Active object search using a pyramid approach to determine the next-best-view

Karen-Lizbeth Flores-Rodríguez[1], Felipe Trujillo-Romero[2], José-Joel González-Barbosa[1]
[1]Instituto Politécnico Nacional, CICATA-Unidad Querétaro, Querétaro, México
[2]División de Ingenierías, Campus Irapuato-Salamanca, Universidad de Guanajuato, Guanajuato, México

Article Info

Article history:

Received Aug 28, 2021
Revised Dec 25, 2021
Accepted Dec 30, 2021

Keywords:

Active camera
Local descriptors
NAO humanoid robot
Next best view
Object searching
Service robotics

ABSTRACT

The development of service robotics continues to arouse interest in the scientific community due to the complexity of the activities performed like interaction in human environments, identifying and manipulating objects, and even learning by themselves. This paper proposed to improve the perception of the environment by searching for objects in service robotics tasks. We present the development and implementation of an active object search method based on three main phases: Firstly, image pyramid segmentation to examine in detail the image features. Second step, object detection at each level of the pyramid through a local feature descriptor and a mutual information calculation. Finally, the next camera position selection through analyzing the object detections accumulation in the pyramid. To evaluate the implementation of the proposed method, we use a NAO robot in a familiar place for humans, such as an office or a home. Ordinary objects are part of our database with the premise that a robot must know them before looking for an object. The results in the experiments showed an acceptable performance in simulation and with a real platform.

Corresponding Author:

José-Joel González-Barbosa
Instituto Politécnico Nacional, CICATA-Unidad Querétaro
Cerro Blanco 141, Colinas de Cimatario, Querétaro, C.P. 76090, México
E-mail: jgonzalezba@ipn.mx

1. INTRODUCTION

The contributions of the scientific community to service robotics are extensive. However, accomplish complex activities by a robot is an open research and development. Kumar *et al.* [1] emphasized that service robots with a general-purpose approach must have: i) Physical interaction and manipulation with the real world; ii) Perception in unstructured environments; iii) Safe operation around humans; iv) Accessible human-robot communication; and v) Control network between sensors.

There are several reasons why the development of service robotics continues to arouse so much interest in the scientific community. One of the main reasons is the concern about the aging population [2]. The increase in older adults is a concern for first-world countries, as reflected in a smaller workforce in most production areas. Among these areas is health care as collaborating with nursing tasks [3], [4]. Current service robots that perform household tasks, nursing work, or assist in office tasks serve partial functions in human activities. The challenges of service robots are interaction in human environments, identifying and manipulating objects, and even learning by themselves. Besides, they have the advantage of being more friendly than a machine.

This work proposed to improve the perception of the environment by searching for objects in service robotics tasks. The use of a robot's camera as an active camera for the search of objects raises a solution. This method permits to compute the robot's next movement and acquire images each time with more information about the searched object. There are notable contributions to robotics and active camera in tasks of exploration, reconstruction and object recognition [5]-[11]. According to geometric characteristics, the final objective is to manipulate the objects correctly [12], [13]. A brief review of developments shows that robotic platforms with various sensors and cameras make easier searching for objects [14]-[17] though high costs and restricted use limit its application [18]-[21]. In contrast, robotic platforms that are relatively easy to acquire can cover many tasks. For example, the NAO robotic platform has been used in several applications related to service robotics [22]-[26]. The most relevant areas of opportunity in environment perception are segmentation, extraction, and feature analysis. The difficulties they face are: i) In geometric contour analysis, the main disadvantage is that geometric figures are prone to deformation; ii) In color analysis, which is affected by lighting changes and interference from background colors; iii) In optical flow analysis it must ensure that the target is always in focus; iv) Global features analysis is prone to having little information due to occlusions; and v) Using deep learning through neural networks entails the use of a lot of training data and delivers general results prone to confusion.

This paper presents the development and implementation of an active object search method. The method has three main phases: i) Image pyramid segmentation to examine in detail the image features; ii) Object detection at each pyramid level using a local feature descriptor; and iii) Next, camera position selection through analyzing the object detection accumulation in the pyramid. The proposed method divides the image into several levels by stimulating a pyramid to avoid the difficulties faced by segmenting objects. Dividing the image allows analyzing the features at different scales to prevent confusion and omissions. The feature analysis identifies and describing invariant features with the A-KAZE method that compensates for occlusions. Before performing an object search, it is learning the object using an object recognition module from [27] based on the A-KAZE descriptor and the growing cell structure (GCS) artificial neural network (ANN). This module allows learning everyday objects with relatively small training data.

The method evaluation is about using a NAO robot in a service robotics context. Figure 1 shows a virtual scene in which a NAO robot is close to a table with several objects. The intention is to place the robot in a place familiar to humans, such as an office or home. Using the object recognition module of [27] the robot builds a database and learns about several common objects. The active object search implementation consists of the following steps: i) The object search begins when the robot acquires an image; ii) The image goes through a pyramidal segmentation process where each segment becomes a new image and a mutual information calculation; iii) Each of the segments passes through a detection process of invariant features; iv) The invariant features proceed to the accumulation evaluation. If the feature accumulation gives a rough description of the object: found object. If the feature accumulation is zero: not found object; v) If it can't find the object, the robot will move to another random position and take a new image capture; vi) The camera's next position is calculated through true positives concentration to decide the next robot position; and vii) To repeat the process until obtaining a feature accumulation that appropriately describes the object searched.

The method allows applying a dynamic comparison to find a learned object while a robot explores a room. Once the robot finds the object, the robot can manipulate it to perform different service tasks in the future. The results in the experiments showed an acceptable performance in the simulation and with the real platform.

The structure of the article is: Section 2 includes related works comparing and discussing similar contributions to ours. Section 3 includes the methodology implemented, detailing the methods and tools available, and presents the database built using the object recognition module. Section 4 describes the active object search method, the pyramid segmentation, the feature analysis, and the next camera position selection. Section 5 presents the experiments and results using virtual and real robots and explains the implementation of the robot. A discussion of the results to highlight the challenges and difficulties. Finally, section 6 includes conclusion and highlights future work.

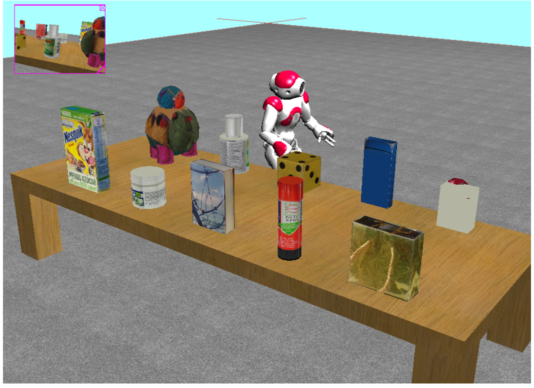


Figure 1. Virtual scene in which a NAO robot is close to a table with several objects

2. RELATED WORKS

Initially, the active camera and next-best-view (NBV) approaches were used only in 3D object reconstruction by automatics systems. Most of the works consider an object fixed in the middle of a platform and an acquisition system moving around it. The system tries to take the most data of the object to build it virtually. Since the ’90s and ’00s beginning, the works about NBV focus on research of an intelligent sensor that would be determining the optimal next sensing pose. That means getting the new point of view in order to gather the maximal information towards scene reconstruction [28]-[31], and object recognition and search [32]-[36]. In a robotics context, the NBV problem is used to decide the future sensing action more appropriate to execute and reach the best view of an object or a scene. In summary, the robot has to move to a position that allows it to acquire images automatically. Any robot, like robotic arms, mobile robots, even humanoid robots, with a vision sensor can be used to do that task.

