Development and use of a convolutional neural network for hierarchical appearance-based localization

S. Cebollada¹ · L. Payá¹ · X. Jiang² · O. Reinoso¹

Published online: 29 September 2021
© The Author(s) 2021

Abstract

This paper reports and evaluates the adaption and re-training of a Convolutional Neural Network (CNN) with the aim of tackling the visual localization of a mobile robot by means of a hierarchical approach. The proposed method addresses the localization problem from the information captured by a catadioptric vision sensor mounted on the mobile robot. A CNN is adapted and evaluated with a twofold purpose. First, to perform a rough localization step (room retrieval) by means of the output layer. Second, to refine this localization in the retrieved room (fine localization step) by means of holistic descriptors obtained from intermediate layers of the same CNN. The robot estimates its position within the selected room/s through a nearest neighbour search by comparing the obtained holistic descriptor with the visual model of the retrieved room/s. Additionally, this method takes advantage of the likelihood information provided by the output layer of the CNN. This likelihood is helpful to determine which rooms should be considered in the fine localization process. This novel hierarchical localization method constitutes an efficient and robust solution, as shown in the experimental section even in presence of severe changes of the lighting conditions.

Keywords Catadioptric Vision Sensor · Omnidirectional Images · Holistic Description · Deep Learning · Hierarchical Localization

This work has been supported by the Generalitat Valenciana and the FSE through the grant ACIF/2017/146 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.”.

✉ S. Cebollada
sergio.cebollada@umh.es

L. Payá
lpaya@umh.com

X. Jiang
xjiang@uni-muenster.de

O. Reinoso
o.reinoso@umh.com

¹ Department of Systems Engineering and Automation, Miguel Hernández University, Elche, Spain

² Department of Computer Science, University of Münster, Münster 48149, Germany

1 Introduction

Over the past few years, the use of omnidirectional cameras together with computer vision techniques have proved to be a robust option to solve the localization task in mobile autonomous robotics. Among the methods proposed to extract the most relevant information from the images, the holistic or global-appearance description approaches are a successful solution, since they lead to more direct localization algorithms based on a pairwise comparison between descriptors.

Regarding the mapping task, building hierarchical models departing from global-appearance descriptors permits solving the localization problem efficiently. This method consists in arranging the visual information hierarchically in different layers of information in such a way that the localization can be solved in two main steps. First, a coarse localization in an area of the environment and second, a fine localization in this pre-selected area.

Additionally, during the past few years, the emergence of faster and more efficient hardware devices has led to contributions which propose artificial intelligence (AI) techniques to address computer vision and robotics problems. Within the AI techniques, convolutional neural networks (CNNs) are a very popular technique to address a variety of problems in mobile robotics. A complete and varied training is crucial for the success of these tools, and to this aim, a large training dataset must be available. Hence, data augmentation is commonly proposed as a solution to increase the training instances while avoiding overfitting.

In light of the above information, the aim of this work is to introduce and critically evaluate the performance of a variety of approaches using a convolutional neural network, to carry out the tasks of mapping and localization for mobile robots in indoor environments. The efficiency of these techniques will be assessed through their ability to robustly estimate the position of the robot using the information stored on the map and the computing time required for it. To address the proposed evaluation, the unique source of information is the set of images obtained by an omnidirectional vision sensor installed on the mobile robot, which moves in an indoor environment under real-operating conditions.

The novelty of this work is a hierarchical approach based on a re-adapted CNN that is used to efficiently solve the localization task. In general, the idea of this work is to re-adapt and use a unique deep learning tool with a dual purpose: (1) estimating in which room the robot is currently located (rough localization step) using the output layer and (2) refining the localization in the retrieved room (fine localization step) by means of holistic descriptors which are obtained from intermediate layers of the same CNN. Our main contributions in this work can be summarized as follows.

- We adapt and train a CNN as a classifier to retrieve the room where an input image was obtained.
- We evaluate the use of different intermediate convolutional layers of this CNN to obtain holistic descriptors and use them to address the task of fine localization in different environments.
- We study the performance of the proposed deep learning approach to address the complete hierarchical localization.
- We propose an algorithm that considers the likelihood information provided by the final layer of the CNN to strengthen the rough localization task and subsequently the whole hierarchical localization.

The remainder of the paper is structured as follows. Section 2 presents a review of the related literature. After that, section 3 presents the method to adapt the CNN in order to address the proposed problem and section 4 explains the hierarchical localization method based on the adapted CNN. Section 5 presents all the experiments which were tackled to test the validity of the proposed methods, in a variety of environments and lighting conditions. Finally, section 6 presents the conclusions and future works.

2 State of the art

Machine learning techniques have been used to solve a variety of problems in computer vision and robotics (Cebollada et al. 2021). Gonzalez et al. (2018) use machine learning to detect different levels of slippage for robotic missions in Mars; Dymczyk et al. (2018) present the use of a boosted classifier to classify landmark observations and carry out the localization task in a more robust fashion. Meattini et al. (2018) propose a human-robot interface system based on electromyography sensors and through merging pattern recognition and factorization techniques, the robot learns the optimal hand configuration for grasping. Concerning deep learning, it is a subfield of machine learning that has gained much interest recently, mainly due to the improvements obtained in fields such as processing systems. This technique basically consists in learning directly from a data set and their expected outputs (or correct labelling) by using layers of increasingly meaningful representations (Goodfellow et al. 2016). A number of recent works use such techniques in the field of robotics. For instance, Lenz et al. (2015) propose a deep learning approach to solve the problem of detecting robotic grasps in a scene which contains objects; Levine et al. (2018) trained a convolutional neural network for robotic grasping from monocular images through learning a hand-eye coordination; Shvets et al. (2018) use deep learning segmentation to distinguish between different surgical instruments regarding Robot-Assisted Surgery. As for mobile robotics, Zhu et al. (2017) propose deep reinforcement learning to address target-driven visual navigation.

Regarding the use of CNNs to solve tasks in mobile robotics, there are many works that have proved success by using this technique. For instance, Sinha et al. (2018) propose a CNN to process data from a monocular camera and tackle an accurate robot re-localization in GPS-denied indoor and outdoor environments. Wozniak et al. (2018) use a transfer learning technique to retrain a CNN to classify places among 16 rooms, in which the images are acquired by a humanoid robot. More recently, Chaves et al. (2019) propose a CNN to build a semantic map. Concretely, they use the network to detect objects in images and after that, the results are placed within a geometric map of the environment. A wide review can be found in the work presented by Voulodimos et al. (2018).

Among the different visual sensors that can be mounted on a mobile robot to capture information from the environment, omnidirectional cameras have been commonly used during the past few years. For instance, Abadi et al. (2015) use omnidirectional vision to detect obstacles with the aim of carrying out autonomous navigation and Liu et al. (2018) use omnidirectional images to provide an accurate estimation of the position and orientation of the robot within outdoor environments. More recently, Li et al. (2019) propose a novel method to avoid obstacles for autonomous wheeled robots using HyperOmni Vision and the DWA (Dynamic Window Approach) collision avoidance algorithm.

In order to tackle the mapping and localization tasks through visual information, the extraction of the most relevant information from the images constitutes a crucial step.

Two main frameworks are commonly proposed to carry out these tasks: either extracting the most outstanding points of the image and calculating a local descriptor of each one or obtaining a unique descriptor per image which contains global information about it. A wide range of works have been proposed in mobile robotics by using local descriptors (for example, Valiente et al. (2018), He et al. (2018), or Luo et al. (2018)) and also by using global-appearance descriptors (such as Amorós et al. (2018), Çevik and Çevik (2019) or Dong-Won S. (2019)) and both methods have been successfully used to address mapping and localization. In the present paper, in line with previous works (Cebollada et al. 2019b), the global-appearance description method is used to obtain information from the visual datasets and address the hierarchical localization.

