IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 4, 2020, accepted December 23, 2020, date of publication January 4, 2021, date of current version January 13, 2021.

Digital Object Identifier 10.1109/ACCESS.2020.3048936

Transfer Learning for Humanoid Robot Appearance-Based Localization in a Visual Map

EMMANUEL OVALLE-MAGALLANES[1], NOÉ G. ALDANA-MURILLO[2], JUAN GABRIEL AVINA-CERVANTES[1], JOSE RUIZ-PINALES[1], (Member, IEEE), JONATHAN CEPEDA-NEGRETE[3], AND SERGIO LEDESMA[1]

[1]Telematics (CA) and Digital Signal Processing (CA) Groups, Engineering Division (DICIS), Campus Irapuato-Salamanca, University of Guanajuato, Salamanca 36885, Mexico
[2]Computer Science Department, Center for Research in Mathematics (CIMAT), Guanajuato 36000, Mexico
[3]Department of Agricultural Engineering, Division of Life Sciences, Campus Irapuato-Salamanca, University of Guanajuato, Irapuato 36500, Mexico

Corresponding author: Juan Gabriel Avina-Cervantes (avina@ugto.mx)

This work was supported in part by the University of Guanajuato Grant NUA 147347, and in part by the Mexican Council of Science and Technology CONACyT under Grant 626154/755609.

ABSTRACT Autonomous robot visual navigation is a fundamental locomotion task based on extracting relevant features from images taken from the surrounded environment to control an independent displacement. In the navigation, the use of a known visual map helps obtain an accurate localization, but in the absence of this map, a guided or free exploration pathway must be executed to obtain the images sequence representing the visual map. This paper presents an appearance-based localization method based on a visual map and an end-to-end Convolutional Neural Network (CNN). The CNN is initialized via transfer learning (trained using the ImageNet dataset), evaluating four state-of-the-art CNN architectures: VGG16, ResNet50, InceptionV3, and Xception. A typical pipeline for transfer learning includes changing the last layer to adapt the number of neurons according to the number of custom classes. In this work, the dense layers after the convolutional and pooling layers were substituted by a Global Average Pooling (GAP) layer, which is parameter-free. Additionally, an L_2-norm constraint was added to the GAP layer feature descriptors, restricting the features from lying on a fixed radius hypersphere. These different pre-trained configurations were analyzed and compared using two visual maps found in the CIMAT-NAO datasets consisting of 187 and 94 images, respectively. For evaluating the localization tasks, a set of 278 and 94 images were available for each visual map, respectively. The numerical results proved that by integrating the L_2-norm constraint in the training pipeline, the appearance-based localization performance is boosted. Specifically, the pre-trained VGG16 and Xception networks achieved the best localization results, reaching a top-3 accuracy of 90.70% and 93.62% for each dataset, respectively, overcoming the referenced approaches based on hand-crafted feature extractors.

INDEX TERMS Transfer learning, convolutional neural networks, robot localization, visual robot navigation.

I. INTRODUCTION

Autonomous navigation is a highly desired capability in mobile robotics because it allows moving from an initial position towards the desired target without external intervention in changing environments. For allowing a robot to evolve autonomously in a complex and dynamic environment, it needs to: develop a reliable perception system (based on intelligent sensors), build an appropriate representation of the environment (mapping), and learn to self-localize into the map [1]. Hence, a locomotion strategy and obstacle avoidance must be implemented in real-time to be sent to the robot moving control system as a conclusive experimental evaluation. Robot perception could be based on many available sensors. Nevertheless, vision systems are one of the most emblematic and rich sources of information, considering that many daily

The associate editor coordinating the review of this manuscript and approving it for publication was Shuihua Wang.

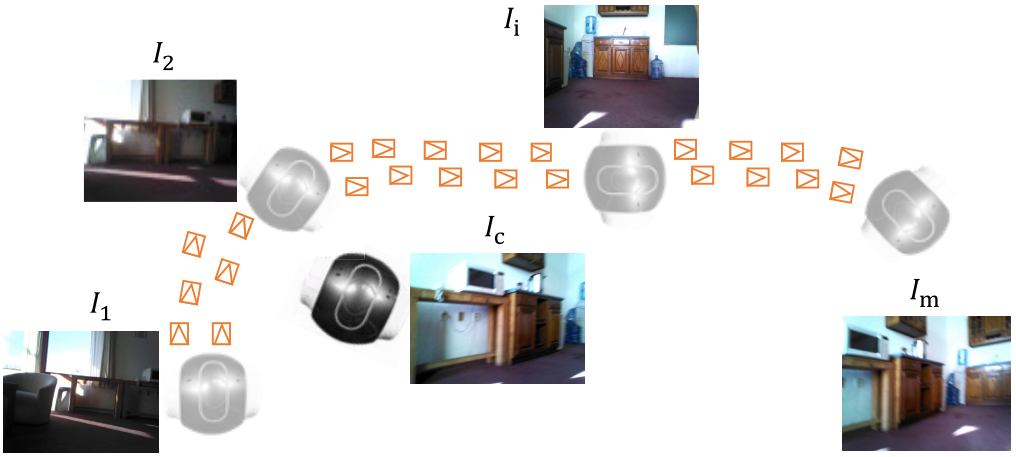


FIGURE 1. Humanoid robot appearance-based localization framework. The robot is initially kidnapped, and the method should give as an output the most similar key-image within the set of images in the visual map.

human activities depend highly on vision. Artificial vision systems are mainly classified into monocular or 2D systems using a single camera and 3D systems using a stereo-vision setup. Robot navigation based on 2D data provided by an on-board camera has increased the interest of the scientific community because of the relative hardware simplicity and dynamic source of information to obtain confident locomotion tasks [2]. A methodology widely studied in the context of wheeled mobile robots is the creation of a visual memory map for autonomous driving [3], [4]. Among the state-of-the-art robot navigation modalities, it is important to notice that the visual map-based navigation systems typically include four fundamental stages [3], [4]:

1) *A learning stage* which consists of constructing a visual map of an unknown environment using a subset of images (*i.e.*, key-images) taken previously by the robot during the human-guided navigation.
2) *A localization stage* which uses recognition algorithms to find the best correspondence between the most similar image in the visual map and the current robot view translated to real-world positioning coordinates.
3) *A visual route planning stage* which searches a set of discriminant images or landmarks on the visual map allowing the robot to reach a given spatial position.
4) *An autonomous navigation stage* which allows the robot to move to a particular location associated with the desired key-image by freely following the predefined visual path.