Hernandez-Vice *et al.* [37], presented a box recognition application. This application extracts the characteristics to recognize the opening face using a camera system. The extracted features and a decision tree allow identifying the opening face of the box. This article establishes the basis for developing an application in which the TEO humanoid robot searches for the most optimal form of manipular packages and opens them, integrating this knowledge into an automated system. Meanwhile, Tsuru *et al.* [38], proposed a framework for an autonomous humanoid robot aimed at searching for an occluded object in an unknown environment based only on the 3D target model.

Arms robots are treated as an active camera using mutual information [39], or by a Kalman filter allowing to predict the camera position under conditions of uncertainty estimation [40]. Researchers [41], [42], began to work with the DLR 3D modeler attached to Kuka KR16 arm robot to explore 3D modeling. Their development evolved to active scene exploration in [6], [43], incorporating object recognition. The final goal was to manipulate the objects correctly, based on their geometry. Meanwhile, the NBV is used to obtain a better observation of incomplete objects by an eye-in-hand configuration for a robotic arm with an RGB-D camera [13]. Bajones *et al.* [12], presented the implementation of an intelligent robotic system that can perform an object searching task in a typical home scenario. The behavior of the robotic system is working smartly, taking into account a semantic segmentation used in indoor scenes. The semantic segmentation is supported by depth information. In the method [44] able to learn an object from a single scan and utilize active perception to face up with hard occlusions. The active camera or NBV method design depends on the application to solve and the robot type and sensory equipment. Mostly of the last works described are robot platforms equipped with stereo cameras, laser sensors, and ranging and other sensors. The sensory equipment increases the ability to solve the NBV problem.

In this work, it is performed an active object search NBV inspired approach by using a humanoid robot NAO. The idea of integrating a humanoid robot with an active object search is to be capable of interact in unstructured environments with humans. Humanoid robots have the advantage of performing full-body motion to get the next camera position. The research around the NAO robot is vast because of its excellent functionality, and of its easy way to acquire one. The object segmentation in [22] made the study of three cases of a real application. There bring in two segmentation strategies: contour analysis and optical flow data analysis, both of them used for getting object position and planning the next action, respectively. Heinrich *et al.* [24], a system based on natural language implements with the aim that a robot could learn about its environment based on color and form for object recognition. Hidago-Pẽna *et al.* [23], developed a cloud resources method to achieve an image searching creating training sets using principal component analysis (PCA). If internet access isn't available, then the robot asks a human to show the object and take pictures. Nefti-Meziani *et al.* [25] showed a stereo vision approach, the system is into the humanoid robot for indoor and outdoor implementations. Using open libraries like open source computer vision (OpenCV) and open graphic library (OpenGL) the robot perform automated detection and recognition and a 3D model visualization, respectively. Peipei *et al.* [45] presented a global and local vision to enhance the visual localization. For testing purposes, the A* algorithm was used to plan a path to avoid collisions and grasping tasks. They used two cameras, a ceiling camera, and the robot's camera. Also, Athar *et al.* [46] applied search-based for body motion planning through optimizing a graph search. There, they constructed a graph dynamically within three types of motion primitives and an A* algorithm variant called Weighted A*.

Li and Wang [47] developed an object localization and tracking using visual features. Here they established a monocular vision model and a fusion of cam-shift, Kalman, and particle algorithms fused to resolve problems in tracking. Such issues could be occlusion, background interference, and sudden move of the target. Besides, Zhang *et al.* [48], proposed a robotic grasping method that uses the deep learning method you only look once for multi-target detection and the auxiliary signs to obtain target location. The method can control the movement of the robot and plan the grasping trajectory based on the visual feedback information. Paulius *et al.* [49] introduced the functional, object-oriented network (FOON) as a graphical knowledge representation for manipulations that can be performed by domestic robots. The human acts as an assistant to the robot; the robot determines the best course of action through task tree retrieval and collaborates with the human to solve the problem posed to the robotic entity. Ofodile *et al.* [50], proposed and demonstrated an action recognition scheme based on a single-pixel direct time-of-flight detection. They apply machine learning in the form of recurrent ANN for data analysis and demonstrate successful action recognition. Zhang *et al.* [51], proposed a control system for accurately measuring distance using a monocular vision based on machine learning. They studied the task of a humanoid robot searching, pushing, and positioning a trolley loaded with various weights. Othman and Rad [52], addressed the problem of indoor room classification via several convolutional neural networks (CNN). The CNN deep learning approach was adopted for this purpose because of its superiority in the areas of object detection and classification.

3. MATERIALS AND METHODS

This paper presents the development and implementation of an active object search method. The system uses the accelerated-KAZE (A-KAZE) local feature descriptor for image feature analysis of the object recognition module [27]. The object recognition module uses the A-KAZE descriptor and GCS to learn and recognize objects. Here, the module supports an active object search using a pyramid approach and determines the next best view allowing applying a dynamic comparison to find a learned object while a robot explores a room.

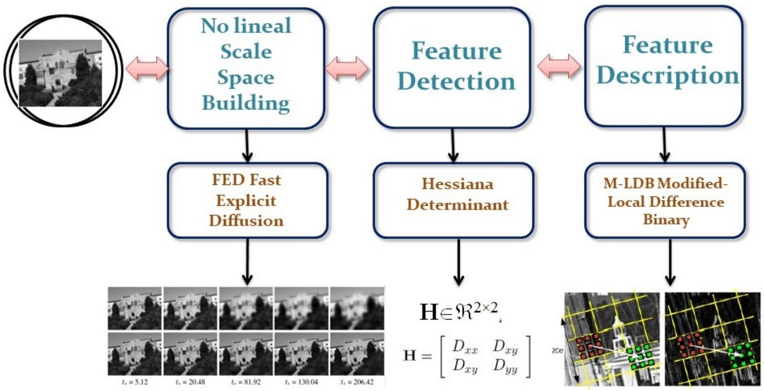
3.1. Object recognition module

There are many different objects in a human environment like an office or a house, where humans do their daily activities. These objects are complex to modeling mathematically, which gives us an interesting problem. The module recognition presented in [27] demonstrates to deal with those kinds of objects.

The object recognition module development involved two main methods: the A-KAZE descriptor and the GCS ANN. The A-KAZE descriptor [53] is an improvement of the previous KAZE [54]. A-KAZE uses a novel mathematical framework fast explicit diffusion (FED), which used a pyramidal approach to improve the computation speed for the scale space. Besides, it computes a robust descriptor that takes advantage of the gradient data from the nonlinear area. This approach is faster than the descriptors speeded up robust features (SURF), scale invariant feature transform (SIFT), KAZE, oriented FAST and rotated BRIEF (ORB), and binary robust invariant scalable keypoints (BRISK). A-KAZE consists of three principal tasks, and Figure 2 is shown a block diagram, include i) a non-linear scale space is constructed, ii) feature detection, and iii) feature description.

