafted methods, that is, they depart from an image and tackle some mathematical transformations to obtain a single vector ($\vec{d} \in \mathbb{R}^{l \times 1}$) with characteristic information from the image.

Nevertheless, recently proposed works have proposed the use of holistic descriptors that are calculated by using intermediate layers of CNNs. In this regard, hidden layers provide descriptors that can be used to characterize input data. To cite some examples, Arroyo et al. (2016) use a CNN that learns to generate descriptors that are robust against station changes, hence, they can be used to perform a long-term localization task. More recently, Wozniak et al. (2018) propose the use of feature extraction from a Support Vector Machines (SVM) classifier. Cebollada et al. (2020) show the advantages of using descriptors obtained from the intermediate layers of a retrained CNN to solve the localization as a batch image retrieval problem. Nevertheless, this work proposes a CNN based on panoramic images. Hence, in order to work with omnidirectional images, a previous transformation to panoramic must be carried out.

As for the training process, deep learning tools require a large dataset. This is essential to obtain robust-enough performances. However, in some cases, the available dataset for training is small and, then, the deep model cannot be trained properly. Among the proposed techniques to address this problem, the present work focuses on data augmentation and an optimization of the training hyperparameters. Concerning the data augmentation technique, this improves the training performance of the model by increasing the number of training instances and avoiding over-fitting. Data augmentation basically consists of creating new data (in this case, images) by applying different effects on the original ones. Some authors have used data augmentation to improve their deep learning tasks. For example, Ding et al. (2016) use three data augmentation methods to tackle a Synthetic Aperture Radar target recognition. The aim of this work is to make the network robust against target translation, speckle variation in different observations, and pose missing. Salamon and Bello (2017) propose audio data augmentation for overcoming the problem of environmental sound data scarcity. By using this technique, they are able to develop a CNN which is able to classify these data. Nevertheless, none of the previously proposed data augmentation methods match the visual effects that can occur when the robot moves through the target environment under real-operation conditions.

Regarding the optimization of training hypeparameters, in the machine learning field, the hyperparameters are those values whose configuration is external and are not learned from the data. For example, whereas the weights of the CNNs are parameters (they are learned during the training process), the learning rate is a hyperparameters (it is established by the CNN architect and it is not learned during the training process). Hence, hyperparameters are used in processes to help to estimate the model parameters. This kind of parameters are usually specified by the practitioner and tuned according to a given predictive modelling problem. The designer of the network cannot know in advance the best hyperparameter values on a given problem. Therefore, they may use rules of thumb, copy values used on other problems, or search for the best value by trial and error.

The correct training of many machine learning methods depends to a large extent on hyperparameters settings and thus on the method used to set hyperparameters. Optimization methods such as grid search and random search have been shown to outperform established methods for this problem (Bergstra and Bengio, 2012). These methods have been capable of obtaining similar or better hyperparameter settings than the established by human domain experts (Bergstra et al., 2013; Kotthoff et al., 2019). As a result, hyperparameter optimization has become an active research area (Bergstra and Bengio, 2012; Falkner et al., 2018; Feurer and Hutter, 2019). During the past few years, Bayesian optimization has emerged as an efficient framework, achieving impressive successes (Snoek et al., 2015; Domhan et al., 2015). Through the Bayesian optimization, the loss minimization is seen as a “black-box” problem, with the aim of finding $argmin_{x \in X}(f(x))$, where $x \in X$ are the hyperparameters and f(x) is the loss function of the model.

Concerning the localization task from a hierarchical point of view, previous works (Cebollada et al., 2019; Payá et al., 2018) have demonstrated that using these models with holistic descriptors and omnidirectional images leads to an efficient and robust solution to tackle the localization task. These previous works consist basically of calculating the nearest neighbour in two layers. First, for the high-level layer, the visual descriptors are grouped according to their similitude and a representative descriptor $R = \{\vec{r}_1, \vec{r}_2, ..., \vec{r}_{n_g}\}$ is obtained for each group, where n_g is the number of groups. Afterward, in order to solve the localization task, a new image is obtained im_{test} and its holistic descriptor is calculated $\vec{d}_{test}$. This descriptor is compared with all the representatives R and the most similar representative $\vec{r}_k$ is retained (rough localization step); after that, a new comparison is carried out between $\vec{d}_{test}$ and the descriptors contained in the group k, $D_k = \{\vec{d}_{k,1}, \vec{d}_{k,2}, ..., \vec{d}_{k,N_k}\}$. Finally, the position of the image im_{test} is estimated as the position where the most similar image in the k-th group was captured (fine localization step).