Such a methodology represents the whole environment as a collection of indexed images in a direct graph. While each node represents a specific location in the environment, each edge gives the associated weight complexity to move from one node to another. As earlier mentioned, the visual map approach was initially applied to the locomotion of wheeled mobile robots. Notwithstanding, this approach has been recently applied to implement advanced tasks on humanoid robots, such as a model predictive control scheme for visual walking pattern generation [5]. Humanoid robots include some non-linear issues on the dynamic model and sensors reading. For instance, during the robot displacements, the biped locomotion produces blurred images, and the sway motion induces image rotation around the camera optical axis. Such a methodology includes a visual map built by selecting a subset of images (key-images) from a learning sequence. This sequence is captured during the learning stage under two constraints. Firstly, two consecutive key-images must share enough visual information so that visual navigation between them could be computed. Secondly, the number of key-images should be compact and representative. Hence, the appearance-based localization paradigm consists of finding (*i.e.*, indexing) the most similar key-image regarding the currently observed image.

Appearance-based localization, derived from handcrafted approaches, relies on the number of features extracted and matched between the currently observed image and the key-images. However, in recent years, these methods have been technically surpassed by automatic feature extraction through Deep CNN. This last approach also has some limitations, requiring a vast number of training images. Besides, when using the SoftMax loss function, it cannot force features to have a higher discriminative power, not ensuring to keep positive pairs closer and negative pairs farther from each other.

In this work, an L_2-norm constraint was introduced on the features extracted by a set of pre-trained CNNs to tackle the problem of obtaining closer feature representation from two dissimilar images. Normalizing the features shows an improvement concerning the non-normalized features in the appearance-based localization in a visual map problem. This methodology was tested using the CIMAT-NAO datasets from the humanoid robot NAO [6]. Figure 1 shows the proposed approach to address the appearance-based localization in a visual map task, where a reliable feature descriptor is obtained from a pre-trained CNN. Consecutively, a normalization feature step is carried out before classification;

therefore, enforcing the features to lie into a hypersphere. Thus, the classification matches the currently observed image and a key-image of the visual map.

The contributions derived from this study are as follows: First, an L_2-constraint is incorporated during the training step. The approach intends to relax the features to lie into a hypersphere of fixed radius leading to higher discriminating power. Second, a Global Average Pooling layer (being parameter-free) was used instead of the fully connected layers at the top of the pre-trained network to reduce the number of trainable parameters. Third, each key-image of the visual map is passed through a data augmentation procedure to generate multiple images from each location, ensuring a balanced dataset and aiming that the neural network to learn different views of one key-image. Finally, the overall framework is efficiently applied to the visual localization of humanoid robots in indoor environments.

The paper is organized as follows. The related work is presented in Section II. The theoretical background is given in Section III. Subsequently, the proposed method is described in Section IV. The numerical results are shown and interpreted in Section V, and finally, the conclusions are given in Section VI.

II. RELATED WORK

An autonomous robot localization using only visual information, also known as an appearance-based localization system, is difficult to implement because of many factors such as illumination, perspective, and type of sensors [7]. Purely appearance-based approaches assume that the robot has no explicit or reliable position/odometry information (*i.e.*, GPS-less environment, unequal or slippery floor contact). Furthermore, this kind of task needs to face the *perceptual aliasing* problem, which happens when two dissimilar locations share a similar visual appearance. The practical objective of robot appearance-based localization is to determine the reference image (in a previously learned set of images), that is, the most similar in appearance to the currently captured image.

Most relevant visual information (features) needs to be extracted from datasets to determine the similarity between compared images. Previously works employed hand-crafted local feature extractors such as the Speeded-Up Robust Features (SURF) [8], Scale Invariant Feature Transform (SIFT), Binary Robust Independent Elementary Features (BRIEF) [9], or Oriented FAST and Rotated BRIEF (ORB) [10]. In indoor environments, Aldana *et al*. [11] propose a local descriptor based on BRIEF to deal with the humanoid robot locomotion issues: blurring and rotation around the optical axis. The authors solved an appearance-based localization problem using only visual information in the humanoid robot NAO.

Nowakowski *et al*. [12], introduced an indoor topological localization algorithm that uses visual and Wi-Fi information. They developed an algorithm that solves global localization and kidnapped robot problems by merging Wi-Fi information with the Fast Appearance-Based Mapping (FABMAP) algorithm [13]. The aim of combining Wi-Fi and FABMAP vocabularies is to deal with complicated cases where distinct locations have similar visual appearances. The resulting algorithm was tested with a Pepper robot data, a sociable humanoid type robot with omnidirectional wheels. The captured images from the robot have no remarkable orientation concerning the camera optical axis.

Deep CNN-based methods have recently been investigated to overcome inconveniences of classical hand-crafted image feature representations. For instance, Sunderhauf *et al*. [14] demonstrated the benefits of using a pre-trained AlexNet [15] architecture for visual place recognition tasks on robots evolving in outdoor environments. Place recognition is performed by a single-image Nearest Neighbor (1-NN) search that is based on the extracted cosine distance of the feature vectors. Li *et al*. [16] proposed a method to measure image similarity; their method is based on features extracted by the pre-trained AlexNet architecture. In such an approach, the image is divided into patches to obtain a global similarity matrix constructed according to the patch similarities. Lastly, an adaptive weighted scheme determines the overall image similarity. Moreover, Wozniak *et al*. [17] proposed a CNN-based algorithm for indoor place recognition. It uses transfer learning to retrain a VGG network [18] to classify places in images acquired by a humanoid robot navigating in different indoor environments. The network was fine-tuned using a dataset containing 8000 images recorded in sixteen rooms.