Besides, the object recognition module uses Kohonen's self-organizing variant: GCS ANN from [55]. This kind of ANN works as data associates, they distribute the neurons according to the attraction or repulsion between them, handle competitive strategies, and are easy to implement. The main advantage is automatically finding a suitable network structure and size, achieved through a controlled growth process that includes the occasional removal of units. The ANN model uses hyper-tetrahedrons because of their minimal complexity and their remarkable combination of large structures.

Figure 3 shows GCS growing and elimination process. In Figure 3(a) is showing the start process, with three neurons in a k-dimensional structure. The method is a variant of Kohonen's self-organized maps. It uses the competitive strategy where the winner wins everything. Through this strategy, the first neurons get all the input data distributions. After a certain number of adaptation steps, it verifies to insert a new neuron. The network will insert neurons forming tetrahedra until reaching the maximum size. In Figure 3(b) is showing eliminating a neuron and its neighborhood. In Figure 3(c) is showing a neuron removal is, after a certain number of adaptation steps, just like insertion. The network checks if there are dead neurons or with negligible weights to eliminate this and its neighborhood.


No lineal Scale Space Building

Feature Detection

Feature Description

FED Fast Explicit Diffusion

Hessiana Determinant

M-LDB Modified-Local Difference Binary

$\mathbf{H} \in \Re^{2\times 2}$,

$$\mathbf{H} = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}$$

(b) M-LDB binary test

Figure 2. General description of A-KAZE multi-scale feature detector and descriptor algorithm

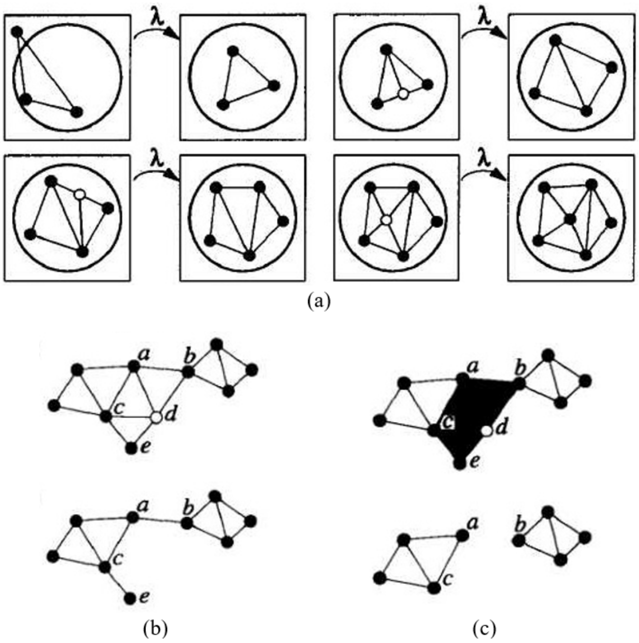


Figure 3. GCS growing and elimination process (a) GCS's neurons growth, (b) neuron removal, and (c) removal from your neighborhood [55]

The object recognition module sends as GCS inputs the features extracted through the A-KAZE descriptor from the objects. It is worth mentioning that this is a semi-supervised process. Then, in Figure 4, is shown the recognition module structured of two fundamental phases: learning and recognition.

The Algorithm 1 describes the learning phase. As input, the module receives different images from the objects. First, using the A-KAZE algorithm, as shown in Figure 2, the module extracts key points and builds for each object a features histogram. After that, all the histogram passes through the GCS for training. Finally, the algorithm creates a database for the classes learned; each one has a label that represents the object. Once complete the object database, it is possible to pass to the second phase, describe in Algorithm 2; in this phase, it uses only one image of the object as input. The module builds the features histogram using the feature description from the A-KAZE method. With the histogram created, GCS determines what object it is and returns the respective label.

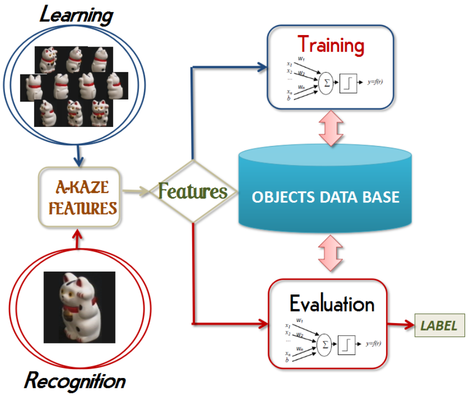


Figure 4. Object recognition module blocks diagram

Algorithm 1: Object recognition module. *Training*

```
Data: I images, N objects, L images per object.
Result: classes(N) Object classes.
for $n \leftarrow 1$ to $N$ do
    for $l \leftarrow 1$ to $L$ do
        keypoints =A-KAZE(I(n,l))H(n,l)=Build-Histos(keypoint)
classes(N)=GCS(H)
```

Algorithm 2: Object recognition module. *Evaluation*

```
Data: I image
Result: class(I) Object class.
keypoints =A-KAZE(I) H(I)=Build-Histos(keypoint) class(I)=GCS(H)
```

3.2. Objects database

With the scoop that before looking for an object must know it, the robot built a database of common objects. Using the object recognition module of [27], the robot learns several common objects. This database consists of the ten common objects showed in Figure 5. The objects are a liquid corrector, a cereal's box as shown in Figure 5(a), a turtle toy as shown in Figure 5(b), a bookas shown in Figure 5(c), a glue tube as shown in Figure 5(d), a cell phone as shown in Figure 5(e), a medicine bottle as shown in Figure 5(f), a gift bag as shown in Figure 5(g), a dice as shown in Figure 5(h), and a gift box as shown in Figure 5(i). These objects are only a little example that common elements found in both a home and an office.

Webots is a robotic environment simulation that allows the creation of scenes, model objects, and apply textures from real pictures. Also, we create virtual models from the real models for use in the virtual environment Webots to test finality. The virtual models are in Figure 6.