Hence, this work proposes addressing a hierarchical localization task by using a CNN, which has been trained with omnidirectional images, to obtain a model with the aim of: (a) retrieving the room where the image was captured and (b) obtaining global-appearance descriptors departing from the re-trained network and solving the fine localization by means of an image retrieval problem. With this aim, a pre-trained CNN architecture is re-adapted and re-trained. Additionally, with the aim of improving the training process, this work presents a data augmentation technique and a Bayesian optimization of the main hyperparameters. The aim of this work is to provide a feasible solution that simplifies the CNN development and also solves efficiently the localization task concerning localization error.

3 TRAINING PROCESS

The idea of the present work is to build a deep learning tool that, apart from retrieving the room where the image was captured, is also capable of providing holistic description information that characterizes the image better than hand-crafted methods. As for the CNN to tackle a classification task, the idea basically consists of training the network with visual data and the corresponding labelling for each image from the training dataset. Once the CNN is properly trained, it will be able to solve the rough localization step (i.e., the room retrieval). Moreover, this work proposes to use the layers of the re-trained CNN to obtain holistic descriptors and to use those descriptors to estimate the position within a room where an omnidirectional image was captured.

3.1 CNN Adaption

Due to the fact that building and training a network from scratch requires experience with network architectures, a huge amount of data for training, and, hence, a significant computing time. This work continues the proposal carried out in previous works (Cebollada et al., 2020): adapting and training networks. This work proposes departing from the AlexNet (Krizhevsky et al., 2012), since it presents a basic architecture and has been successfully used in previous works to develop new classification tasks by means of transfer learning (such as Han et al. (2018)). Also, it presented successful results by means of architecture re-adaption and retraining (Cebollada et al., 2020). Unlike these previous works, which were based on conventional (non-panoramic and panoramic) images, the aim of the present work is to study the feasibility of this architecture by departing from omnidirectional images. This proposal presents a twofold benefit: (1) saving computing time, since it is not necessary a transformation from omnidirectional to panoramic images and (2) obtaining holistic descriptors based on omnidirectional images, which have been scarcely proposed in the current state of the art. Moreover, the present work also develops a robust hyperparameters optimization with the aim of addressing an optimal training of the deep learning model.

Therefore, first, some layers of the AlexNet architecture are modified to adapt the network to the proposed room retrieval task. In this case, the layers fully-connected layer (fc_8), softmax layer and classification layer are replaced. The layer fc_8 is re-adapted to output a vector of nine components. The softmax and classification layers are replaced to calculate the probabilities among nine categories and to compute the cross-entropy loss for multi-class classification with nine classes (classification into one of the 9 rooms that the target environment contains). Fig. 1 shows the architecture used throughout this work.

3.2 Data Augmentation

Data augmentation technique has been proposed as a method to improve the performance of the model by augmenting the number of training instances and preventing over-fitting. This basically consists of creating new pieces of ‘data’ by applying different effects. Moreover, by considering visual effects, the deep learning model ‘learns’ to be robust against them. Regarding the present work, the data augmentation proposed consists of applying visual effects over the original images from the training dataset. The effects applied are those that can actually occur when images are captured in real operating conditions:

- **Rotation:** A random rotation between 10 and 350 degrees is applied over the omnidirectional image.
- **Reflection:** The panoramic image is reflected.
- **Brightness:** The low intensity values are re-adjusted (increased) in order to create a new image brighter than the original one.
- **Darkness:** The high intensity values are re-adjusted (decreased) in order to create a new image darker than the original one. The brightness and darkness effects try to imitate the changes that the illumination conditions of the environment may experience. Moreover, no brightness and darkness are applied at the same time on the same image.
- **Gaussian Noise:** White Gaussian noise is added to the image.
- **Occlusion:** This effect simulates the cases when some parts of the picture are hidden either by some parts of the sensor setup, or some event (such as a person who is in front of an object). This effect is applied by introducing geometrical gray objects over random parts of the image.
- **Blur Effect:** This effect occurs when the image is captured while the camera is moving (the image is blurred).

Fig. 2 shows some examples of the effects applied over a training image. The first image is the original one, obtained directly from the original training dataset, the rest of the images are the original but with a visual effect over them. Departing from the original training dataset, which contains 519 images, the data augmentation is applied and either one or more than one effects are simultaneously applied (except for the bright and dark effects). Hence, the total number of training images is enlarged to 49824 images.

3.3 Bayesian Optimization

The proposed hyperparameters optimization consists of varying those values which can be crucial to address the training process and, at the same time, can be very different depending on the objective of the network. The aim of finding the best setting is to optimize the training process of the CNNs. The hyperparameters considered to be evaluated are the following:

- **Max Epochs.** Positive integer value that indicates the maximum number of epochs to use for training.