In this paper, the image feature representation for the appearance-based localization of a robot in a visual map was addressed using the most relevant and recent pre-trained CNN architectures. In particular, the VGG16 [18], ResNet50 [19], InceptionV3 [20], and Xception [21] architectures were analyzed. Additionally, an L_2-norm constraint was added to map the obtained features into a fixed radius hypersphere. Thus, all features have the same L_2 norm with a constant relaxing value given by the hypersphere radius. Furthermore, a data augmentation step was implemented before training because of the sparse representation of the visual map, where each key-image represents a node in the graph (*i.e.*, one location in the map). By doing so, each node is now represented by a full set of N augmented images. The top-k similar key-images are considered as possible match because they can handle the presence of close similar key-images within the visual map.

It is worthy to note that structurally, two consecutive key-images share visual information such that a control policy can be computed between them. The evaluation and comparison of the performance of different pre-trained CNN are analyzed using two datasets captured from an NAO humanoid robot concerning the visual map and query images.

The numerical results have shown that using an L_2-norm constraint (for the image feature representation) that has been extracted by a pre-trained CNN improves the localization performance. Furthermore, no dense layers after the convolutional and pooling layers were required during training

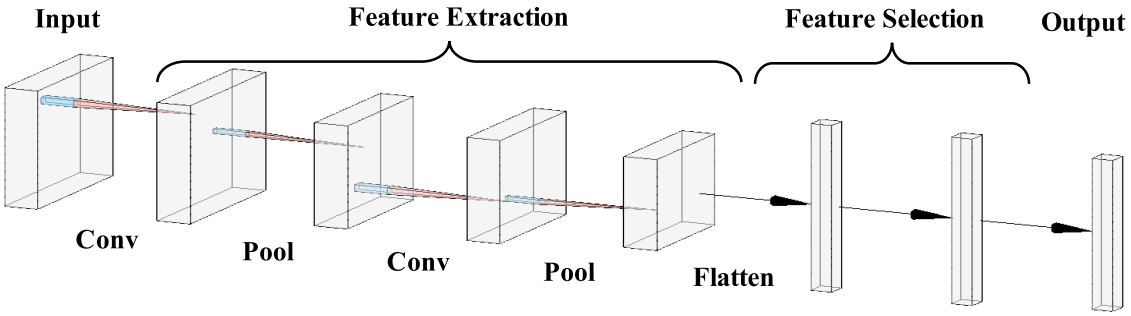


FIGURE 2. The architecture of a typical Convolutional Neural Network, consisting of three main types of layers: convolutional, pooling, and fully connected layers.

because they were substituted by a Global Average Pooling layer, which is parameter-free. Consequently, the number of hyperparameters is drastically reduced when compared with the number of free parameters that hand-crafted approaches require, *i.e.*, the size and number of local features, depth, and clusters.

III. THEORETICAL BACKGROUND

A. CONVOLUTIONAL NEURAL NETWORKS

Convolutional Neural Networks are a special type of neural networks that are used in deep learning. They are most commonly used to perform image classification tasks [15]. CNNs consist of a group of alternating convolutional and pooling layers attempting to extract discriminative features (*e.g.*, edges, interest points) across a large set of input images. The extracted features are, then passed through a set of fully connected layers to estimate the correct class for each input.

The *convolutional layer* uses K filters that perform convolution operations over the input data or image. Every filter is represented by a small spatial 2D matrix (*e.g.*, 3×3 size filter) extended through the full-depth of input data. All filters are convolved across the complete input image to produce a 2D activation map for each filter during the optimization process. The activation map or feature map gives the responses of a given filter at every spatial position. Each feature map is defined as follows:

$$\mathbf{O}_k = f\left(b_k + \sum_c \mathbf{W}_k[c] * I[c]\right), \tag{1}$$

where f is a non-linear activation function, b_k is the k-bias, $I[c]$ is the image at c-channel, $\mathbf{W}_k[c]$ is the k-filter, and $*$ denotes the convolution operation. Therefore, a CNN creates dynamic filter that allows the network to detect a specific or relevant type of visual features. Hitherto, the features extracted from low-level layers are more generic (*e.g.*, luminance, edges, contrasting colors, and curves) than those extracted by the top-layers.

The *pooling layer*, also known as the downsampling layer, is typically applied in-between successive convolution layers. The main purpose of the pooling layer is to reduce the spatial size of the feature maps. Furthermore, the pooling layer reduces the number of parameters in the network and controls overfitting [22]. The most common forms of pooling are the max-pooling and average-pooling. In max-pooling, the maximum value around a window is taken. On the other hand, average-pooling computes the mean value of the window. The pooling layer size is 3×3 pixels with a 2-pixel stride is common practice. It is noteworthy that a stride of 2 downsamples every feature map by 2 (along both width and height), discarding 75% of the activations.

Fully connected layers are usually found at the end of CNN architectures, which connect every neuron in the previous layer to every neuron on the next layer. The extracted features from previous layers are then used for classifying the input image into their corresponding class. Figure 2 illustrates a typical CNN architecture, consisting of several pairs of convolutional and pooling layers (feature extraction) followed by consecutive fully connected layers (feature selection). Finally, the last classification layer is used to generate the predicted class labels.

B. TRANSFER LEARNING

Generally, training a CNN from scratch (with random initialization), where the unknown network weights are updated in each epoch through backpropagation by minimizing a specific cost function, requires a large training dataset to achieve high accuracy. Thereunto, *Transfer Learning* (TL) [23] addresses this challenge by transferring the learned knowledge on a large dataset, such as ImageNet [24] that contains 1.2 million images with 1000 categories, to modestly related datasets. Hence, pre-trained CNNs can be used either as a weight initialization or as a fixed feature extractor for the task of interest.

In the first scenario, the pre-trained CNN is taken as a fixed feature extraction. Certainly, the features can be extracted from any layer of the CNN. A widespread practice is to change the number of neurons of the last fully connected layer to the number of classes in the problem and optimize the network with the pre-trained weights set into non-trainable during the optimization process. The second strategy fine-tunes the weights of the pre-trained network during the optimization of the analyzed task. The number of fine-tuned layers depends on the dissimilarity of the source and target dataset domains [25].