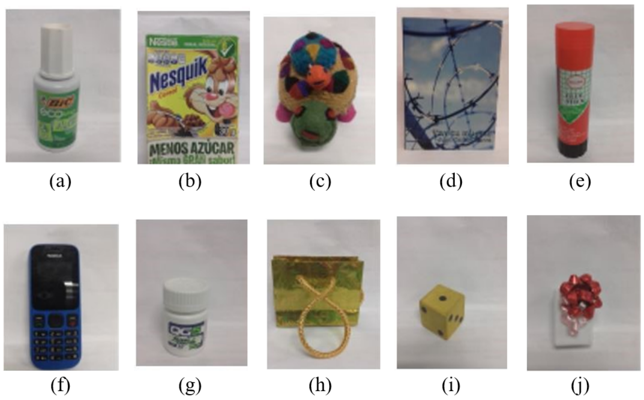


Figure 5. Objects database: (a) liquid corrector,(b) cereal box, (c) turtle toy, (d) book, (e) glue tube, (f) cellphone, (g) medicine bottle, (h) gift bag, (i) dice, and (j) gift box

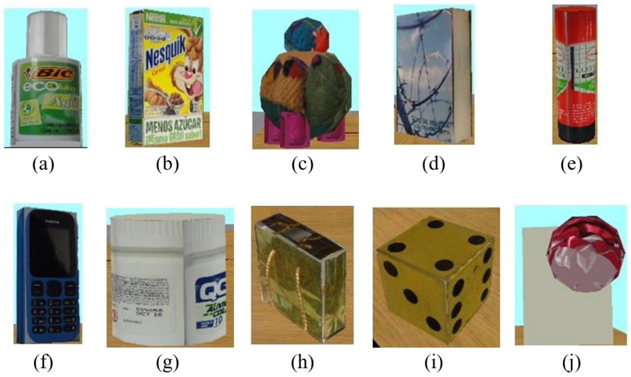


Figure 6. The ten simulated objects in Webots environment same as in Figure 5

4. ACTIVE OBJECT SEARCH

The active object search method has three main phases: i) Image pyramid segmentation to examine in detail the image features; ii) Object detection at each pyramid level using a local feature descriptor and a mutual information calculation; and iii) Next, camera position selection through analyzing the object detection accumulation in the pyramid.

The active object search goal is to allow a robot to get closer to it and facilitate developing actions like manipulation. Here is the description of the pyramid approach to determine the NBV, allowing the robot to detect if an object is in the scene and decide his next move. The robot can get closer and find the best position to see a specific object. At this time, it considers that there are no obstacles between the robot. It can navigate freely around a room.

The active object search implementation consists of the following steps: i) The active object search begins when the robot acquires an image from a random position and passes it to the object detection module; and ii) In the object detection module, the image goes through a pyramidal segmentation process where each segment becomes a new image. Figure 7. Image pyramidal segmentation process, the image is divided adding a line vertical and horizontal progressively. Figure 7(a) one division horizontal: two images, Figure 7(b) the image is divided horizontally: four images, Figure 7(c) the previous division increase to six by three lines, Figure 7(d) the previous division increases into 24 by three horizontal lines and six vertical, Figure 7(e) and Figure 7(f) the procedure continues until division increase to 64 by a total of 14 lines. The process is an X, Y

For loop nested dividing the image iteratively. The polynomial regression model the iterative increase images division in the (1).

$$y = \frac{1}{4}x^2 + 1.0119x + 0.8214 \tag{1}$$

Where x are the iterative lines, and y are the new images; iii) Each of the segments passes through a detection process of invariant features. The process is through the object recognition module to identify the object searched and a mutual information calculation; iv) The invariant features proceed to the accumulation evaluation. If the feature accumulation gives a rough description of the object: found object. If the feature accumulation is zero: not found object; v) If the object is not found, the robot will move to another random position and take a new image capture; vi) The camera's next position is calculated through true positives concentration to decide the next robot position. Figure 8 shows the cereal box search; the positives are concentrated close to it despite the occlusion. The concentration allows deciding where the robot will move: turn left, turn right, or go forward; and vii) Repeat the process until obtaining a feature accumulation that appropriately describes the object searched.

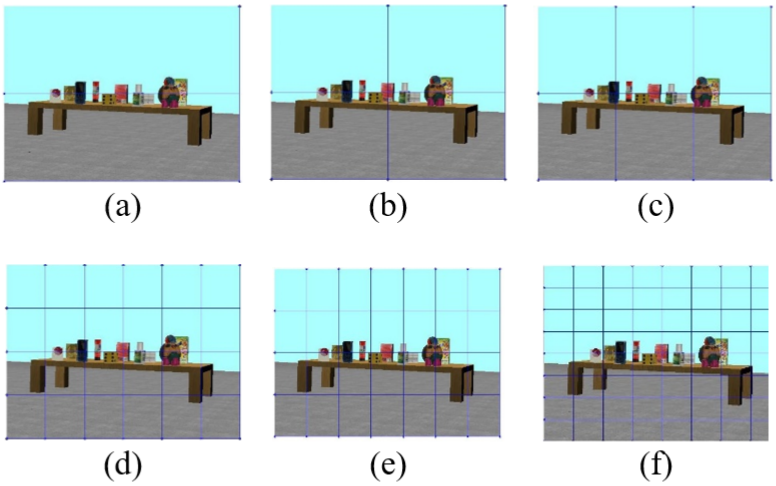


Figure 7. Image pyramidal segmentation process, the image is divided adding a line vertical and horizontal progressively. From (a) one division horizontal: two images, (b) the image is divided horizontally: four images, (c) the previous division increase to six by three lines, (d) the previous division increases into 24 by three horizontal lines and six vertical, (e) and (f) the procedure continues until division increase to 64 by a total of 14 lines

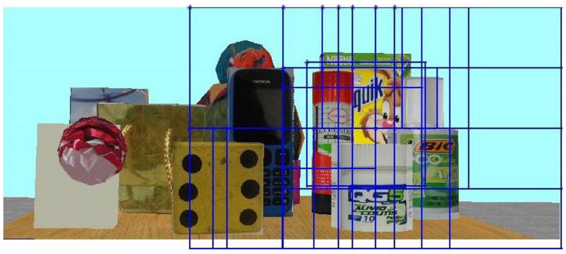


Figure 8. Nesquik cereal box search in one image, the image shows all the positions identify as true

4.1. Object detection

The process in which the active search works is as follows. First, in the learning phase, the system computes a feature histogram using (2). These values recover the feature description from the A-KAZE method as well as the pose of the robot. That means the robot's direction must move the camera next position to keep the object centering on the image computed the mean and variance for every object data learned.

$$I_p \begin{pmatrix} I_x(p) \\ I_y(p) \end{pmatrix} = \frac{1}{n} \sum_{i=1}^{n} \begin{pmatrix} I_x(i) \\ I_y(i) \end{pmatrix} \tag{2}$$

Where I_p represents the mean of the coordinates, x and y get from the feature description. I_i represents the coordinates feature description for all image data from each object. Moreover, n is the total number of images data for each object. These values make it possible to compute the probability of observing some object features when the robot takes a picture of it in a given direction.

Therefore, It computes the conditional probability for observing features true positives (TP) given some direction d by (3):

$$P(TP_t|d_t) = \int_{x_t} P(TP_t|x_t, d_t)P(x_t)dx_t \tag{3}$$

In the searching phase, it computes mutual information assuming equal *a priori* probability that a feature vector belongs to each object. Thus mutual information is calculated as shown in (4).

$$I_0(\Omega, TP|d) = \sum_{k=1}^{K} e_k(d)P_k \tag{4}$$

Where Ω is the class in which a feature or TP belongs. Here Ω is directly related to our $I(Class - (subimage))$; e_k is the entropy of the direction that the robot will take. The following equation computes the entropy:

$$e_k(d) = \sum_{TP_i} P(TP_i|\Omega_k, d) \log \frac{P(TP_i|\Omega_k, d)}{P(TP_i|d)} \tag{5}$$

Mutual information gives us the best value between the estimated and the observation made in the searching phase. Hence, it is looking for the direction d_0^* that maximizes the mutual information show in (6).

$$d_0^* = \max_a I_0(\Omega, TP|d) \tag{6}$$

Accordingly, to this last relation, the robot must execute the direction obtained, and the module has to update the probabilities P_k (see (7)) for every possible class.

$$P_k = \frac{P(TP_0|\Omega_k, d_0)P(\Omega_k|d_0)}{P(TP_0, d_0)} \tag{7}$$

This process is iterated sequentially until the probability of the searching object has reached a threshold of 95%.