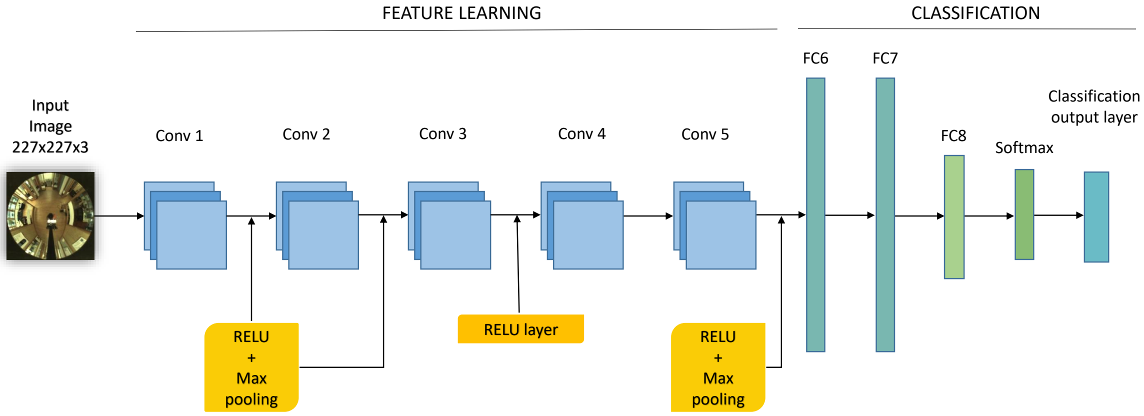


Figure 1: Architecture of the CNN. This network was created departing from AlexNet, adapted and re-trained to retrieve the room where the image was captured within the Freiburg dataset.

- **Initial Learn Rate.** Positive scalar value that controls how much to change the model in response to the estimated error each time the model weights are updated.
- **Momentum.** Scalar value from 0 to 1 that indicates the contribution of the parameter update step of the previous iteration to the current iteration. A value of 0 means no contribution from the previous step, whereas a value of 1 means maximal contribution from the previous step. This hyperparameter is only used with SGDM.
- **L2 Regularization.** Positive scalar value that adds a regularization term for the weights to the loss function. The regularization term is also called weight decay.
- **Squared Gradient Decay Factor.** Positive scalar (less than 1) that indicates the decay rate of squared gradient moving average. This hyperparameter is only used with Adam and RMSProp.
- **Gradient Decay Factor.** Positive scalar (less than 1) that indicates the decay rate of gradient moving average. This hyperparameter is only used with Adam.
- **Epsilon.** Positive scalar that indicates denominator offset. That is, the solver adds the offset to the denominator in the network parameter updates to avoid division by zero. This hyperparameter is only used with Adam and RMSProp.