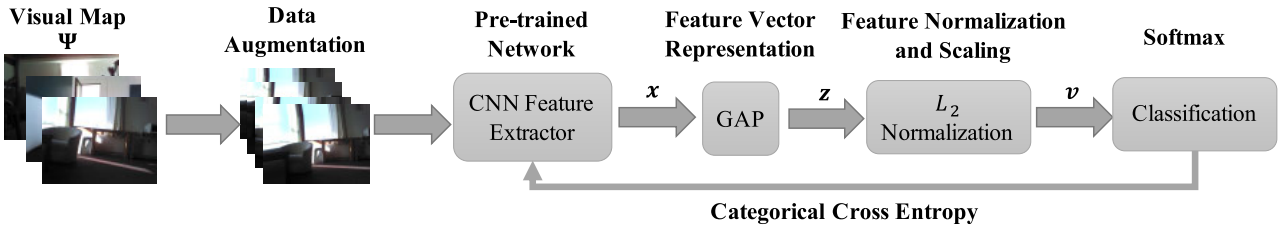


FIGURE 3. The pipeline of the proposed appearance-based localization in a visual map. A Global Average Pooling (GAP) was added to extract a feature descriptor from the pre-trained network outcome. Finally, an L2-norm normalization and scale layer is used to constrain the feature descriptors to lie on a hypersphere of radius α.

TABLE 1. Pre-trained CNN models overview. The Top-1 accuracy refers to the model performance on the ImageNet validation dataset.

Model	Top-1 Accuracy	Parameters
Xception	0.790	22,910,480
VGG16	0.713	138,357,544
ResNet50	0.749	25,636,712
InceptionV3	0.779	23,851,784

This paper evaluates four state-of-the-art network architectures pre-trained on the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) competition as a feature extractor. These networks exhibit an interesting trade-off between computational requirements and performances: VGG16, ResNet50, InceptionV3, and Xception. Table 1 shows a general overview of these last CNNs architectures. It is noteworthy that these networks employ a Rectified Linear Unit (ReLU) as an activation function and pooling layers with a stride of two pixels. The ReLU activation function is formally defined as

$$f(x) = \max(0, x), \tag{2}$$

where x is the outcome of the convolution operation described by (1).

IV. PROPOSED VISUAL LOCALIZATION METHOD

This paper proposes an end-to-end CNN architecture for the appearance-based localization on a visual map problem. The CNN was pre-trained via transfer learning. A global average pooling was added after the convolutional and pooling layers to obtain a feature vector descriptor for the given input image. Finally, an L_2-norm constraint is used to transform or project the feature descriptor into a hypersphere of a fixed radius α. This constraint was earlier introduced by Ranjan *et al.* [26] for discriminative face verification showing a significant boost in classification performance. The general pipeline of the proposed appearance-based localization in a visual map is shown in Figure 3.

A. FEATURE DESCRIPTOR EXTRACTION

Given a visual map defined by

$$\Psi = \{(\mathcal{I}_i, y_i),\ i \in \{1, 2, \cdots, m\}\}, \tag{3}$$

with m key-images corresponding to m locations (classes y_i), a pre-trained CNN is taken as a classification task keeping the convolutional and pooling layers non-trainable and adding a new classification layer at the top of the network. First, a data augmentation procedure was performed due to a sparse representation of each class, including rotation, zooming, and shifting. This last step generates N additional views $\hat{I}$ for each location concerning the visual map, such that the augmented data is given by

$$\mathcal{I}_i = \{\hat{I}_j,\ j \in \{1, 2, \cdots, N\}\}. \tag{4}$$

Secondly, the augmented images are re-scaled to match the input dimension of the pre-trained network. The height, width, and the number of channels of the input image denoted by $H_I \times W_I \times C_I$ are re-scaled to the network input size $H_N \times W_N \times C_N$. Later, the re-scaled images are forwarded through the pre-trained network $\mathcal{E}$.

The output feature maps $\mathbf{x}_l$ from an arbitrary layer l of dimensions $H_l \times W_l \times K_l$ is extracted, where K_l is the number of features maps, and H_l and W_l are their height and width, respectively. A Global Average Pooling (GAP) [27] to transform the feature maps into a reduced representation is carried out, yielding a feature vector descriptor $\mathbf{z} \in \mathbb{R}^{K_l}$. The GAP operation reduces the preceding layer size by taking each feature map average as follows

$$\mathbf{z} = \text{GAP}(\mathbf{x}_l) = \mathbf{avg}(x_k) = \bar{x}_k, \quad k \in K_l. \tag{5}$$

B. L2-SOFTMAX LOSS

Figure 3 shows a pre-trained CNN, in which given a training sample, a set of features are extracted through a set of convolutional and pooling layers. Subsequently, some linear layers and a classifier layer are used to determine whether the input image belongs to a specific class with an associated probability. As with any classical multi-class classification problem, the network weights are optimized by backpropagation by calculating the categorical cross-entropy loss, minimizing the corresponding loss function. Therefore, the loss function is given by

$$L = -\frac{1}{M}\sum_{i=1}^{M} \log \frac{\exp\left(\mathbf{W}_{y_i}^{\intercal}\mathbf{z}_i + b_{y_i}\right)}{\sum_{j=1}^{C} \exp\left(\mathbf{W}_j^{\intercal}\mathbf{z}_i + b_j\right)}, \tag{6}$$

where M is the training batch size, $\mathbf{z}_i$ is the corresponding i-features descriptor in the batch before the last

fully-connected layer, C is the number of classes, y_i is the corresponding class label, and $\mathbf{W}$ and b are the trainable weights and bias in the network, respectively. The SoftMax loss function (6) is biased to the sample distribution, not optimizing to obtain positive pairs closer and negative pairs far from each other. A feature normalization can be applied to solve these issues. By doing so, allows the features descriptors to lie on a hypersphere. In specific, a 2nd order normalization is based on the L_2-norm given by $\|\mathbf{z}\|_2 = \sqrt{\mathbf{z}^\intercal \mathbf{z}}$, which maps the features to lie in a unitary hypersphere. The L_2-SoftMax loss is given by