In the next list, are present some observations about the process above mentioned. i) The TPs in the searching phase is the same that the features obtained in the learning phase; ii) During the learning phase, the robot must center the object in the image according to the directions saved. Those directions are related to the features of the corresponding image; and iii) Equation (7) used the recognition module to get the TPs; hence, the neural network GCS has the assignment of recognizing the object present in the scene. Here the probabilities update to help the ANN to give an output.

The method allows applying a dynamic comparison to find a learned object while a robot explores a room. Once the robot finds the object, the expectation is to manipulate it to perform different service tasks in the future. The results in the experiments showed an acceptable performance in the simulation and the NAO platform.

5. RESULTS AND DISCUSSION

5.1. Results

The method evaluation is about using an NAO robot in a service robotics context. In a free obstacle semi-structured environment where the robot knows the dimensions, he has to navigate to find an object. It places a table somewhere in the scene and on it a few objects. Figure 1 shows a virtual scene in which an NAO robot is close to a table with several objects. The robot is in a place familiar to humans, such as an office or home. With the scoop that before looking for an object must know it, the robot built a database of ordinary objects. At this point, the robot already learns and labels the objects on the table by using the object recognition module described in section 3.1. The robot must receive a label of an object learned to start an object search. Next are the object recognition module results, experiments with the Webots simulator with a virtual robot NAO, an statistical analysis, and experiments within the real robot NAO.

5.1.1. Object recognition module results

This section briefly presents the result obtained when evaluating the object recognition module in both virtual and real objects from section 3.2. To verify the high functioning of our module implementation, it trains the system with relatively small data of the ten objects database (ten images each one). For evaluation, the system took ten more images for each object of an arbitrary position. The section shows results in both virtual and real objects. The evaluation results for virtual and real system implementation are in the confusion matrix shown in Table 1 and Table 2 respectively.

Table 1 displays the results of the virtual system with a classification rate of 71%. While Table 2 belonging to the real system results, the classification percentage arises with a recognition rate of 94%. The results obtained evidence that the module developed for recognizing the objects has a high performance on real data. However, a virtual environment presents a low performance due to the images' resolution since a simulated camera loses quality. It must say that the real cameras in the NAO robots have not so well performance either. For that reason, the recognition rate is not high; nonetheless, the module for object recognition has a satisfactory performance.

Table 1. Virtual system performance evaluation confusion matrix

	1	2	3	4	5	6	7	8	9	10
1	5	0	0	0	0	0	0	5	0	0
2	0	10	0	0	0	0	0	0	0	0
3	0	4	6	0	0	0	0	0	0	0
4	2	0	0	8	0	0	0	0	0	0
5	0	0	3	0	7	0	0	0	0	0
6	0	0	0	0	0	5	0	0	0	5
7	1	2	0	0	0	0	7	0	0	0
8	1	0	0	0	0	0	2	7	0	0
9	0	0	0	0	0	0	0	1	8	1
10	0	2	0	0	0	0	0	0	0	8

Table 2. Real system performance evaluation confusion matrix

	1	2	3	4	5	6	7	8	9	10
1	10	0	0	0	0	0	0	0	0	0
2	0	10	0	0	0	0	0	0	0	0
3	0	0	10	0	0	0	0	0	0	0
4	0	0	0	10	0	0	0	0	0	0
5	0	0	0	0	8	0	0	2	0	0
6	0	0	0	0	0	10	0	0	0	0
7	0	0	0	0	0	0	10	0	0	0
8	2	0	0	0	0	0	0	8	0	0
9	0	0	0	0	0	0	0	0	8	2
10	0	0	0	0	0	0	0	0	0	10

5.1.2. Virtual robot NAO

In this section are the experiments for the virtual data set in section 3.2. An NAO robot virtual searched for each one: i) bottle of liquid corrector, ii) Nesquik cereal box, iii) turtle toy, iv) book, v) glue tube, vi) cellphone, vii) medicine bottle, viii) gift bag, ix) dice, and x) gift box.

The first virtual experiment is for validation. The virtual robot acquired four images around the table to identify each of the ten objects. As shown in Figure 9, the objects are in a different location, and some of them are occluded. In Figure 9(a), the objects 1, 2, 3, and 7, meanwhile objects 4, 5, and 6 are occluded, and objects 8, 9, 10 do not even appear. In Figure 9(b), meantime, the objects 1, 2, 3, and 7 are not there, object

4 appear occluded, and objects 5, 6, 8, 9, and 10 are seeing. In Figure 9(c), objects 1, 3, 6, 7, 9, and 10 are seeing, and the rest are occluded. In Figure 9(d), objects 3, 5, 8, and 10 are seeing, and the rest are occluded.

Table 3 presents the ground truth for validation showing where are each of the objects (object number) at the four images (Image 1 - Image 4). The ground truth is as follows, green marks if the object is completely seen; blue marks if the object is occluded, and red marks if the object is not there.

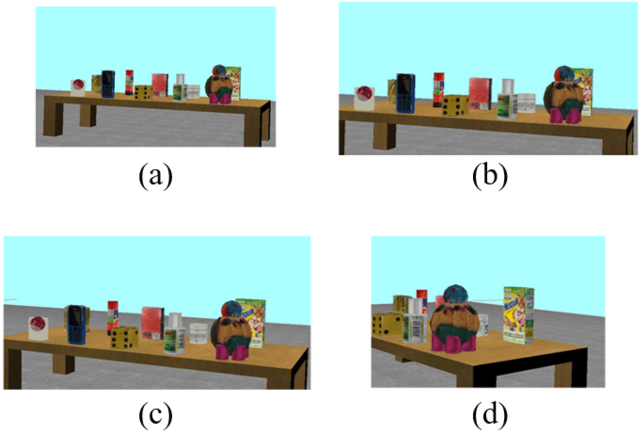


Figure 9. The 4 images the virtual experiment validation acquired by the NAO robot, the objects are at different location, and some of them are occluded, (a) object 1, 2, 3 and 7: good view; 4, 5 and 6: occluded, (b) 4: occluded, and 5, 6, 8, 9 and 10: good view, (c) 1, 3, 6, 7, 9, and 10: good view, and (d) 3, 5, 8 and 10: good view

Table 3. Ground truth for validation. Here is the object at each of the four images. Green: seen, Blue: occluded, Red: not seen

Object	Image 1	Image 2	Image 3	Image 4
1	Green	Red	Green	Blue
2	Green	Red	Blue	Blue
3	Green	Red	Green	Green
4	Blue	Blue	Blue	Blue
5	Blue	Green	Blue	Green
6	Blue	Green	Green	Blue
7	Green	Red	Green	Blue
8	Red	Green	Blue	Green
9	Red	Green	Green	Blue
10	Red	Green	Green	Green

The validation goal is to prove if the active object search can determine the next best view. The ten object search helps to visualize the performance of the system when a decision must be made. The active object search results are in Table 4 as follows: the first column is the object number; the second is the images where the object appears; the third column is the true positive rate (TP); the fourth column is the false positive rate (FP), and the last column is the average of true positives. The Equation (3) calculate the positives using by establishing the actual position of the object in the image, mutual information (4) and the class in which bellows with (5). The TP are those around and close the real position of the object searched. These will allow deciding where to move the robot next. The FP rate is when the system confuses the object, and the average is the result obtained, considering the previous ones.