4 EXPERIMENTS

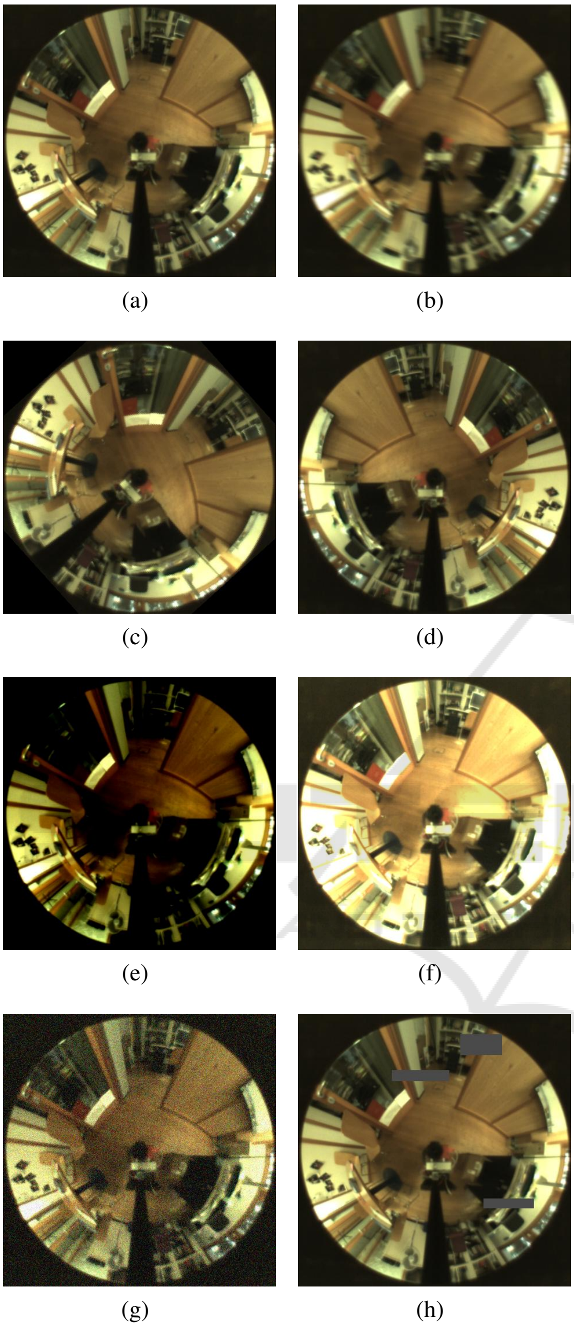
4.1 Localization

This work proposes to use the CNN as a hierarchical model with the aim of: (a) addressing the rough localization as a room retrieval problem (high-level layer) departing from the test image and (b) obtaining holistic descriptors from the input images. The descriptors of the training images will form the low-level layer, and they allow to solve a fine localization as an image retrieval problem, with the holistic descriptors of the test images (also obtained from the CNN).

Regarding the hierarchical localization, the high-level layers permit a **rough localization** and the low-level layers a **fine localization**. The rough step provides faster localization and the fine step considers more accurate information which is used to perform a fine localization step. The proposed hierarchical localization is carried out as the diagram in fig. 3 shows. First (rough localization step), a test image im_{test} is introduced into the CNN and the most likely room c_i in which the image was captured is estimated from the information in the output layers. At the same time, the CNN is also capable of providing holistic descriptors from intermediate layers. Subsequently, after retrieving the room, a more accurate localization is conducted (fine localization step). In this stage, one of the descriptors $\vec{d}_{test}$ is compared with the descriptors $D_{c_i} = \{\vec{d}_{c_i,1}, \vec{d}_{c_i,2}, ..., \vec{d}_{c_i,N_i}\}$ from the training dataset which belong to the retrieved room c_i and the most similar descriptor $\vec{d}_{c_i,k}$ is retained. Finally, the position where the test image was captured is estimated as the coordinates where $im_{c_i,k}$ was captured.

4.2 The Freiburg Dataset

The images used in the present work were obtained from the Freiburg dataset, which is included in the COLD (COsy Localization Database) database (Pronobis and Caputo, 2009). This dataset contains omnidirectional images captured while the robot traversed several paths within the environment.



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 2: Example of data augmentation. (a) Original image captured within the Freiburg environment. One effect is applied over each image: (b) blur effect, (c) random rotation (d) reflection, (e) darkness, (f) brightness, (g) Gaussian noise and (h) occlusion. The images contained in this dataset can be downloaded from the web site https://www.cas.kth.se/COLD/.

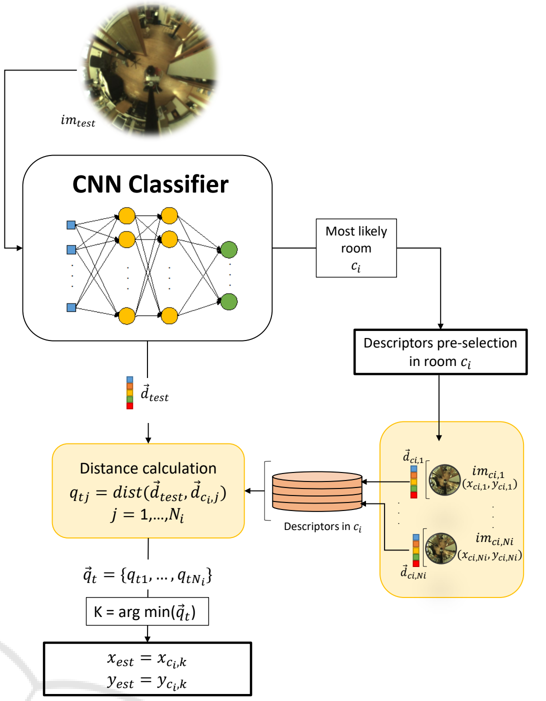


Figure 3: Hierarchical localization diagram. The test image im_{test} is introduced into the CNN. The most likely room is retrieved c_i and the holistic descriptor $\vec{d}_{test}$ is obtained from one of the layers. A nearest neighbour search is done with the descriptors from the training dataset included in the retrieved room and the most similar descriptor ($im_{c_i,k}$) is retained. The position of im_{test} is estimated as the position where $im_{c_i,k}$ was captured.

It includes several rooms such as corridors, personal offices, printer areas, kitchens, bathrooms, etc. The robot tackle the image capturing task under real operating conditions, that is, people that appear and disappear from scenes, changes in the furniture, etc. We use omnidirectional images that were captured in a building of the Freiburg University under three illumination conditions (cloudy days, sunny days and nights). Therefore, with the aim of evaluating how those changes affect the localization, we propose using some of the images captured on a cloudy day for training. Furthermore, the total dataset of cloudy images is used to evaluate the localization error without illumination changes. In addition, the datasets captured during sunny days and at night are used to evaluate the proposed localization approach under illumination changes (brightness and darkness respectively). Apart from the images, a ground truth is also available. This is provided by a laser sensor and it is used only to measure the localization error produced.