$$L = -\frac{1}{M}\sum_{i=1}^{M} \log \frac{\exp\left(\mathbf{W}_{y_i}^\intercal \mathbf{z}_i + b_{y_i}\right)}{\sum_{j=1}^{C} \exp\left(\mathbf{W}_j^\intercal \mathbf{z}_i + b_j\right)},$$
$$\text{s.t.} \quad \|\mathbf{z}_i\|_2 = \alpha, \quad \forall_i = 1, 2, \cdots, M, \tag{7}$$

where α is a scalar value for relaxing the radius of the hypersphere. As shown in Figure 3, the L_2-softmax loss relies on three steps. First, an L_2 normalize layer re-scales the input feature descriptor $\mathbf{z}$ to a unit vector as follows

$$\hat{\mathbf{z}} = \frac{\mathbf{z}}{\|\mathbf{z}\|_2}. \tag{8}$$

Then, the normalized feature descriptors are scaled to a fixed radius given by α such that the new vector is represented by

$$\mathbf{v} = \alpha \cdot \hat{\mathbf{z}}. \tag{9}$$

Finally, the L_2-SoftMax loss is minimized during the training process by computing the loss function gradient with respect to the current weights' values. This constraint introduces one scalar parameter (α) that is trained along with the other parameters of the network during backpropagation.

C. LOCALIZATION PROCESS

The localization process consists of determining a potential match between the current image scene I_c and previous image locations $\mathcal{I}_k \in \Psi$. The key-image and current image are symbolically matched according to their respective predicted class probabilities. As mentioned before, consecutive key-images in the visual map share similar visual information such that a control policy must be computed to reliably move the robot from one location to the other on the map. Therefore, the appearance-based localization task faces perceptual aliasing, where two separate locations share a similar visual appearance. Figure 4 illustrates a real example of this visual map constraint. To handle the presence of close, similar key-images within the visual map, a top-k accuracy was employed. The top-k accuracy computes how often targets are in the top-k predictions (the k-ones with the highest probabilities).

V. RESULTS AND DISCUSSION

In this section, a brief description of the public datasets used in the proposed method is first introduced. Second, the hyper-parameters employed to train the proposed CNN

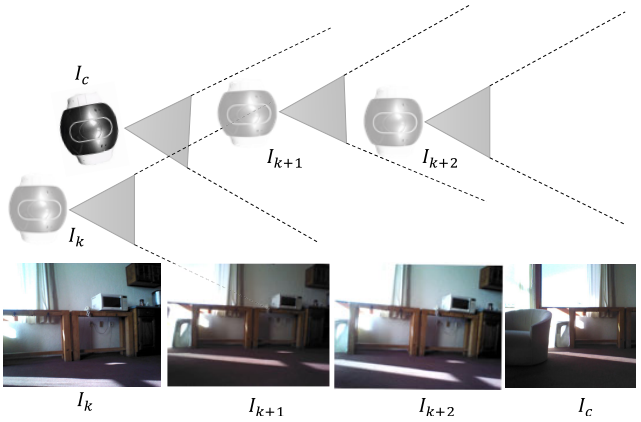
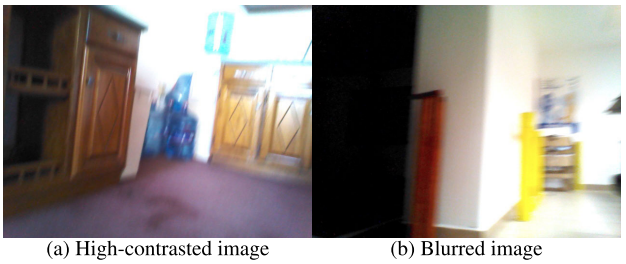


FIGURE 4. Consecutive key-images in the visual map. The key-images share a portion of the field of view of the previous key-image.



(a) High-contrasted image (b) Blurred image

FIGURE 5. Representative key-images in the visual map taken from the on-board camera in the robot NAO [6].

FIGURE 6. New image samples generated by data augmentation. Left image: original key-image.

is described. Finally, performance comparisons are carried out using different pre-trained CNNs and an L_2-norm constraint. The results are also compared with a baseline hand-crafted approach implemented for a humanoid robot. The computational experiments were performed on a Cloud Platform with an Intel(R) Xeon(R) CPU, 12 GB of RAM, and 2.00 GHz dual-processor. The GPU Platform was based on a Tesla P4 having 2560 CUDA cores and 8 GB VRAM. The experiments were conducted in Python 3.6, Keras 2.3.1, and TensorFlow 2.2.0. The code can be found on https://github.com/eovallemagallanes/Transfer-Learning-Localization.

A. DATASETS USED IN EVALUATION

The proposed appearance-based localization methodology was tested on two distinct datasets publicly available online as CIMAT-NAO [6]. These datasets correspond to indoor environments taken by an NAO humanoid robot. The first dataset (CIMAT-DATASET-A) contains 445 images of 640 × 480 pixels divided into two subsets: 187 images (hand-selected) as the visual map and 258 images for testing. The second dataset (CIMAT-DATASET-B) consists

FIGURE 7. Localization results using the CIMAT-NAO-A dataset. At left: Query image, next three images: Top-3 class probabilities. Under each retrieved key-image, the probability is shown. In a green box, the ground-truth key image is highlighted. The last row indicates when the top-3 retrieved key images are not the ground-truth localization key image.

of 188 images of 640 × 480 pixels, where 50% were considered for the visual map and the remainder for testing. Some dataset images are seriously affected by rotation, blur, and illumination changes that originated intrinsically by robot locomotion. Figure 5 shows two representative database elements exhibiting defects in quality and image artifacts. In the numerical experiments, pixels range were linearly transformed from $[0, 255]$ to $[0, 1]$. Besides, the images were downsampled to size 224 × 224 and 299 × 299 pixels to fit the original image dimensions in the pre-trained networks.