Some examples to visualize the object segments detected are in Figure 10 with the active object searching for the ten objects. The liquid corrector bottle, in Figure 10(a), as most of the positives are at the left, the next view must be a capture, moving the robot to the left. This object shows a successful detection with 75.38%. The cereal box within 66.67% of the positive looks confused to consider detected, in Figure 10(b). The turtle toy, in Figure 10(c), shows a successful detection with 74.07%, the images show a clear crowding of positives closes to the turtle toy. The book, in Figure 10(d), is occluded, but the positives get 70.21% and are close to it, allowing to move the robot toward the object and detect it better. The glue tube, in Figure 10(e), shows more positives than falses within 81.25%. Indeed, the cellphone in Figure 10(f) has the best view since with 100% of positives and the object is in the center of the image. The medicine bottle, in Figure 10(g), gets 68.42% instead of being far; however, there are more positives on the object. The gift bag, in Figure 10(h), only has a positive, but it is sufficient to detect the object within 100%. The dice, in Figure 10(i) and the gift box, in Figure 10(j), both were confused and maybe will be very difficult to find them.

Table 4. NBV accumulation detection, the average of true positive per object. Green: next best pose correct decision, Red: poor next best pose correct decision

Object	Image	TP	FP	Average	Colour
1	1	49	16	75.38%	Green
2	1	48	24	66.67%	Green
3	1	80	28	74.07%	Green
4	3	33	14	70.21%	Green
5	4	26	6	81.25%	Green
6	2	5	0	100%	Green
7	3	13	6	68.42%	Green
8	4	1	0	100%	Green
9	2	5	54	8.47%	Red
10	3	6	10	37.5%	Red

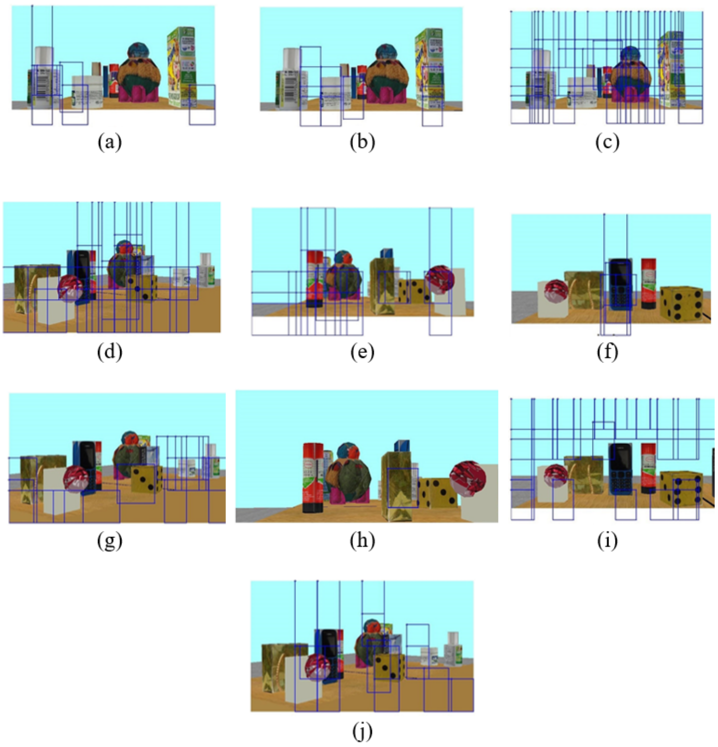


Figure 10. Some examples to visualize the object segments detected by the system searching for: (a) bottle of liquid corrector, (b) Nesquik cereal box, (c) t'urtle toy, (d) book, (e) glue tube, (f) cellphone, (g) medicine bottle, (h) gift bag, (i) dice, and (j) gift box

The results show a good performance for the active search system. An important detail to remark is that the image quality in simulation is not the best, and the learning was made with simulated images. The total average of active object searches goes from 66-100% the system found eight objects with success. The system only needs more TP than FP accumulation to consider an object found. The results prove that the method has suitable bases to execute an object search. So that to better visualize the use of this method, it is necessary to carry out the implementation with a real robot.

5.1.3. Statistical analysis

The evaluation search algorithm has performed a series of 50 tests per object. These are additional tests to those developed to obtain the relationship of TP and FP within the search for a particular object. The tests are not interested in the amount of TP found in each of the sub-images. However, instead, if the robot were able to find the object that was asked to search. If the robot did not find the requested object, it was not our interest to know what the object was. It was required to know if the robot was able to fulfill the assigned task.

In carrying out these tests, it was necessary to take into account some considerations. These considerations are listed: i) It was considered a null or empty class. This class was the same as the robot will not find the requested object. Not finding the object corresponding to two possible causes are the requested object was not on the table or that the robot could not find the object because it deviated from the position in which the object was; In the scene, the robot started at a random position. Moreover, it does not necessarily look in the direction of the table where the objects are; iii) It decided randomly if the object to search would be present in the scene. This decision is weighing by a probabilistic function giving preference to the object being present; and iv) This evaluation was carried out in the simulation environment Webots. The above facilitate the repetition of the tests with different parameters.

Table 5 shows the results obtained when performing the corresponding tests of the search of each of the objects 50 times. That object 9 is the one that cost the most work to find, failing 14 times of the 50 searches performed. While objects 3, 7, and 8 were easier to find. In this case, only 2 of the search attempts failed. Finally, it shows that the system fails 59 times, presented in the same Table 5 within class 11. This class 11 represents the null class.

It should be noted that although it seems that this null class is the one with higher value does not mean that the system failed more times than we guess. It should be noted that in this class the total number of times that the system could not find any of the ten objects is presented. Therefore, if it is considered that 50 searches were performed per object in total, 500 searches were done, so the success rate is 88.2%, with only 11.8% missing.

Table 5. Experiments: number of times found

Object	Found
1	46
2	45
3	48
4	44
5	38
6	47
7	48
8	48
9	36
10	40
11	59

5.1.4. Real robot NAO

Four explorations around the table took place, analyzing the robot performance identifying each of the 10 objects, see Figure 12. In the first exploration, Figure 12(a) is showing 20 objects, including those in the database, are on the table; in the second exploration, only 10 objects were left on the table, see Figure 12(b) In the third exploration, five, and the fourth exploration, just two objects remained, see in Figure 12(c) and Figure 12(d). In Table 6 is the ground truth specifying the objects appearing in each search. The objects 1, 2, 3, and 4 appear only in the first search. The objects 5, 6, 7, and 8 appear in the first and second searches. Object 9 appears in search 1, 2, and 3. The 10 object appears in all of them.