Originally, global-appearance or holistic description is based on analytical or hand-crafted methods, i.e., they depart from an image and carry out some mathematical transformations to obtain a vector ($\mathbf{d} \in \mathbb{R}^{(\times 1)}$) with representative information from the image. For instance, Dalal and Triggs (2005) introduced the HOG descriptor, that consists in dividing the image into k_1 horizontal cells and calculating a histogram from the gradient orientation per each cell with b bins per each histogram (Payá et al. 2016). These histograms, arranged in a unique column, compose the final descriptor $\mathbf{d} \in \mathbb{R}^{b \cdot k_1 \times 1}$. Oliva and Torralba (2006) proposed the descriptor *gist*. In previous works (Cebollada et al. 2019a, b), this description method consisted in creating m_2 images from the original panoramic image with different resolution, then applying Gabor filters over the m_2 images with m_1 different orientations and afterwards, grouping the pixels of each image into k_2 horizontal blocks to calculate the average value of each block. A more detailed description of these description methods can be found in Payá et al. (2016).

More recent works have proposed the use of CNNs to obtain holistic descriptors from the activations of the intermediate layers. In this sense, the hidden layers provide descriptors which can be used to characterize the input data. This idea has already been used by some authors such as Arroyo et al. (2016), who use a CNN that automatically learns to generate descriptors which are robust against changes of seasons in order to carry out a topological localization. Wozniak et al. (2018) also use the feature extracted from a layer to train a linear SVM (Support Vector Machines) classifier. Mancini et al. (2017) use this visual information to carry out place categorization with a Naïve Bayes classifier. Payá et al. (2018) propose using the information in intermediate layers of a pre-existing CNN (places CNN (Zhou et al. 2014)) to perform localization. However, this pre-existing network was trained to a different purpose. Instead of it, training a network with images from the target environment could be doubly beneficial in hierarchical localization, since it is expected: (1) to improve the rough localization step and (2) to provide holistic descriptors from intermediate layers which achieve a more accurate fine localization in the target environment. Cebollada et al. (2020) show the advantages of using descriptors obtained from the intermediate layers of a re-trained CNN to solve the visual localization as a batch image retrieval problem (with no hierarchical process).

Concerning the training process, a large dataset is crucial to achieve a robust performance. Nevertheless, sometimes, the training dataset is smaller than required and then, the deep model can not be properly trained to reach the desired solution. In order to solve this issue, the 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 overfitting. Data augmentation basically consists in creating new pieces of 'data' by applying different effects over the original images. Some authors have already used data augmentation to solve their deep learning tasks. For example, Guo and Gould (2015) used data augmentation to improve a CNN training to solve an object detection task, Ding et al. (2016) proposed three data

augmentation methods to carry out a SAR (Synthetic Aperture Radar) target recognition in order to make the CNN 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 and then create a CNN to classify these data. Moreover, Perez and Wang (2017) present a work about the effectiveness of the data augmentation to solve the classification by means of deep learning. Shorten and Khoshgoftaar (2019) present a survey about the existing methods for Data Augmentation, promising developments, and meta-level decisions for implementing Data Augmentation. Nonetheless, the previously proposed data augmentation methods do not match the visual effects that can occur when the robot moves through the target environment under real-operation conditions. Therefore, the present work performs a data augmentation process which focuses on those specific visual effects.

Previous works (Cebollada et al. 2019b; Payá et al. 2018) have demonstrated that using hierarchical models with omnidirectional imaging and global-appearance descriptors provides an efficient and robust solution to address the visual localization. Those works rely on arranging the visual information (obtained by global-appearance description methods) in several layers. Afterwards, the localization task is solved by means of an image retrieval problem in two steps: a fast but inaccurate step (rough localization) and a local localization step which provides more accuracy (fine localization).

Therefore, this work proposes using a CNN to obtain a hierarchical model with the aim of: (a) addressing the rough localization step as a room retrieval problem (high-level layer), (b) using the likelihood information to increase the efficiency of the rough step and (c) obtaining holistic descriptors departing from the developed network and solving the fine localization step by means of a nearest neighbour search. With this aim, the AlexNet CNN architecture (Krizhevsky et al. 2012) is re-adapted (layers replacement and training from scratch). After that, the CNN is capable of retrieving the room where the image was captured (rough step) and at the same time, global-appearance descriptors are generated from intermediate layers to refine the position estimation (fine localization step). The objective is to provide a feasible solution, which can be used to solve efficiently the localization task in different environments and circumstances, avoiding complex and computing-time-demanding deep learning developments. Hence, due to the simplicity of AlexNet, this network is suitable for the objective.

To sum up, some authors have developed CNNs to carry out classification tasks. Additionally, previous works have also proposed solving the localization task by using intermediate layers of CNNs as holistic description method. The present work tries to go one step beyond and proposes an approach based on a unique network which is re-adapted to address both tasks at the same time (room retrieval and holistic description extraction), hence, solving the complete hierarchical localization problem. The method takes also advantage of the likelihood information provided by the final layer of the CNN to decide how many rooms should be considered to solve the fine localization step. Solving both problems with the same CNN leads to a hierarchical method which has not been studied in the current state of the art and it provides robust solutions regarding localization error and computing time in comparison with previously proposed approaches, as detailed in the experimental section.

3 CNN adaptation

The process followed to adapt the CNN with the purpose of addressing hierarchical localization can be summarized as follows: the CNN (AlexNet) architecture is firstly adapted and then re-trained to solve a room retrieval task. This section details the process followed to adapt a re-train the CNN. Once the model is trained, it is ready to carry out the hierarchical localization process from the input image, as explained in sect. 4.

Building and training a network from scratch can lead to reasonably good results, but it requires a lot of effort: (1) experience with network architectures, (2) a huge amount of training data and (3) a considerable computing time. Using a pre-trained network like AlexNet or GoogLeNet for transfer learning eases considerably the starting point. Nevertheless, the proposed approach can not depart from transfer learning, since the input data are panoramic images (size of the panoramic images is $128 \times 512 \times 3$). Hence, in this case, the input layers must be resized and many of the downstream parameters will be no longer valid. The present work proposes using the AlexNet architecture and following a process similar to transfer learning (starting with pre-existing architectures), but starting from scratch with the parameters tuning. We propose departing from AlexNet as basic architecture because it has been successfully used in previous works to develop new classification tasks (such as Han et al. (2018)). From it, we perform a modification of some layers and a complete training from scratch, to adapt the network to the proposed hierarchical localization task. In this case, the last three layers are replaced to adapt the network to a room classification task. These layers are: fully-connected layer (fc_8), softmax layer and classification layer. First, the layer fc_8 has been re-adapted to output a vector of nine components. Second, the softmax and classification layers have been re-adapted to respectively determine 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). Additionally, the input layer is also replaced, since the input layer of AlexNet was configured to receive 227×227 images and our panoramic dataset contains panoramic images (128×512). Resizing the input panoramic images to a size 227×227 would avoid starting the training from scratch, but resizing the panoramic images would abruptly change their appearance and affect significantly the performance of the network.

After these changes of layers, the network is ready to be trained with the training set of panoramic images. We trained the CNN off-line on a NVIDIA GEFORCE GTX 1080TI ® GPU system. After every 30 iterations, the performance of the partially trained network was evaluated by using the data for validation. The first training departs from the modified version of AlexNet. Once the first training is finished, the network obtained is used as departing network in the following training with a modification of the training parameters. The idea is to conserve the architecture but to continue the tuning of the layers' parameters. Fig. 1 shows the architecture used throughout this work and Fig. 2 shows the training progress regarding accuracy and loss.