Concerning the capturing process, the information was captured following a trajectory along with the whole environment. Therefore, some images contain blur effects and dynamic changes. Moreover, this environment presents the longest trajectory among all the available environments and it also presents wide windows and some glass walls that make the visual localization task tougher. Hence, these issues provide suitability to evaluate the method under real operating conditions.

The images of the dataset were captured in 9 different rooms: a printer area, a kitchen, four offices, a bathroom, a stair area and a long corridor that connects the rooms. The cloudy dataset is downsampled with the aim of obtaining a resultant dataset with a distance of 20 cm between consecutive images. Afterward, the resultant dataset (training dataset) is used to train the CNNs. To summarize, this work proposes the use of a training dataset captured under cloudy conditions and a distance of 20 cm between capture points; a cloudy test dataset, a sunny test dataset and a night test dataset with 519, 2778, 2231 and 2876 images respectively.

4.3 CNN Training

The room retrieval problem has been addressed as a classification problem approach. That is, a deep learning model is re-trained with the objective of retrieving the room where the input image was captured. Concerning the hyperparameters optimization by means of the Bayesian Optimization, five experiments have been carried out. They consist of varying the training dataset, the number of explored points the values of the hyperparameters. The five tackled experiments to train the CNN are as follows:

- Experiment 1: Training with the original dataset. Hyperparamaters to optimize: Initial Learn Rate, Momentum and L2 Regularization. In this experiment, 30 combination of hyperparameters values were carried out.
- Experiment 2: Training with the augmented dataset and hyperparameters that were found as optimal for experiment 1.
- Experiment 3: Training with the augmented dataset and hyperparameters to optimize: Momentum. In this experiment, 8 combination of hyperparameters values were carried out.
- Experiment 4: Training with the augmented dataset and hyperparameters to optimize: Initial Learn Rate and Momentum. In this experiment, 8 combination of hyperparameters values were carried out.
- Experiment 5: Training with the augmented dataset and hyperparameters to optimize: Initial Learn Rate and L2 Regularization. In this experiment, 30 combination of hyperparameters values were carried out.

The optimization training is shown for experiments 4 and 5 in the fig. 4. Concerning the tackled experiments for training optimization, table 1 shows the range of hyperparameters and the obtained optimal values for each experiment. In addition, the classification accuracy for the test images of the previous experiments can be seen in the fig. 5.

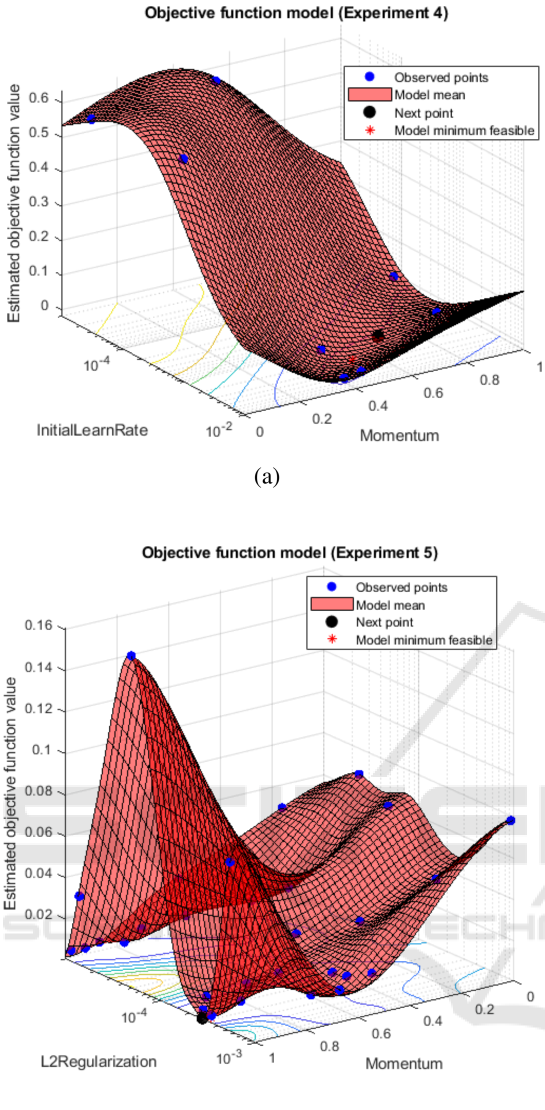
As we can see, the best result for sunny conditions is obtained in experiment 1, which is the only one that does not use the data augmentation images. As we have researched, it does not prove that data augmentation does not work well with sunny conditions in general. The CNN only performs mistakes in a room where the sun's beams pass through the windows at an oblique angle. This fact in combination with the effects of data augmentation, causes confusion in the algorithm reducing the global accuracy. For night and cloudy the best results are found in experiment 2. Moreover, considering the three lighting conditions, the best solution is presented for experiment 3.