The NAO robot captured images during the explorations from different locations around the table. The active object search result is in Table 7 as follows: the first column is the object number, the second column is the exploration, the third column is the TP rate, the fourth column is the FP rate, and the last column is the total average. The counts of positives were calculated using (3) by establishing the actual position of the object in the image, mutual information (4), and the class in which bellows with (5). The positive counts are those around and closest to the real position. The TP rate is the total count of correct times where the object appears, the FP rate is the total count of times that confused the object, and the total average is the result obtained considering the previous.

Figure 13 shows some images of the active search made by the robot. In Figure 13(a), is the liquid corrector bottle on the right of the image within 66.8%, the next pose will be moving the robot to the right side. In Figure 13(b), is the dice with 60%, only two positives closest to it are enough to find the object. In Figure 13(c), is the gift box detected for more true positives 70.27%. In Figure 13(d), enough of the positives are closest to the McDonald bear 72.29%. In Figure 13(e), the positives closest to the cell phone will allow moving the robot by the right side to have a better view of 67.53%. In Figure 13(f), it is possible to observe a total confusion of the blue marker by a poster in the back with a similar appearance. In Figure 13(g), the search of the red marker seems half to accomplish. In Figure 13(h), Figure 13(i), and Figure 13(j), the notebook, the turtle toy, and the Nesquik cereal box have each one a very well observed accumulation of positives on them within 80% or more. Each search by exploration has a next-best pose correct decision with satisfactory detection average from 60-81.81%. To be a success only needs more true positives than false positives. Therefore, it shows that the method can be a good approach to perform object searching.

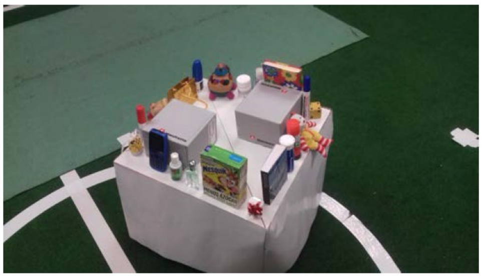
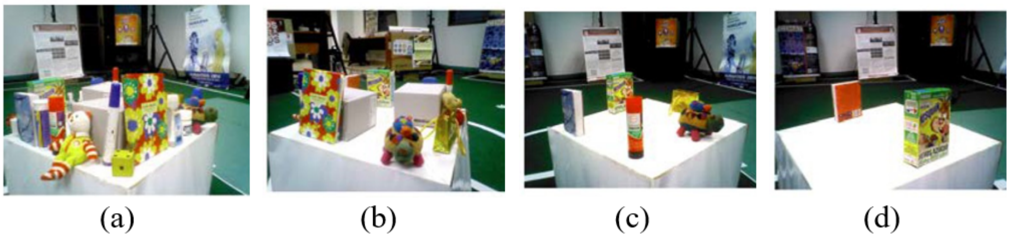


Figure 11. Platform 30 cm height with the 20 objects on it for exploration

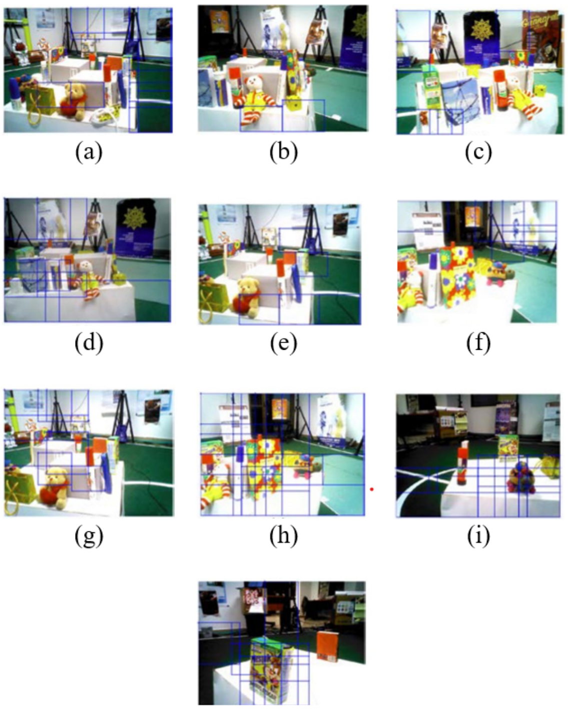


(a) (b) (c) (d)

Figure 12. The four exploration around the table made by the NAO robot, (a) 20 objects, (b) 10 objects, (c) 5 objects, and (d) 2 objects

Table 6. Ground truth for validation. The real object at each of the four images. Green: seen, Red: not seen

Object	Search 1	Search 2	Search 3	Search 4
1	Green	Red	Red	Red
2	Green	Red	Red	Red
3	Green	Red	Red	Red
4	Green	Red	Red	Red
5	Green	Green	Red	Red
6	Green	Green	Red	Red
7	Green	Green	Red	Red
8	Green	Green	Red	Red
9	Green	Green	Green	Red
10	Green	Green	Green	Green



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

Figure 13. NBV object detection by the real robot: (a) bottle of liquid corrector, (b) dice, (c) gift box, (d) Mcdonald bear, (e) cellphone, (f) blue marker, (g) red marker, (h) notebook, (i) turtle toy, and (j) Nesquik cereal box

Table 7. NBV accumulation detection, the average of true positive per object. Green: next best pose correct decision, Red: poor next best pose correct decision

Object	Search	TP	FP	Average	Colour
1	1	101	50	66.88%	Green
2	1	6	4	60.00%	Green
3	1	26	11	70.27%	Green
4	1	60	23	72.29%	Green
5	2	52	25	67.53%	Green
6	2	17	9	65.38%	Green
7	2	24	19	55.81%	Green
8	2	172	43	80.00%	Green
9	3	90	22	80.35%	Green
10	4	81	18	81.81%	Green

5.2. Discussion

The work described here aimed to prove the good performance identifying the next best view during active object searching. Here, the object recognition module upgrades an active object search for future robot service tasks. Moya *et al.* [27] presented many static experiments showing excellent performances. Suddenly, the module acts as a dynamic search by a humanoid robot through image pyramid segmentation. Previous work assumed that the objects would only be recognized and manipulated without a searching phase. Moreover, it is neither taking into account the displacement of the humanoid robot.

In this approach, it must be highlighted that to avoid the difficulties faced when segmenting objects. A method was designed that divides the image into several levels by simulating a pyramid. Dividing the image allows analyzing the features at different scales to avoid confusion and omissions. The feature analysis was carried out by identifying and describing invariant features with the A-KAZE method that compensates for occlusions.

We performed experiments in a virtual and real environment. The virtual objects recognition results showed a classification percentage of 71%, while with real objects, the percentage raises a recognition rate of 94%. The results left us thinking about having better results in search with a real robot than a virtual robot because of the low quality of the images taken with a virtual camera inside a virtual scenario.