Regarding the data augmentation proposed in the present work, it consists in applying visual effects over the original images from the training dataset. Traditional data augmentation techniques consider some alterations in the images, such as flips, translations along the horizontal and vertical axes, pure rotations of the pixels in the image, scale or crop (Guo and Gould 2015). In the present work, the data augmentation has been designed specifically to obtain a robust CNN for localization. Therefore, to obtain new samples, we consider a variety of visual effects to each training image, which can

Fig. 1 Diagram which shows the CNN architecture created departing from AlexNet and trained to classify the room within the Freiburg dataset. It also shows how the holistic descriptors are extracted from the layers of the CNN. Descriptors $\vec{d}_{conv4}$ and $\vec{d}_{conv5}$ are obtained from the 2D convolutional layers, and descriptors $\vec{d}_{fc6}$, $\vec{d}_{fc7}$ and $\vec{d}_{fc8}$ are obtained from the fully-connected layers

actually occur when the robot operates in real operation conditions. Hence, through this data augmentation, the CNN is expected to be more robust against the challenging conditions that can occur in the scenario where the robot moves. Considering it, the effects that we consider to perform the data augmentation are:

- **Rotation** A random rotation between 10 and 350 degrees is applied over the omnidirectional image, which implies a horizontal shift of the panoramic image. This effect

Fig. 2 The figure shows the evolution of the training process and some additional information about it. The upper plot represents the evolution of the mini-batch accuracy (blue) and validation accuracy (black) and the bottom plot shows the evolution of the mini-batch loss (red) and validation loss (black) (Colour figure online)

emulates the different orientations that the robot may have at a specific point in the ground plane when acquiring a new image.

- **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. No brightness and darkness are applied at the same time on the same image. The brightness and darkness effects try to imitate the changes that the lighting conditions of the environment may experience during the day.
- **Gaussian noise** White Gaussian noise is added to the image. It emulates the possible noise that the visual sensor can introduce in 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** Some degrees of blur are applied to each training image to emulate the case in which the robot is moving while the image is captured.

Figure 3 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 it. Departing from the original training dataset, which contains 519 images, the data augmentation is applied and either none, one, or more than one effects are simultaneously applied (except for the brightness and darkness effects, which are never applied at the same time over an image). Hence, the total number of training images is enlarged to 49824 images.

Concerning the training hyperparameters, a study was performed with the aim of selecting those that optimize the training process. These hyperparameters are selected for the first training process, that is, when the model is re-trained for the first time after re-adapting the

Fig. 3 Example of data augmentation. **a** Original image captured within the Freiburg environment. One effect is applied over each image: **b** blur effect, **c** darkness, **d** brightness, **e** Gaussian noise, **f** occlusion

intermediate layers. Some of the hyperparameters are set to the established default values. The following options are remarked:

- Mini Batch Size: 10
- Initial Learn Rate: 0.01
- Validation Frequency: 3
- Validation Patience: Inf

4 Localization using deep learning

Since the use of holistic description methods based on deep learning can improve the results obtained for localization, the present work presents a re-adapted CNN to solve the visual localization in a hierarchical way. To summarize, the AlexNet CNN architecture was redefined and trained from scratch, as described in the previous section. After that, global-appearance descriptors are generated from the intermediate layers to address the localization. Hence, the hierarchical localization is basically solved as follows: (a) addressing the rough localization as a room retrieval problem (high-level layer) departing from the test image, (b) using the likelihood information to optimize the rough step and (c) 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).

Concerning the process to obtain the holistic descriptors from the CNN, it is as follows. First, the CNN is trained with the images from the training dataset (including data augmentation). Second, once the CNN is trained, a test image im_{test} is introduced into the CNN. Third, the holistic descriptors are obtained from different layers. About the 2D convolutional layers ($conv_4$, and $conv_5$), the descriptors are obtained by selecting a channel from the layer and arranging the generated data (matrix) in a single column (vector). To establish the optimal channel per convolutional layer, previous experiments are carried out and

afterwards, the same channel is used for all the experiments developed. In the case of the fully-connected layers (fc_6 , fc_7 and fc_8), the output is directly the vector used as descriptor. Figure 1 shows the process to extract the global-appearance descriptors from the trained CNN and Table 1 summarizes the size of each descriptor.

Regarding the hierarchical localization, it is based on models whose information is organized in several layers with different levels of granularity. The objective of arranging the information in this way is to carry out the localization task more efficiently than the conventional method proposed in previous works (Cebollada et al. 2019c). In this sense, the high-level layers permit a **rough localization step** and the low-level layers a **fine localization step**. The rough step provides faster localization and the fine step considers more accurate information which is used to perform a fine localization step. The hierarchical localization analyzed in previous works (Cebollada et al. 2019b) consists basically in 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 $R = \{\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_{n_g}\}$ is obtained for each group, where n_g is the number of groups. Afterwards, in order to solve the localization task, a new image is obtained im_{test} and its holistic descriptor is calculated $\mathbf{d}_{test}$. This descriptor is compared with all the representatives R and the most similar representative $\mathbf{r}_k$ is retained (rough localization step); after that, a new comparison is carried out between $\mathbf{d}_{test}$ and the descriptors contained in the group k , $D_k = \{\mathbf{d}_{k,1}, \mathbf{d}_{k,2}, \dots, \mathbf{d}_{k,N_k}\}$. 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).

Therefore, the idea of the present work is to build a unique CNN that, apart from retrieving the room where the image was captured, is also able to provide a holistic descriptor that characterizes the image better than the holistic methods proposed in the current state of the art. Once the CNN is properly trained, it will be able to solve the rough localization step (i.e. the room retrieval). Regarding the use of the CNN to solve the fine localization step, this work proposes to use the layers $conv_4$, $conv_5$, fc_6 , fc_7 and fc_8 of the re-trained CNN to obtain holistic descriptors and to use those descriptors to estimate the position within a room where an image was captured (i.e. the image retrieval).

The diagram in Fig. 4 outlines the proposed hierarchical localization. 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 retrieved. The information in the output layer is used to this purpose. At the same time, the CNN is also capable of providing holistic descriptors ($\mathbf{d}_{test,conv_4}$, $\mathbf{d}_{test,conv_5}$, $\mathbf{d}_{test,fc_6}$, $\mathbf{d}_{test,fc_7}$ or $\mathbf{d}_{test,fc_8}$) from intermediate layers. Subsequently, after retrieving the room, a more accurate localization is conducted (fine localization step). In this stage, one of the descriptors $\mathbf{d}_{test}$ is compared with the descriptors $D_{c_i} = \{\mathbf{d}_{c_i,1}, \mathbf{d}_{c_i,2}, \dots, \mathbf{d}_{c_i,N_i}\}$ from the training dataset which belong to the retrieved room c_i and the most similar descriptor $\mathbf{d}_{c_i,k}$ is retained. This comparison is carried out by calculating the cosine distance between descriptors, because it presented a good performance to

Table 1 Size of each descriptor obtained from the different layers of the CNN

Layer	Size of descriptor
conv4	180
conv5	180
fc6	4096
fc7	4096
fc8	9

Fig. 4 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 ($im_{ci,k}$) is retained. The position of im_{test} is estimated as the position where $im_{ci,k}$ was captured. The cosine distance is used to calculate the distance between descriptors

calculate distances between descriptors in previous works (Cebollada et al. 2019a). Finally, the position where the test image was captured is estimated as the coordinates where $im_{ci,k}$ was captured.