Table 1: Hyperparameters values for Bayesian optimization.

Exp	Hyperparameters	Range	Optimum
1	Initial Learn Rate	[1e-4, 1]	0.006
	Momentum	[0.5, 1]	0.539
	L2 Regularization	[1e-10, 1e-2]	3.87e-9
2	Initial Learn Rate	0.006021	-
	Momentum	0.53961	-
	L2 Regularization	3.873e-9	-
3	Initial Learn Rate	1e-3	-
	Momentum	[0, 1]	0.911
	L2 Regularization	1e-4	-
4	Initial Learn Rate	[1e-5, 1e-2]	0.007
	Momentum	[0, 1]	0.384
	L2 Regularization	1e-4	-
5	Initial Learn Rate	1e-3	-
	Momentum	[0, 1]	0.979
	L2 Regularization	[1e-5, 1e-3]	3.06e-4

4.4 Hierarchical Localization by using Holistic Descriptors

As mentioned above, the localization method proposed to address this task is based on a hierarchical localization approach. This consists of using holistic descriptors obtained from an intermediate layer of the trained CNN. This localization is addressed in two steps. The first step is the rough localization, which consists of carrying out the room retrieval task by means of the re-trained CNN. The second step is the fine localization and it consists in estimating the capturing position by using a nearest neighbour search



(a)

(b)

Figure 4: Optimization training for experiments 4 and 5.

method using holistic descriptors. Among the different intermediate layers, we have decided to study the fully connected layers 6 and 7, since they presented more robustness against changes of illumination. Moreover, in general, using the fc_6 layer leads to better results than using fc_7.

Therefore, the considered hierarchical localization approaches are as follows:

1. $CNN_0 + fc_6$: (rough step) A CNN whose training was not optimized (default hyperparameters were selected) and (fine step) using the layer fc_6 to obtain holistic descriptors.
2. $CNN_0 + fc_7$: (rough step) A CNN whose training was not optimized (default hyperparameters were selected) and (fine step) using the layer fc_7 to obtain holistic descriptors.

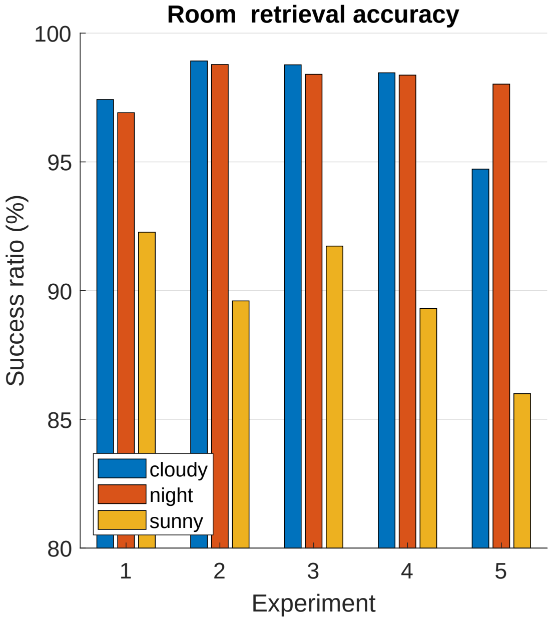


Figure 5: The success ratio (room retrieval) of the CNN trained by means of Bayesian optimization. Results obtained under cloudy (blue), night (red) and sunny (yellow) illumination conditions.

3. $CNN_1 + fc_6$: (rough step) CNN with training optimization associated to experiment 1 (see subsection 4.3) and (fine step) using the layer fc_6 to obtain holistic descriptors.
4. $CNN_2 + fc_6$: (rough step) CNN with training optimization associated to experiment 2 (see subsection 4.3) and (fine step) using the layer fc_6 to obtain holistic descriptors.
5. $CNN_3 + fc_6$: (rough step) CNN with training optimization associated to experiment 3 (see subsection 4.3) and (fine step) using the layer fc_6 to obtain holistic descriptors.
6. $CNN_4 + fc_6$: (rough step) CNN with training optimization associated to experiment 4 (see subsection 4.3) and (fine step) using the layer fc_6 to obtain holistic descriptors.
7. $CNN_5 + fc_6$: (rough step) CNN with training optimization associated to experiment 5 (see subsection 4.3) and (fine step) using the layer fc_6 to obtain holistic descriptors.