B. TRAINING PROCESS

The Adam optimization method was used to find the optimal solution during the training of the network [28]. The optimization was reinforced by minimizing the L_2-SoftMax loss function given by (7). Each architecture (*i.e.*, the four different pre-trained networks) was trained with similar tuning parameters, a learning rate of 0.01 for 100 epochs. Furthermore, the batch size was fixed to 128, and an early stopping strategy based on the validation loss was implemented to reduce overfitting risks. During the optimization, the pre-trained weights remain as non-trainable; consequently, their weights remain fixed. As mentioned earlier, each key-image of the visual map is passed through a data augmentation procedure to generate multiple images from one location (associated with a class). Specifically, for each key-image, a set of 30 images was generated; this process is shown in Figure 6. This step was obviously executed before training. Next, the newly generated images are only considered to create a training data set. Whilst the original key-images are used in the validation set. This data distribution aimed that the neural network learns different views (representations) of one key-image to simulate scenarios. Furthermore, a validation set is not provided in the original dataset because of dataset sparsity. Notwithstanding, a balanced dataset was ensured by using transfer learning and the proposed distribution.

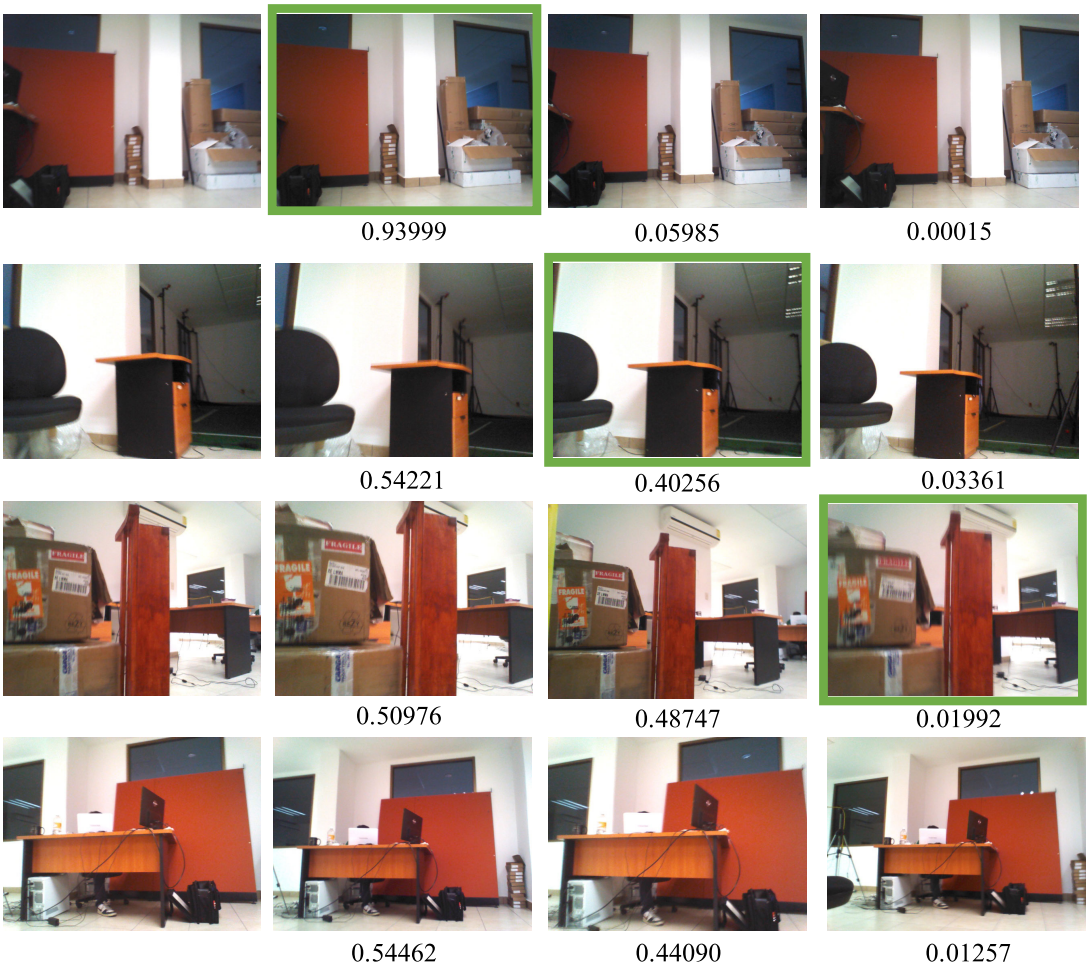


FIGURE 8. Localization results using the CIMAT-NAO-B Dataset. At left: Query image, next three images: Top-3 class probabilities. Under each retrieved key-image, the probability is shown. In a green box, the ground-truth key image is highlighted. The last row indicates when the top-3 retrieved key images are not the ground-truth localization key image.

C. PERFORMANCE COMPARISON

The proposed approach was designed to evaluate the appearance-based localization in a visual map using a pre-trained CNN as a feature extractor. Additionally, this study was performed to analyze the effect of an L_2-normalization applied in the extracted feature descriptor. For comparison purposes, the CNNs have been evaluated without using the L_2 constraint.

Additionally, a custom hand-crafted feature descriptor proposed for humanoid robot localization was taken into account as a baseline. It is critical to define a fair k value for the top-k accuracy measure. In this work, the top 3 probability predictions were considered to estimate reliable localization results. Table 2 summarizes the performance of the localization accuracy of each pre-trained CNNs with L_2-SoftMax loss and (traditional) SoftMax loss functions for CIMAT-NAO datasets. For the first dataset, the VGG16 with L_2-SoftMax obtained the best top-3 results with accuracy of 90.7%. On the other hand, the second dataset achieved accuracy of 93.62% employing an Xception network with L_2-SoftMax loss function. It can be noticed that the localization performance employing a pre-trained CNN overcame the hand-crafted reference approach (BRIEFROT [11]) that reached only

TABLE 2. Localization accuracy results. For each CNN architecture the top-3 accuracy is showed.