5 Experiments

The experiments detailed in this section and the training of the CNN have been carried out with a PC with a CPU Intel Core i7-7700 ® at 3.6 GHz. In this section, we evaluate the performance of the CNN to solve the localization problem, and we analyze the results. Hence, the remainder of this section is structured as follows. Subsection 5.1 presents the dataset of images used for mapping and localization, as well as for training the CNN. The subject. 5.2 shows the development, training and performance of the CNN; subsection. 5.3 outlines the use of

this deep learning technique to obtain holistic descriptors to carry out the batch localization task. Finally, subsect. 5.4 presents the use of the CNN to tackle a hierarchical localization task.

5.1 Dataset

The images used in the present work were obtained from the COLD (COsy Localization Database) dataset (Pronobis and Caputo 2009). They were used both to train the CNN and to carry out the experiments. This database is open access and is composed of images captured from different indoor environments under three illumination conditions (cloudy days, sunny days and at nights). The information was captured following a trajectory along the whole environment. The movement of the robot is contained in the floor plane and it captures omnidirectional images using a catadioptric vision system mounted on it. Moreover, some images also contain blur effects and dynamic changes, then, all this variety of adverse effects make this set of images suitable to test the proposed method in an indoor environment under real operation conditions. The dataset of images used to train the CNN and to evaluate the localization task proposed is the Freiburg Dataset and among all the information provided, the omnidirectional images are selected as starting point to carry out the CNN training. The choice of this dataset is due to the fact that it was captured in a relatively large environment and also it presents wide windows and some glass walls that challenge the visual localization task. Before using the visual information, a conversion from omnidirectional to panoramic images is tackled, since one of the aims of this work is to compare the global-appearance descriptors obtained from the CNN with the hand-crafted analytic description methods based on panoramic images. Furthermore, the design of a CNN based on panoramic images constitutes an interesting option, because this type of networks are commonly based on conventional non-panoramic images, hence, this CNN can be used for future similar works based also on panoramic images.

The images of the Freiburg 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 dataset obtained under cloudy illumination conditions is used as training dataset, since these images are less affected by illumination issues than the images of the sunny and night datasets. This set of images is downsampled with the aim of obtaining a resultant dataset with a distance of 20 cm between consecutive images, since this allows us to compare results with those obtained in previous works (Cebollada et al. 2019a, b). Afterwards, the resultant dataset (training dataset) is used to train the CNN and it is also considered as the visual model for later localization. The rest of images are used to create the test dataset that is used to evaluate the accuracy of the CNN and the efficiency of the localization methods proposed. Concerning the datasets of images captured under sunny days (sunny dataset) and during night (night dataset), they are directly used to evaluate the efficiency of the localization methods under changes of illumination conditions. Figure 5 shows some examples of panoramic images under the three different illumination conditions. These examples permit noticing how the illumination affects the images. For instance, the image captured at night (Fig. 5b) is darker and the light comes directly from the bulb of the roof, whereas, the image captured during a sunny day (Fig. 5c)

Fig. 5 Example of panoramic images from the Freiburg environment under **a** cloudy, **b** night and **c** sunny illumination conditions. They were captured in **a** the printer area, **b** the stairs area and **c** the kitchen

shows that the light source comes from the windows and also shows some reflection on the floor. Therefore, to sum up, the image dataset used along this work consists in 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.

Apart from using the Freiburg dataset, some extra evaluations are carried out with the Saarbrücken dataset, which is also contained in the COLD dataset. This environment is similar to Freiburg, and it also contains several rooms such as printer area, bathroom and offices. This dataset is used to evaluate the effectiveness of using the Freiburg CNN to obtain holistic descriptors in different environments. The training and test datasets are obtained in the same way: downsampling the cloudy dataset to obtain the training dataset and storing the discarded images to obtain the cloudy test dataset. Table 2 shows the datasets used along the present work to carry out the experiments and also the dataset created departing from Freib_train (519 images) to tackle the data augmentation process, obtaining Freib_train_DA (49824 images).

5.2 Experiment 1. Development, training and evaluation of the CNN in a room retrieval task

The re-training process of the neural network is as follows. First, (1) the CNN architecture is obtained from the AlexNet CNN and a layer replacement is tackled (Fig. 1 shows the final architecture). Then, (2) the training data (consisting in a set of images with labeling) is augmented by a data augmentation technique. After that, (3) the training options are adjusted according to the training specifications. (4) Last, re-trainings of the network are conducted by adapting the training options to produce a more accurate CNN until the network is capable of achieving a 97% of correct estimations by using validation data, that is, data contained in the training dataset which are exclusively used during the process of checking the amount of correct estimations with the current parameters of the layers.

Finally, once the CNN is properly trained, its accuracy ($acc\%$) is measured as $acc\% = (N_{ok}/N_{test}) \times 100$, where N_{ok} is the number of images that have been correctly retrieved and N_{test} is the number of images that compose the test dataset to evaluate. In this case, the three test datasets (cloudy, night and sunny) are used to evaluate the accuracy of the CNN after each training phase. Through this evaluation, the final accuracy values obtained were 98.71%, 96.52% and 92.87% respectively. Moreover, with the aim of addressing a more challenging evaluation of the network trained, visual effects are applied over the cloudy test dataset by means of data augmentation. This augmented dataset has

Table 2 Datasets obtained from the COLD database to carry out the experiments

Dataset name	Illumination condition	Number of images
Freib_train	Cloudy	519
Freib_test_cloudy	Cloudy	2778
Freib_test_night	Night	2876
Freib_test_sunny	Sunny	2231
Saarbr_train	Cloudy	566
Saarbr_test	Cloudy	1630
Freib_train_DA	Cloudy	49824

249.120 images. The accuracy obtained by evaluating this cloudy test dataset augmented over the CNN is 98.34%. Therefore, from the results obtained, the conclusion achieved is that the CNN is properly trained to classify the input image into the room where it was captured. Figures 6, 7 and 8 show the confusion matrices obtained by introducing the cloudy, night and sunny test datasets into the network. The separated final rows and columns summarize the information in the confusion matrix. First, the row summary displays the numbers of correctly and incorrectly classified observations for each true class. Second, the column summary displays the number of correctly and incorrectly classified observations for each predicted class. For instance, regarding the confusion matrix related to the cloudy test dataset (Fig. 6), 1178 images were correctly predicted as corridor and 14 images were incorrectly predicted as corridor (see row summary): 3 images from the print area and 11 images from the stairs area. Additionally, the images captured from the corridor were 1178 times correctly predicted and 4 times incorrectly predicted (see column summary). Among them, 2 images were wrongly predicted in the kitchen, 1 image in the office-2P 1 and 1 image in the office-2P 2.

From these figures, we can analyze that the few wrong classifications are produced with wrong rooms which are adjacent and visually similar to the correct one. For instance, in cloudy, in the case of the images that belong to the 2-persons office 2 and were wrongly classified, the mistaken room was the contiguous and similar 1-person office room. Additionally, more mistakes can be noticed when the evaluated images were captured under changes of illumination (night and sunny). For example, under dark illumination conditions (night dataset), the stair area is wrongly predicted 47 times, 15 and 29 times are corridor and bathroom respectively, which are rooms adjacent and similar. But 3 times is retrieved the printer area. Regarding the results with the sunny illumination conditions, the wrong classifications between the 2-person office 2 and 1-person office room is increased.