The localization error is measured as the euclidean distance between the estimated position and the current position (given by the ground truth of the dataset). Moreover, the localization error is evaluated against different illumination conditions. Since the objective is to study the robustness of the proposed method against illumination changes in the environment of work. Obtained results are shown in the fig. 6.

In general, the localization error is related to the success rate. That is, when the CNN is more capable of retrieving successfully the images, the localization error is lower. Furthermore, the worst localization error under sunny illumination conditions is found in the CNN associated with training 5, which presented the lowest success ratio among all the tackled training processes. Regarding the best values for cloudy and night, they are given in CNN_2, CNN_3 and CNN_4, whose training was based on the Bayesian optimization.

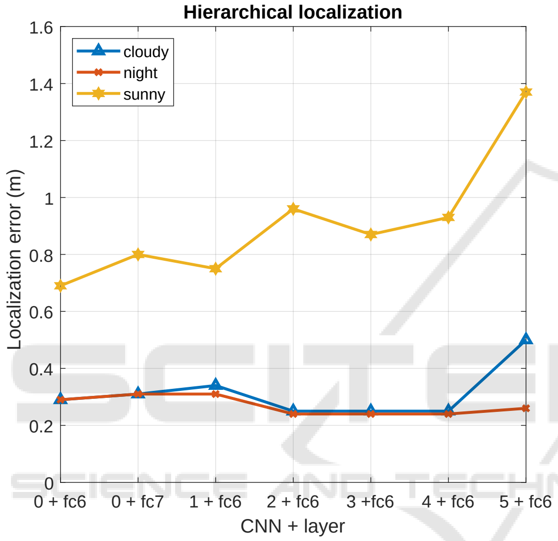


Figure 6: Hierarchical localization results. Seven combinations of CNN and its associated intermediate layer are evaluated to address the localization task in a hierarchical way. Results presented under cloudy (blue), night (red) and sunny (yellow) illumination conditions. The average localization error was calculated by using 2778 (cloudy), 2231 (night) and 2876 (sunny) robot positions.

5 CONCLUSIONS

In this work, we have evaluated the use of a deep learning technique to build hierarchical topological models for localization. The developed tool consists of a convolutional neural network trained with omnidirectional images for addressing a room retrieval task. Moreover, the re-trained CNN is also proposed to obtain holistic descriptors from intermediate layers with the aim of obtaining information that characterizes the input image. Therefore, throughout the present work, we have evaluated the use of two techniques to improve the training process of the CNN and the use of the network to solve the localization by means of a method based on a hierarchical localization approach.

As for the training optimization process, Bayesian optimization is capable of improving the training process of the CNN in general, since the average success rate increases when this approach is considered. Furthermore, data augmentation also leads to obtain CNNs that perform better results. Concerning the outputs obtained under night illumination conditions, they have been considerably improved. Since they lead to reach a similar success ratio than those performances that do not consider illumination changes (i.e., under cloudy conditions). However, performance is not considerably improved when the sunny illumination condition is evaluated. After a profound analysis, we reached the conclusion that this increase of mistakes is due to the fact that room number 1 (printer area) is severely affected by the beams of light since the orientation of the windows is north. This fact, together with the effects of data augmentation, causes confusion during the training process and, hence, reduces global accuracy.

Regarding the training of a pre-trained CNN, this presents good results to carry out a room retrieval task departing from omnidirectional images. This result presents a novelty in the field, since, until now, scarce works had proposed a deep learning model based on omnidirectional images for localization purposes. In addition, the intermediate layers also are capable of providing vectors of information that can be used to obtain global-appearance descriptors.

Therefore, a hierarchical localization approach is addressed by using the CNN to retrieve the room and also to obtain holistic descriptors. This method has been evaluated and has demonstrated to be suitable to address the localization task. Obtained results show that the localization error is considerably low under cloudy and night conditions since the minimum error obtained is 25 cm and the considered environment presents an average distance between images of around 20 cm. As for the sunny illumination conditions, the localization error is higher, due especially to the lower success rate, given by the beams of light in the printer area. On the contrary, the effects produced by darkness have been completely reduced, since the localization error associated with the night illumination condition is equal to the ones obtained without illumination changes.

In future works, we will focus on reducing the localization error under sunny illumination conditions. Furthermore, we will extend the use of deep learning techniques for localization by using different tools such as autoencoders or LSTM networks. Last, we would also like to create and evaluate localization approaches based on CNNs in outdoor environments.