Dataset	Model	Loss/Distance	Accuracy (%)	α
CIMAT-NAO-A	BRIEFROT	χ^2	75.19	N/A
	Xception	SoftMax	82.95	N/A
		L_2-SoftMax	82.56	28.80
	VGG16	SoftMax	89.15	N/A
		L_2-SoftMax	**90.70**	**50.46**
	ResNet50	SoftMax	79.84	N/A
		L_2-SoftMax	76.74	52.99
	InceptionV3	SoftMax	82.17	N/A
		L_2-SoftMax	82.56	29.58
CIMAT-NAO-B	BRIEFROT	χ^2	86.17	N/A
	Xception	SoftMax	93.62	N/A
		L_2-SoftMax	**93.62**	**12.46**
	VGG16	SoftMax	87.23	N/A
		L_2-SoftMax	85.11	28.73
	ResNet50	SoftMax	88.30	N/A
		L_2-SoftMax	84.04	28.68
	InceptionV3	SoftMax	90.43	N/A
		L_2-SoftMax	90.43	14.09

accuracy of 75.19% and 86.17%, for each dataset, respectively. Furthermore, the L_2-norm constraint maps the features into a hyper-sphere, and therefore, boosts the accuracy for the VGG16 and InceptionV3 for the first dataset. In contrast, Xception and InceptionV3 (for the second dataset) reached the same accuracy when using the traditional SoftMax loss to compare the results. The scale parameter α reaches a range around (30, 50) and (12, 29) for each dataset, respectively.

Figure 7 shows several examples of the localization process for the CIMAT-NAO-A dataset using the top-3 predictions. Such results are generated by the VGG16 network with the L_2-SoftMax loss function. For each query image at the left, the best three predictions are shown at its right. The ground-truth localization image is surrounded by a green bounding box within the visual map. Similarly, Figure 8 illustrates the top-3 outcomes using the Xception network with the L_2-SoftMax loss and CIMAT-NAO-B dataset. Besides, the 3-one's images with the highest probabilities are retrieved for each query image.

In both cases, the top-3 key-images associated with a given query image are remarkably similar because they share visual information. Ergo, the key-images exhibit light changes in lighting, zooming, contrast, or blurring among them. A difficult example occurs when the ground-truth key-image does not belong to the top-3 retrieved key-images. Such a case is shown in Figures 7 and 8 (see the corresponding last rows). In the first scenario, the retrieved images only contain a small region in common. On the other hand, the key-images are very similar for the second dataset, but they hold different zooming scales. Despite those inconveniences, the localization approach detects key-images such that a visual control policy can be computed between them.

Some factors may influence localization performance, and like its handcrafted counterpart, the method found difficulties when key-images exhibit defects in quality and artifacts such as in high-contrasted and blurred images. Additionally, the generated images during the data augmentation procedure could also be affected by perceptual aliasing, leading to a similar feature representation.

However, the proposed method provides a solution to solve a limited amount of key-images within the visual map, such that a Deep CNN approach can be successfully applied. Moreover, the L_2-norm constraint included in the feature descriptors yields better accuracy, increasing by 15% and 7% of performance against the handcrafted approach for each dataset, respectively.

VI. CONCLUSION

In this paper, four different state-of-the-art CNN architectures were evaluated for the humanoid robot appearance-based localization problem. These architectures were initialized via transfer learning. Note that these networks were previously trained with the ImageNet dataset. For each pre-trained architecture, concerning the VGG16, ResNet50, InceptionV3, and Xception, the weights during the optimization process remain unchanged. The numerical results have demonstrated that employing a pre-trained network on a sparse visual map efficiently performs the appearance-based localization task. An end-to-end architecture that comprises the pre-trained convolutional and pooling layers, a global average pooling, and a SoftMax layer allowed obtaining an accuracy improvement regards the baseline localization approach using hand-crafted feature extractors. Furthermore, an L_2-norm constraint was included in the feature descriptors, yielding a boosted accuracy for the VGG16 and Xception architectures. In particular, numerical results using the first and second datasets with the proposed architecture reached a top-3 accuracy of 90.70% and 93.62%, respectively. According to the top-3 retrieved key-images visual examination, it was found that the presence of blur and drastic changes in perspective and zooming on the scene may affect the predicted most similar key-image. Finally, the appearance-based localization in a visual map employing transfer learning leads to strong improvement with respect to features extracted by traditional hand-crafted approaches.

FUNDING

The APC was funded by the University of Guanajuato.

CONFLICTS OF INTEREST

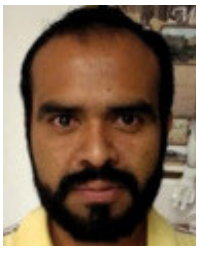
The authors declare there is no conflict of interest.

REFERENCES

[1] O. Stasse, "SLAM and vision-based humanoid navigation," in *Humanoid Robotics: A Reference*. Dordrecht, The Netherlands: Springer, 2018, pp. 1739–1761, doi: 10.1007/978-94-007-6046-2_59.
[2] H. M. Becerra, C. Sagüés, Y. Mezouar, and J.-B. Hayet, "Visual navigation of wheeled mobile robots using direct feedback of a geometric constraint," *Auto. Robots*, vol. 37, no. 2, pp. 137–156, Aug. 2014.
[3] A. Diosi, S. Segvic, A. Remazeilles, and F. Chaumette, "Experimental evaluation of autonomous driving based on visual memory and image-based visual servoing," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 3, pp. 870–883, Sep. 2011.
[4] J. Courbon, Y. Mezouar, and P. Martinet, "Autonomous navigation of vehicles from a visual memory using a generic camera model," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 3, pp. 392–402, Sep. 2009.
[5] N. G. Aldana-Murillo, L. Sandoval, J.-B. Hayet, C. Esteves, and H. M. Becerra, "Coupling humanoid walking pattern generation and visual constraint feedback for pose-regulation and visual path-following," *Robot. Auto. Syst.*, vol. 128, Jun. 2020, Art. no. 103497.
[6] N. G. Aldana-Murillo, J.-B. Hayet, and H. M. Becerra. (Apr. 2020). *CIMAT Nao Datasets*. [Online]. Available: http://personal.cimat.mx:8181/~hmbecerra/CimatDatasets.zip
[7] S. Lowry, N. Sunderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, "Visual place recognition: A survey," *IEEE Trans. Robot.*, vol. 32, no. 1, pp. 1–19, Feb. 2016.
[8] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Comput. Vis. Image Understand.*, vol. 110, no. 3, pp. 346–359, Jun. 2008.
[9] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: Binary robust independent elementary features," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 778–792.
[10] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2564–2571.
[11] N. G. Aldana-Murillo, J.-B. Hayet, and H. M. Becerra, "Comparison of local descriptors for humanoid robots localization using a visual bag of words approach," *Intell. Automat. Soft Comput.*, vol. 24, no. 3, pp. 471–481, Apr. 2018. [Online]. Available: https://www.techscience.com/iasc/v24n3/39773
[12] M. Nowakowski, C. Joly, S. Dalibard, N. Garcia, and F. Moutarde, "Vision and Wi-Fi fusion in probabilistic appearance-based localization," *Int. J. Robot. Res.*, pp. 1–18, Apr. 2020.