Confusion Matrix. Cloudy

True Class	1. Print Area	282	3							282	3
	2. Corridor		1178	2		1	1			1178	4
	3. Kitchen			229						229	
	4. Large Office				132					132	
	5. Office-2P 1					233				233	
	6. Office-2P 2						148	10		148	10
	7. Office-1P							218		218	
	8. Bathroom								190	190	
	9. Stairs Area		11						18	122	29
		282	1178	229	132	233	148	218	190	122	
			14	2		1	1	10	18		
		1. Print Area	2. Corridor	3. Kitchen	4. Large Office	5. Office-2P 1	6. Office-2P 2	7. Office-1P	8. Bathroom	9. Stairs Area	
Predicted Class											

Fig. 6 Confusion matrix obtained after solving the first step of the hierarchical localization (room retrieval) with all the cloudy test images

Confusion Matrix. Night									
True Class	1. Print Area	287	26						
	2. Corridor	5	1096	2	3	2	2		
	3. Kitchen		2	270					
	4. Large Office				143				
	5. Office-2P 1		2			294			
	6. Office-2P 2		1				120	6	
	7. Office-1P						2	136	
	8. Bathroom								257
	9. Stairs Area	3	15					29	173
		287	1096	270	143	294	120	136	257
	8	46	2	3	2	4	6	29	
Predicted Class									
	1. Print Area	2. Corridor	3. Kitchen	4. Large Office	5. Office-2P 1	6. Office-2P 2	7. Office-1P	8. Bathroom	9. Stairs Area

Fig. 7 Confusion matrix obtained after solving the first step of the hierarchical localization (room retrieval) with all the night test images

Confusion Matrix. Sunny										
True Class	1. Print Area	207								
	2. Corridor	17	947				1			
	3. Kitchen		8	165						
	4. Large Office		5		115					
	5. Office-2P 1		4	1		124	4	32		
	6. Office-2P 2		4				79	60		
	7. Office-1P							162		
	8. Bathroom								158	14
	9. Stairs Area		9							115
		207	947	165	115	124	79	162	158	115
	17	30	1			5	92		14	
	1. Print Area	2. Corridor	3. Kitchen	4. Large Office	5. Office-2P 1	6. Office-2P 2	7. Office-1P	8. Bathroom	9. Stairs Area	
Predicted Class										

Fig. 8 Confusion matrix obtained after solving the first step of the hierarchical localization (room retrieval) with all the sunny test images

Additionally, Fig. 9 shows two bar charts concerning the behaviour of the CNN when the estimations are correct or wrong. That is, they show the average likelihood of the evaluated images to belong to the room retrieved (the best option), the likelihood to belong to second best option and so forth. This information is calculated by the final layer of the CNN. As we can observe in Fig. 9 a, when the rooms of the images are correctly estimated, the correct option presents an average likelihood near to the 100% and the second best option presents an average likelihood of 1.09%. In contrast, the Fig. 9 b shows these average percentages when the retrieved room is not correct. In this case, we can appreciate a considerably lower likelihood for the best option (74.24%) and a higher likelihood for the second best option (22.5%). Therefore, from these graphs, we can conclude that the likelihoods calculated for a test image can be helpful to decide whether the classification was correct or wrong and also, which other rooms should be considered apart from the best option retrieved.

5.3 Experiment 2. Use of the CNN to obtain holistic descriptors for batch localization

This experiment presents an evaluation of the performance of the holistic descriptors obtained from different layers of the CNN for localization. The idea consists in introducing an image into the CNN and obtaining the global-appearance descriptor from the layers $conv_4$, $conv_5$, fc_6 , fc_7 and fc_8 (Fig. 1 shows a diagram of this process). First, these descriptors are used to build the visual model by calculating the holistic descriptor for each image contained in the training dataset $D = \{\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_{N_{train}}\}$. Afterwards, the localization is solved by using a nearest neighbour search, that is, a test image is captured (im_{test}), its holistic descriptor, $\mathbf{d}_{test}$, is obtained from a layer of the CNN; then, the descriptor is compared with the visual model D and the most similar descriptor (minimum cosine distance between them) $\mathbf{d}_k$ is retained, after that, the position of the captured image im_{test} is estimated as the position where im_k was captured. In this experiment, the cloudy test dataset is used to measure the effectiveness of the proposed

Fig. 9 Average likelihood provided by the CNN, used as classification tool (i.e., average likelihood that the retrieved room is correct, according to the CNN) when the classification is **a** correct or **b** wrong according to the ground truth

description methods. Additionally, the night and sunny datasets are used to evaluate the robustness of these descriptions against changes of illumination. Fig. 10 shows the results obtained after solving the batch localization with the test images, using the holistic descriptors obtained from the CNN. Additionally, for comparative purposes, this figure includes the results obtained with two hand-crafted descriptors (gist and HOG) which have been used in previous works to solve the image retrieval problem (Muriillo et al. 2012). Figure 10 shows the average localization error, calculated as the average Euclidean distance between the estimated position and the position provided by the ground truth of the dataset. Also, the average computing time is depicted. This value measures the time required to carry out the whole process: from calculating the holistic descriptor of the test image until estimating its position.

First, about the experiments without changes of illumination (using the cloudy test dataset), regarding the localization error, the descriptor obtained from the layer $conv_4$ presents the minimum error (5.07 cm), followed by the descriptors from the layers $conv_5$ and fc_6 (5.09 cm for both cases). As for the computing time, the fastest option is also achieved with the $conv_4$ layer (6.7 ms), since the global-appearance descriptor obtained from this layer has a relatively small size (180 components) and the data obtained for this layer are calculated in an early stage of the CNN architecture. Comparing the holistic descriptors obtained from the CNN with the classic descriptors, the conclusion is that the descriptors obtained with the CNN improve the localization task both considering accuracy and computing time.

Fig. 10 Visual localization solved by means of nearest neighbour search considering different holistic descriptors: CNN based descriptors ($conv_4$, $conv_5$, fc_6 , fc_7 and fc_8) and hand-crafted descriptors (gist and HOG). The efficiency is measured through the average localization error (cm) and also the average computing time (ms) required to calculate and estimate the position where the images were captured

Table 3 Visual localization solved by means of nearest neighbour search in Saarbrücken. The holistic descriptors used are obtained either from the Freiburg CNN trained in this work ($conv_4$, $conv_5$, fc_6 and fc_7), from the AlexNet ($conv_4$ and fc_6), or by using a classic hand-crafted descriptor ($gist$). The efficiency is measured through the average localization error (cm) and also the average computing time (ms) required to calculate and estimate the position where the images were captured

Descriptor	Avg. error (cm)	Avg. computing time (ms)
$conv_4$ (Freib-CNN)	7.33 ± 0.29	11.07
$conv_5$ (Freib-CNN)	15.82 ± 0.30	12.14
fc_6 (Freib-CNN)	7.49 ± 0.31	58.61
fc_7 (Freib-CNN)	7.67 ± 0.34	61.15
$conv_4$ (AlexNet)	7.79 ± 0.36	11.28
fc_6 (AlexNet)	7.28 ± 0.28	56.87
$gist$	7.28 ± 0.59	10.72

As for the results obtained with changes of illumination (using night and sunny datasets), as noticed in previous works (Cebollada et al. 2019b), this effect worsens the localization task. As shown in Fig. 10, in all the cases, the average localization error increases in comparison to the values obtained when no changes of illumination are considered (using the test cloudy dataset). In general, sunny illumination conditions affect more negatively the localization method proposed. Additionally, $conv_5$ and fc_8 are the layers of the CNN whose global-appearance descriptors are more affected. The most robust descriptors against changes of illumination are those generated by the layers fc_6 and fc_7 . For example, regarding the holistic descriptor obtained from the fc_6 layer, the Fig. 10 shows that the average localization error increases from 5.09 cm (without changes of illumination) to 28.80 and 38.94 cm with night and sunny illumination conditions respectively. Notwithstanding that, the descriptor provided by the layers fc_6 and fc_7 perform substantially more accurately than the classical analytic methods under changes of lighting conditions.