ACKNOWLEDGEMENTS

This work has been supported by the Generalitat Valenciana and the FSE through the grant ACIF/2020/141 and the project AICO/2019/031: “Creación de modelos jerárquicos y localización robusta de robots móviles en entornos sociales”; and by the Spanish government through the project DPI 2016-78361-R (AEI/FEDER, UE): “Creación de mapas mediante métodos de apariencia visual para la navegación de robots”.

REFERENCES

Arroyo, R., Alcantarilla, P. F., Bergasa, L. M., and Romera, E. (2016). Fusion and binarization of cnn features for robust topological localization across seasons. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4656–4663.

Bergstra, J. and Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(Feb):281–305.

Bergstra, J., Yamins, D., and Cox, D. D. (2013). Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures.

Cebollada, S., Payá, L., Flores, M., Peidró, A., and Reinoso, O. (2021). A state-of-the-art review on mobile robotics tasks using artificial intelligence and visual data. *Expert Systems with Applications*, 167:114195.

Cebollada, S., Payá, L., Flores, M., Román, V., Peidró, A., and Reinoso, O. (2020). A deep learning tool to solve localization in mobile autonomous robotics. In *ICINCO 2020, 17th International Conference on Informatics in Control, Automation and Robotics (Lieusaint-Paris, France, 7-9 July, 2020)*. Ed. INSTICC.

Cebollada, S., Payá, L., Román, V., and Reinoso, O. (2019). Hierarchical localization in topological models under varying illumination using holistic visual descriptors. *IEEE Access*, 7:49580–49595.

Chaves, D., Ruiz-Sarmiento, J., Petkov, N., and Gonzalez-Jimenez, J. (2019). Integration of cnn into a robotic architecture to build semantic maps of indoor environments. In *International Work-Conference on Artificial Neural Networks*, pages 313–324. Springer.

Ding, J., Chen, B., Liu, H., and Huang, M. (2016). Convolutional neural network with data augmentation for sar target recognition. *IEEE Geoscience and remote sensing letters*, 13(3):364–368.

Domhan, T., Springenberg, J. T., and Hutter, F. (2015). Speeding up automatic hyperparameter optimization of deep neural networks by extrapolation of learning curves. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

Dymczyk, M., Gilitschenski, I., Nieto, J., Lynen, S., Zeisl, B., and Siegwart, R. (2018). Landmarkboost: Efficient visualcontext classifiers for robust localization. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 677–684.

Falkner, S., Klein, A., and Hutter, F. (2018). Bohb: Robust and efficient hyperparameter optimization at scale. *arXiv preprint arXiv:1807.01774*.

Feurer, M. and Hutter, F. (2019). Hyperparameter optimization. In *Automated Machine Learning*, pages 3–33. Springer.

Han, D., Liu, Q., and Fan, W. (2018). A new image classification method using cnn transfer learning and web data augmentation. *Expert Systems with Applications*, 95:43–56.

Kotthoff, L., Thornton, C., Hoos, H. H., Hutter, F., and Leyton-Brown, K. (2019). Auto-weka: Automatic model selection and hyperparameter optimization in. *Automated Machine Learning: Methods, Systems, Challenges*, page 81.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105.

Levine, S., Pastor, P., Krizhevsky, A., Ibarz, J., and Quillen, D. (2018). Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *The International Journal of Robotics Research*, 37(4-5):421–436.

Payá, L., Peidró, A., Amorós, F., Valiente, D., and Reinoso, O. (2018). Modeling environments hierarchically with omnidirectional imaging and global-appearance descriptors. *Remote Sensing*, 10(4):522.

Pronobis, A. and Caputo, B. (2009). COLD: COsy Localization Database. *The International Journal of Robotics Research (IJRR)*, 28(5):588–594.

Salamon, J. and Bello, J. P. (2017). Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal Processing Letters*, 24(3):279–283.

Shvets, A. A., Rakhlin, A., Kalinin, A. A., and Iglovikov, V. I. (2018). Automatic instrument segmentation in robot-assisted surgery using deep learning. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 624–628.

Sinha, H., Patrikar, J., Dhekane, E. G., Pandey, G., and Kothari, M. (2018). Convolutional neural network based sensors for mobile robot relocalization. In *2018 23rd International Conference on Methods Models in Automation Robotics (MMAR)*, pages 774–779.

Snoek, J., Rippel, O., Swersky, K., Kiros, R., Satish, N., Sundaram, N., Patwary, M., Prabhat, M., and Adams, R. (2015). Scalable bayesian optimization using deep neural networks. In *International conference on machine learning*, pages 2171–2180.

Wozniak, P., Afrisal, H., Esparza, R. G., and Kwolek, B. (2018). Scene recognition for indoor localization of mobile robots using deep cnn. In *International Conference on Computer Vision and Graphics*, pages 137–147. Springer.