[13] M. Cummins and P. Newman, "Appearance-only SLAM at large scale with FAB-MAP 2.0," *Int. J. Robot. Res.*, vol. 30, no. 9, pp. 1100–1123, Aug. 2011.
[14] N. Sunderhauf, S. Shirazi, F. Dayoub, B. Upcroft, and M. Milford, "On the performance of ConvNet features for place recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2015, pp. 4297–4304.
[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
[16] Q. Li, K. Li, X. You, S. Bu, and Z. Liu, "Place recognition based on deep feature and adaptive weighting of similarity matrix," *Neurocomputing*, vol. 199, pp. 114–127, Jul. 2016.
[17] P. Wozniak, H. Afrisal, R. G. Esparza, and B. Kwolek, "Scene recognition for indoor localization of mobile robots using deep CNN," in *Proc. Int. Conf. Comput. Vis. Graph.* Warsaw, Poland: Springer, 2018, pp. 137–147.
[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR) Conf. Track*, Y. Bengio and Y. LeCun, Eds. San Diego, CA, USA, May 2015, pp. 1–14. [Online]. Available: https://arxiv.org/abs/1409.1556v6
[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[20] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.
[21] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1251–1258.
[22] D. Scherer, A. Müller, and S. Behnke, "Evaluation of pooling operations in convolutional architectures for object recognition," in *Proc. Int. Conf. Artif. Neural Netw.* Thessaloniki, Greece: Springer, 2010, pp. 92–101.
[23] S. Jialin Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
[24] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
[25] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 3320–3328.
[26] R. Ranjan, C. D. Castillo, and R. Chellappa, "L_2-constrained softmax loss for discriminative face verification," 2017, *arXiv:1703.09507*. [Online]. Available: http://arxiv.org/abs/1703.09507
[27] M. Lin, Q. Chen, and S. Yan, "Network in network," 2013, *arXiv:1312.4400*. [Online]. Available: http://arxiv.org/abs/1312.4400
[28] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980

EMMANUEL OVALLE-MAGALLANES received the degree in computer engineering from the Universidad Autónoma de Zacatecas, UAZ-Zacatecas, in 2014, and the M.Sc. degree in computer science and industrial mathematics from the Centro de Investigación en Matemáticas, CIMAT-Guanajuato, in 2016. He is currently pursuing the Ph.D. degree in electrical engineering with the University of Guanajuato. His research interests include computer vision, image processing, numerical optimization, and deep learning.



NOÉ G. ALDANA-MURILLO received the degree in electro-mechanics engineering from the Tecnológico Nacional de México, Campus León, in 2011, and the M.Sc. degree in optomechatronics from Centro de Investigaciones en óptica, CIO-León, México, in 2014. He is currently pursuing the Ph.D. degree with the Centro de Investigación en Matemáticas, CIMAT-Guanajuato, Mexico. His research interests include visual localization and visual control of humanoid robotics.

JUAN GABRIEL AVINA-CERVANTES received the B.S. degree in communications and electronics engineering and the master's degree in electrical engineering (instrumentation and digital systems) from the University of Guanajuato, in 1998 and 1999, respectively, and the Ph.D. degree in informatics and telecommunications from the Institut National Polytechnique de Toulouse and the LAAS-CNRS, France, in 2005. He is currently a Researcher and full-time Professor with the University of Guanajuato. His research interests include the artificial vision for outdoor mobile robotics, pattern recognition, optimal control systems, and image processing.

JOSE RUIZ-PINALES (Member, IEEE) received the Ph.D. degree in computer science from the École Nationale Supérieure des TÉlÉcommunications of Paris, in 2001. He joined the Electronics Engineering Department, University of Guanajuato, as a full-time Professor in the same year. His research interests are in computer vision and artificial intelligence, including face recognition, handwriting recognition, deep learning, models of the human visual systems, and electronics. He has authored more than 64 research articles.

JONATHAN CEPEDA-NEGRETE received the bachelor's degree in engineering of communications and electronics, the master's degree in electrical engineering, and the Ph.D. degree in electrical engineering from the Engineering Division of the University of Guanajuato, Mexico, in 2011, 2012, and 2016, respectively. He is currently a full-time Professor with the Department of Agricultural Engineering, University of Guanajuato Campus Irapuato-Salamanca. His research interests include computer vision applications, mainly color image processing, pattern recognition, and computer graphics. All these topics especially applied to the digital art, artificial intelligence, and agricultural processes.

SERGIO LEDESMA received the M.S. degree from the University of Guanajuato, while working on the Setupof Internet in Mexico, and received the Ph.D. degree from the Stevens Institute of Technology, Hoboken, NJ, USA, in 2001. After graduating, he worked for Bar-clays Bank as a part of the IT-HR Group. He has worked as a Software Engineer for several years, and is the Creator of the software Neural Lab, Wintempla, and TexLab. He is a Research Professor at the University of Guanajuato, Mexico. He is currently on a sabbatical stay at the University of Ottawa, Canada. His areas of interest are artificial intelligence and software engineering.