In general, considering the localization error and the computing time, either layers $conv_4$, $conv_5$, fc_6 or fc_7 can be considered to carry out this task. The descriptors obtained from the layers $conv_4$ and $conv_5$ are appropriate if no changes of illumination are expected, because they work relatively fast (9.07 ms and 10.7 ms respectively). On the contrary, the descriptors d_{fc6} and d_{fc7} are suitable if there are changes of illumination at the expense of a lightly higher computing time. The descriptor obtained from the layer fc_8 works relatively fast (19.34 ms), but the localization errors obtained are substantially worse comparing to the rest of descriptors evaluated.

After evaluating the use of the CNN to generate global-appearance descriptors, this work also aims to evaluate the use of this network with images that are captured from a different environment for mapping and localization. The idea is to check whether the CNN developed and trained with images from a specific environment can generalize and generate robust holistic descriptors for images captured in other environments different from the one used for training. Therefore, an experiment is carried out, in this case, using the images from the Saarbrücken environment as test images. Again, average localization error and average computing time are collected for different description methods: four different layers of the Freiburg CNN proposed in this work, the $gist$ descriptor and a descriptor based on the layers $conv_4$ and fc_6 of the original AlexNet network (without training nor replacing layers). The Table 3 shows the results for localization with the Saarbrücken dataset by

using the proposed holistic descriptors. As it is observed, most of the descriptors based on the Freiburg CNN are still relatively accurate. To illustrate one example, the performance of $\mathbf{d}_{conv_4}$ (Freib-CNN) is similar to $\mathbf{d}_{gist}$ and $\mathbf{d}_{fc_6}$ (AlexNet) and the calculation time is similar. Therefore, the conclusion achieved through this experiment is that obtaining global-appearance descriptors from the trained CNN is a relatively good method and it is generalizable to other environments different from the one which is used for training.

5.4 Experiment 3. Use of the CNN to tackle hierarchical localization

In the previous subsection, several global-appearance descriptors were evaluated to tackle the batch localization task through an image retrieval problem and globally, comparing the test image with all the images of the training set. This subsection focuses on evaluating the complete use of the CNN to carry out the hierarchical localization. In this way, the CNN is not only used to generate holistic descriptors, but also to retrieve the most probable room within the environment where the test image was captured. As it was explained in sect. 4, the hierarchical localization task proposed consists in: first (rough localization step), the test image is introduced to the CNN and it retrieves the most likely room where the image was captured by the robot. Second, a holistic descriptor is obtained from one of the layers of the CNN and this information is used to carry out the fine localization step by conducting a nearest neighbour search between the holistic descriptor of the test image and the holistic descriptors of the training images which belong to the retrieved room (see Fig. 4).

With the objective of comparing this localization method with the method proposed in the subsection 5.3, the evaluation is the same, that is, we obtain the average localization error and the average computing time to carry out the hierarchical localization process. Figure 11 shows the results obtained through the hierarchical localization proposed in the present paper, considering different intermediate layers of the CNN. Additionally, for comparative purposes, this figure also includes the results obtained with a previous approach (Cebollada et al. 2019a) which used hand-crafted features (either HOG or gist) along with a spectral clustering algorithm to create the high-level map of the hierarchical model. These comparative results are presented in the two last groups of columns of Fig. 11 (gist and HOG). Overall, the descriptors based on the CNN perform better than the method based on hand-crafted descriptors (Cebollada et al. 2019b). Comparing the descriptors based on CNN layers and the hand-crafted ones, the localization error with CNN descriptors is considerably lower. This improvement is noticed independently of the illumination conditions. Additionally, the computing time required to solve the localization is also lower using the CNN based descriptors.

Comparing the results obtained by applying batch localization and hierarchical localization, the hierarchical localization introduces a lightly higher localization error and dispersion of the results. This is given in all the descriptors evaluated and is due to failures produced in the rough localization step. Nevertheless, if we focus on the results obtained by using the descriptors $\mathbf{d}_{fc_6}$ and $\mathbf{d}_{fc_7}$, they both present a robust behaviour, since their results keep the localization error obtained through batch localization and at the same time, the computing time is substantially decreased. This behaviour is presented for the three illumination conditions evaluated.

As for the localization error increase produced by the CNN wrong classifications, we can check that the CNN is properly trained, since it retrieves the room successfully the 98% of the cases. Nevertheless, observing the graphs from the Fig. 9, extra information from the output layer of the CNN can be used to improve this method. These

Fig. 11 Results of the hierarchical localization proposed in this paper, based on CNN. The rough step is solved by estimating the most likely room with the CNN. The fine step is solved by means of nearest neighbour search within the retrieved room by using holistic descriptors (horizontal axis). The efficiency is measured through the average localization error (cm) and also the average computing time (ms) required to calculate and estimate the position where the images were captured

graphs show a considerably different behaviour of the likelihoods when the CNN succeeds or fails. When the correct rooms is retrieved, the most probable room presents an average likelihood around 98% and the rest of options are under the 2%, whereas when the CNN retrieves a wrong room, the most probable room presents an average likelihood of 74.24% and the following two most likely options are substantially over 2%. Therefore, departing from this analysis, the present work also proposes a novel hierarchical localization method also based on the CNN to solve the rough localization step but considering a threshold value to decide how many rooms are considered in the fine localization step. The whole method consists in the following steps. First, the test image is introduced into the CNN. The classification layer outputs 9 likelihoods related to the nine possible rooms. If the likelihood of the most probable room is higher than the threshold 1, th_1 , this room is retrieved; else, all the rooms whose likelihood is higher than the threshold 2, th_2 are retrieved. Afterwards, the fine localization is carried out again through a nearest neighbour search by comparing the holistic descriptor of the test image (obtained from a layer of the CNN) with the set of training descriptors contained in the retrieved rooms.

Therefore, through this new method, the hierarchical localization is carried out with all the test images and the results are presented in Fig. 12. For this experiment, only $conv_4$, $conv_5$, fc_6 and fc_7 were evaluated, since fc_8 has proved in previous experiments not to be suitable to generate a holistic descriptor that characterizes the images. The thresholds values were tuned to $th_1 = 0.8$ and $th_2 = 0.1$. In this figure we can observe that for all the cases, the average computing time increases with respect to Fig. 11. This

Fig. 12 Results of the hierarchical localization by using likelihood thresholds. The efficiency is measured by the average localization error (cm) and also the average computing time (ms) required to calculate and estimate the position where the images were captured

increase was expected, since this method leads to consider more instances in the fine localization step. Regarding the descriptors generated from the layers fc_6 and fc_7 , which were the cases that had a lower computing time in hierarchical localization, their related computing time is increased from 20.84 and 11.23 to 27 and 27.1 ms respectively. However, the localization process is still substantially faster than the obtained with the batch localization method based on a simple nearest neighbour search (Fig. 10), because this method takes an average computing time of 47.55 and 49.26 ms respectively. Comparing to the hierarchical method, this new method outputs a substantially lower error (mainly when night and sunny conditions are considered) and the dispersion of the results is also lower.

Tables 4, 5 and 6 show the average localization errors and standard deviation calculated with each method, respectively, under cloudy, night and sunny conditions. These tables confirm the fact that the hierarchical method with thresholds improves both the localization error and the dispersion of the results with respect to the pure hierarchical method. To illustrate one example, in the case of the holistic descriptor generated by the layer fc_6 , the average localization error is reduced by using the hierarchical localization with thresholds (from 5.23; 32.09 and 51.71 cm to 5.13; 25.53 and 38.10 cm respectively for the cloudy, night and sunny conditions). Hence, through the results reached from this experiment, the conclusion is that this novel method proposed to carry out the hierarchical localization task with thresholds is a competitive option regarding localization error and computing time.

Finally, concerning the likelihood thresholds th_1 and th_2 , a sensitivity analysis is performed. Figure 13 shows the results regarding (a) the percentage of times that the correct room is one of the rooms retrieved in the first step of the hierarchical localization (room retrieval). (b) Results regarding the average computing time to tackle the

Table 4 Summary of the average localization errors (in cm) output by the three proposed methods under cloudy illumination conditions

		Method		
		Batch	Hierarchical	Hierarchical with threshold
Layer / descriptor	conv ₄	5.07 ± 3.02	5.22 ± 0.04	5.10 ± 2.26
	conv ₅	5.09 ± 1.7	5.22 ± 7.34	5.51 ± 1.94
	fc ₆	5.09 ± 0.04	5.23 ± 0.21	5.13 ± 0.16
	fc ₇	5.14 ± 0.04	5.28 ± 0.04	5.18 ± 0.08
	fc ₈	16.6 ± 3.68	16.96 ± 7.54	–
	<i>gist</i>	5.19 ± 1.98	7.51 ± 2.22	–
	HOG	16.34 ± 4.38	30.98 ± 6.31	–

Table 5 Summary of the average localization errors (in cm) output by the three proposed methods under night illumination conditions

		Method		
		Batch	Hierarchical	Hierarchical with threshold
Layer / descriptor	conv ₄	65.23 ± 23.14	33.67 ± 16.36	28.87 ± 23.59
	conv ₅	43.97 ± 16.06	82.23 ± 40.01	36.60 ± 15.34
	fc ₆	28.80 ± 3.76	32.09 ± 15.16	27.53 ± 7.84
	fc ₇	28.24 ± 2.28	33.89 ± 15.55	28.29 ± 2.72
	fc ₈	82.59 ± 25.2	86.61 ± 40.01	–
	<i>gist</i>	106.50 ± 23.82	130.48 ± 24.59	–
	HOG	45.09 ± 22.18	95.27 ± 24.19	–

Table 6 Summary of the average localization errors (in cm) output by the three proposed methods under sunny illumination conditions

		Method		
		Batch	Hierarchical	Hierarchical with threshold
Layer / descriptor	conv ₄	55.54 ± 5.81	53.08 ± 25.03	37.40 ± 8.75
	conv ₅	116.81 ± 11.27	120.40 ± 25.16	54.01 ± 9.65
	fc ₆	38.94 ± 7.15	51.71 ± 25.33	38.10 ± 5.52
	fc ₇	42.45 ± 6.77	55.55 ± 27.70	42.77 ± 4.91
	fc ₈	137.39 ± 25.11	150.54 ± 33.93	–
	<i>gist</i>	88.39 ± 8.79	123.27 ± 11.38	–
	HOG	81.98 ± 6.29	156.59 ± 9.14	–

localization task. Figure 13a, the conclusion reached is that the room retrieval tends to offer better results as th_1 increases. Moreover, the percentage of times that the correct room is retrieved increases for lower values of th_2 . About the computing time (Fig. 13b), the values are similar regardless the thresholds values, with the exception of $th_1 = 30\%$ and $th_2 = 5\%$. For this case, the average computing time is higher. This is due to the fact that a significant number of rooms is considered to address the fine localization step. On

Fig. 13 Influence of the thresholds th_1 and th_2 in the room retrieval problem. **a** The vertical axis shows the percentage of times that the correct room is one of the rooms retrieved in the first step of the hierarchical localization (room retrieval). **b** Computing time (ms). th_1 defines whether only the most probable room should be considered or not. th_2 defines which rooms should be retrieved according to their related likelihood

the contrary, the computing time is reduced when $th_1 > 90\%$ (fewer rooms considered in the fine localization step).

6 Conclusion

Throughout the present work, we have evaluated the use of a deep learning technique to build hierarchical topological models for localization. The developed tool is a convolutional neural network trained for room retrieval purposes. In this sense, the network receives a panoramic image as input and it retrieves the most likely room where the image was captured. Additionally, this CNN is not only proposed to estimate rooms, but also to obtain holistic descriptors from its intermediate layers to characterize the information of the input image. Hence, the present work evaluates the use of this technique to solve the localization by means of three different methods: an image retrieval task (batch localization), a hierarchical localization based on different levels of accuracy and a hierarchical localization method with thresholds to decide which rooms are used in the fine localization step.

The training of the CNN, as well as the experiments were carried out with indoor datasets that contain omnidirectional images and present dynamic changes and blur effects. The datasets also provide images captured under different illumination conditions (during cloudy days, during sunny days and at night). Additionally, a data augmentation technique is proposed to supply a larger visual dataset to train more robustly the CNN. This technique is also used to add adverse visual effects to the dataset used to test the accuracy of the CNN developed. Regarding the CNN design, the network inherits the architecture from AlexNet and changes the initial and the final set of layers. Then, it is re-trained with the panoramic images obtained from the dataset.

Throughout this paper, several studies have been tackled. First, the CNN classifiers have been validated as a technique to perform the rough step of a hierarchical localization process. Additionally, the behaviour of the classification layer provides information that can be useful to detect wrong estimations. Second, the holistic descriptors obtained from the intermediate layers $conv_4$, $conv_5$, fc_6 and fc_7 are more suitable to solve the localization task than the classic descriptors *gist* and HOG. Moreover, fc_6 and fc_7 produce global-appearance descriptors which prove to be quite robust against changes of illumination. Also, the descriptors obtained from the CNN are also suitable to solve visual localization in other different environments, but they do not improve substantially the results output by a descriptor obtained from other pre-trained CNNs such as AlexNet. Third, the hierarchical localization based on the proposed CNN produces more efficient results regarding localization error and computing time than hierarchical methods based on classical descriptors and image retrieval. Additionally, considering the likelihood information provided by the classification layer of the CNN, the proposed method produces competent localization solutions. Figure 14 shows a bird's eye view of the ground truth of the test images and the estimated position, considering the three evaluated methods based on CNN and using the holistic descriptor generated by the layer fc_6 . Figure 14a shows the estimation by using batch localization, Fig. 14b shows the estimation using hierarchical localization and Fig. 14c shows the estimation when thresholds are applied to the hierarchical localization method. Moreover, for comparative purposes, Fig. 15 shows summarizes the average localization error and the average computing time of each method, when the descriptor of the layer fc_6 is used to solve the fine localization step. From these images, we conclude that hierarchical localization based on CNN keeps the precision of batch localization, but this method is substantially

Fig. 14 Comparison between poses estimation in the Freiburg cloudy test dataset by means of the different localization methods proposed in the present work: **a** Batch localization, **b** hierarchical localization and **c** hierarchical localization with thresholds. Both the ground truth and the estimated position of the test images are shown. These examples are based on the holistic descriptor generated by the layer fc_6 of the Freiburg CNN

faster. The use of thresholds is useful to keep a good accuracy even in presence of substantial changes in the lighting conditions.

Future works will focus on developing a CNN directly based on raw omnidirectional images, as captured by the catadioptric vision system. Furthermore, we will also

Fig. 15 Comparison between localization methods: average localization error and average computing time are presented for the three methods proposed using the holistic descriptor generated by the layer fc_6 of the CNN

develop a regression convolutional neural network that is able to estimate directly the position where the input images were captured.

Acknowledgements This work has been supported by the Spanish government through the project PID2020-116418RB-I00: “HYREBOT: Robots híbridos y reconstrucción multisensorial para aplicaciones en estructuras reticulares.”.

Funding Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature.

Declarations

Conflicts of interest The authors declare that they have no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

Abadi MHB, Oskoei MA, Fakharian A (2015) Mobile robot navigation using sonar vision algorithm applied to omnidirectional vision. In: 2015 AI and Robotics (IRANOPEN), IEEE, pp 1–6

- Amorós F, Payá L, Marín JM, Reinoso O (2018) Trajectory estimation and optimization through loop closure detection, using omnidirectional imaging and global-appearance descriptors. *Exp Syst Appl* 102:273–290
- Arroyo R, Alcantarilla PF, Bergasa LM, 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), pp 4656–4663, <https://doi.org/10.1109/IROS.2016.7759685>
- Cebollada S, Payá L, Mayol W, Reinoso O (2019a) Evaluation of clustering methods in compression of topological models and visual place recognition using global appearance descriptors. *Appl Sci* 9(3):377
- Cebollada S, Payá L, Román V, Reinoso O (2019b) Hierarchical localization in topological models under varying illumination using holistic visual descriptors. *IEEE Access* 7:49580–49595. <https://doi.org/10.1109/ACCESS.2019.2910581>
- Cebollada S, Payá L, Valiente D, Jiang X, Reinoso O (2019c) An evaluation between global appearance descriptors based on analytic methods and deep learning techniques for localization in autonomous mobile robots. In: ICINCO 2019, 16th International Conference on Informatics in Control, Automation and Robotics (Prague, Czech Republic, 29–31 July, 2019), Ed. INSTICC, pp 284–291
- Cebollada S, Payá L, Flores M, Román V, Peidró A, 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 (Liesusaint-Paris, France, 7–9 July, 2020), Ed. INSTICC
- Cebollada S, Payá L, Flores M, Peidró A, Reinoso O (2021) A state-of-the-art review on mobile robotics tasks using artificial intelligence and visual data. *Exp Syst Appl* 167:114195. <https://doi.org/10.1016/j.eswa.2020.114195>
- Çevik N, Çevik T (2019) A novel high-performance holistic descriptor for face retrieval. *Pattern Analysis and Applications* pp 1–13
- Chaves D, Ruiz-Sarmiento J, Petkov N, 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, Springer, pp 313–324
- Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Diego, USA. Vol. II, pp. 886–893
- Ding J, Chen B, Liu H, Huang M (2016) Convolutional neural network with data augmentation for sar target recognition. *IEEE Geosci Remote Sens Lett* 13(3):364–368
- Dong-Won S, ESUK Yo-Sung H (2019) Loop closure detection in simultaneous localization and mapping using descriptor from generative adversarial network. *Journal of Electronic Imaging* 28(1):1–13 – 13, <https://doi.org/10.1117/1.JEI.28.1.013014>
- Dymczyk M, Gilitschenski I, Nieto J, Lynen S, Zeisl B, Siegwart R (2018) Landmarkboost: Efficient visualcontext classifiers for robust localization. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp 677–684, <https://doi.org/10.1109/IROS.2018.8594100>
- Gonzalez R, Apostolopoulos D, Iagnemma K (2018) Slippage and immobilization detection for planetary exploration rovers via machine learning and proprioceptive sensing. *J Field Robot* 35(2):231–247
- Goodfellow I, Bengio Y, Courville A (2016) Deep learning. MIT press
- Guo J, Gould S (2015) Deep cnn ensemble with data augmentation for object detection. *arXiv preprint arXiv:150607224*
- Han D, Liu Q, Fan W (2018) A new image classification method using cnn transfer learning and web data augmentation. *Exp Syst Appl* 95:43–56
- He K, Lu Y, Sclaroff S (2018) Local descriptors optimized for average precision. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
- Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
- Lenz I, Lee H, Saxena A (2015) Deep learning for detecting robotic grasps. *Int J Robot Res* 34(4–5):705–724
- Levine S, Pastor P, Krizhevsky A, Ibarz J, Quillen D (2018) Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *Int J Robot Res* 37(4–5):421–436
- Li S, Chou L, Chang T, Yang C, Chang Y (2019) Obstacle avoidance of mobile robot based on hyper-omni vision. *Sens Mater* 31(3):1021–1036
- Liu R, Zhang J, Yin K, Pan Z, Lin R, Chen S (2018) Absolute orientation and localization estimation from an omnidirectional image. In: Pacific Rim International Conference on Artificial Intelligence, Springer, pp 309–316
- Luo Z, Shen T, Zhou L, Zhu S, Zhang R, Yao Y, Fang T, Quan L (2018) Geodesc: Learning local descriptors by integrating geometry constraints. In: The European Conference on Computer Vision (ECCV)
- Mancini M, Bulò SR, Ricci E, Caputo B (2017) Learning deep nbnn representations for robust place categorization. *IEEE Robot Automation Lett* 2(3):1794–1801

- Meattini R, Benatti S, Scarcia U, De Gregorio D, Benini L, Melchiorri C (2018) An semg-based human-robot interface for robotic hands using machine learning and synergies. *IEEE Trans Component Packag Manufact Technol* 8(7):1149–1158. <https://doi.org/10.1109/TCPMT.2018.2799987>
- Murillo AC, Singh G, Kosecka J, Guerrero JJ (2012) Localization in urban environments using a panoramic gist descriptor. *IEEE Trans Robot* 29(1):146–160
- Oliva A, Torralba A (2006) Building the gist of a scene: the role of global image features in recognition. In: *Progress in Brain Research: Special Issue on Visual Perception*. Vol. 155
- Payá L, Reinoso O, Berenguer Y, Úbeda D (2016) Using omnidirectional vision to create a model of the environment: A comparative evaluation of global-appearance descriptors. *Journal of Sensors*
- Payá L, Peidró A, Amorós F, Valiente D, Reinoso O (2018) Modeling environments hierarchically with omnidirectional imaging and global-appearance descriptors. *Remote Sens* 10(4):522
- Perez L, Wang J (2017) The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:171204621*
- Pronobis A, Caputo B (2009) COLD: COsY Localization Database. *The International Journal of Robotics Research (IJRR)* 28(5):588–594 <https://doi.org/10.1177/0278364909103912>, <http://www.pronobis.pro/publications/pronobis2009ijrr>
- Salamon J, Bello JP (2017) Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal Process Lett* 24(3):279–283. <https://doi.org/10.1109/LSP.2017.2657381>
- Shorten C, Khoshgoftaar TM (2019) A survey on image data augmentation for deep learning. *J Big Data* 6(1):60
- Shvets AA, Rakhlin A, Kalinin AA, Iglovikov VI (2018) Automatic instrument segmentation in robot-assisted surgery using deep learning. In: *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp 624–628. <https://doi.org/10.1109/ICMLA.2018.00100>
- Sinha H, Patrikar J, Dhekane EG, Pandey G, Kothari M (2018) Convolutional neural network based sensors for mobile robot relocation. In: *2018 23rd International Conference on Methods Models in Automation Robotics (MMAR)*, pp 774–779. <https://doi.org/10.1109/MMAR.2018.8485921>
- Valiente D, Payá L, Jiménez LM, Sebastián JM, Reinoso Ó (2018) Visual information fusion through bayesian inference for adaptive probability-oriented feature matching. *Sensors* 18(7):2041
- Voulodimos A, Doulamis N, Doulamis A (2018) Protopapadakis E (2018) Deep learning for computer vision: A brief review. *Computational intelligence and neuroscience*
- Wozniak P, Afrisal H, Esparza RG, Kwolek B (2018) Scene recognition for indoor localization of mobile robots using deep cnn. In: *International Conference on Computer Vision and Graphics*, Springer, pp 137–147
- Zhou B, Lapedriza A, Xiao J, Torralba A, Oliva A (2014) Learning deep features for scene recognition using places database. In: *Advances in neural information processing systems*, pp 487–495
- Zhu Y, Mottaghi R, Kolve E, Lim JJ, Gupta A, Fei-Fei L, Farhadi A (2017) Target-driven visual navigation in indoor scenes using deep reinforcement learning. In: *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp 3357–3364. <https://doi.org/10.1109/ICRA.2017.7989381>