The active object search performance allows us to prove the ability of the system to perform a dynamic comparison to find one object learned while a robot explores a room. In both the implementation by software Webots and the virtual humanoid robot NAO; and the implementation in a real platform NAO, the system only needs more TP than FP accumulation closest to the object search to consider it detected. The results prove that the NBV method presented here has suitable bases to execute an object search.

The active object search results showed a good performance; the recognition percentage varies depending on the learning object module and the objects. It can be observed that the image quality in simulation is not the best, and learning was with simulated images. Even though the total average of object recognition goes from 66-100%, eight of the ten objects were successfully detected. In contrast, with a real robot, the recognition average goes from 60-81.81%. It must say that the real cameras in the NAO robots do not have a good performance either. The above comment is why the recognition rate is not higher; nonetheless, we can say that module for object recognition has an excellent performance.

Even the results showed by real robot performance are satisfactory, around 70%. If real results are compared with simulation results, we can say that virtual is a little better because particular objects reach 100%. Nonetheless, it is considered that the percentage has come out low since the robot images are diffuse and of low quality. Another reason is that the recognition module depends on the objects, then if it obtained a low classification percentage par consequence, NBV would tend to confuse the objects. However, the results show that the NBV method has suitable bases to execute an object search. Anyone with experience in this field can observe that the percentage could improve if a better camera is used or if image processing is applied before evaluating.

6. CONCLUSION

In this work, system development and implementation for active object search using a pyramid approach to determine the next best view was presented. We used an object recognition module based on feature descriptor and semi-supervised ANN. The system applied a dynamic comparison to find one object learned while a robot explores a room. Different experiments validated the correct system implementation. We used the software Webots and both virtual and real humanoid robots for experimentation. The humanoid robot NAO was used to know if the robot could decide the next move to get closest to the object searched.

Results showed a satisfactory performance; the total average of active object detection through virtual robots went from 66-100%. Eight of the ten objects were successfully detected. In real robot performance, the total average of object detection went from 60-81.81%. The system detected the ten objects. The system only needs more TP than FP accumulation to consider an object detected. The percentage of recognition can be improved in both real and virtual robots if the image quality is improved because the searching depends on the learning process. The results proved that the NBV inspired method presented here has suitable bases to execute an object search.

Further, it is considered applying whole-body movements to the humanoid robot to be more precise to reach an object and do the manipulation task. Also, it is considering the obstacle avoidance and semantic information about the environment and object features too. Humanoid robot NAO is an excellent platform with some disadvantages like no stereo vision and no depth information, but it can exploit another platform's ability.

ACKNOWLEDGEMENT

Acknowledgments to CONACYT México and IPN México for the PhD study grants 584190.

REFERENCES

[1] V. Kumar, G. Bekey, and Y. Zheng, "Industrial,personal, and service robots," in *INTERNATIONAL ASSESSMENT OF RESEARCH AND DEVELOPMENT IN ROBOTICS*, I. WTEC, Ed. WTEC: World Technology Evaluation Center, Inc. (WTEC), 2005, ch. 5, pp. 41–48. [Online]. Available: http://scienceus.org/wtec/docs/screen-robotics-final-report-highres.pdf.

[2] United Nations, "World population prospects 2019," 2019. [Online]. Available: https://population.un.org/wpp/.

[3] Duquesne University School of Nursing, "Robotics in nursing," 2020. [Online]. Available: https://onlinenursing.duq.edu/blog/robotics-in-nursing.

[4] U. Tripathi, R. S. J, V. Chamola, A. Jolfaei, and A. Chintanpalli, "Advancing remote healthcare using humanoid and affective systems," *IEEE Sensors Journal*, doi: 10.1109/JSEN.2021.3049247.

[5] N. Atanasov, B. Sankaran, J. Le Ny, G. J. Pappas, and K. Daniilidis, "Nonmyopic view planning for active object classification and pose estimation," *IEEE Transactions on Robotics*, vol. 30, no. 5, pp. 1078-1090, Oct. 2014, doi: 10.1109/TRO.2014.2320795.

[6] S. Kriegel, C. Rink, T. Bodenmüller, and M. Suppa, "Efficient next-best-scan planning for autonomous 3D surface reconstruction of unknown objects," *Journal of Real-Time Image Processing*, vol. 10, no. 4, pp. 611–631, Dec. 2015, doi: 10.1007/s11554-013-0386-6.

[7] C. McGreavy, L. Kunze, and N. Hawes, "Next best view planning for object recognition in mobile robotics," in *Proceedings of the 34th Workshop of the UK Planning and Scheduling Special Interest Group, PlanSIG 2016, Huddersfield, UK, December 15-16, 2016*, ser. CEUR Workshop Proceedings, L. Chrpa, S. Parkinson, and M. Vallati, Eds., vol. 1782. CEUR-WS.org, 2016. [Online]. Available: http://ceur-ws.org/Vol-1782/paper_6.pdf

[8] T. Fäulhammer, *et al.*, and M. Vincze, "Autonomous learning of object models on a mobile robot," *IEEE Robotics and Automation Letters*, vol. 2, no. 1, pp. 26–33, Jan 2017, doi: 10.1109/LRA.2016.2522086.

[9] L. J. Manso, M. A. Gutierrez, P. Bustos, and P. Bachiller, "Integrating planning perception and action for informed object search," *Cognitive Processing*, vol. 19, no. 2, pp. 285–296, may 2018, doi: 10.1007/s10339-017-0828-3.

[10] J. Delmerico, S. Isler, R. Sabzevari, and D. Scaramuzza, "A comparison of volumetric information gain metrics for active 3D object reconstruction," *Autonomous Robots*, vol. 42, no. 2, pp. 197–208, Feb. 2018, doi: 10.1007/s10514-017-9634-0.

[11] F. Gomez-Donoso, F. Escalona, F. M. Rivas, J. M. Cañas, and M. Cazorla, "Enhancing the Ambient Assisted Living Capabilities with a Mobile Robot," *Computational Intelligence and Neuroscience*, vol. 2019, p. 9412384, Apr. 2019, doi: 10.1155/2019/9412384.

[12] M. Bajones, D. Wolf, J. Prankl, and M. Vincze, "Where to look first? behaviour control for fetch-and-carry missions of service robots," Oct. 2015. [Online]. Available: http://arxiv.org/abs/1510.01554.

[13] R. Monica and J. Aleotti, "Contour-based next-best view planning from point cloud segmentation of unknown objects," *Autonomous Robots*, vol. 42, no. 2, pp. 443–458, Feb. 2018, doi: 10.1007/s10514-017-9618-0.

[14] Z. Marton, D. Pangercic, N. Blodow, J. Kleinehellefort, and M. Beetz, "General 3d modelling of novel objects from a single view," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 3700–3705, doi: 110.1109/IROS.2010.5650434.

[15] M. Krainin, B. Curless, and D. Fox, "Autonomous generation of complete 3d object models using next best view manipulation planning," in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 5031–5037, doi: 10.1109/ICRA.2011.5980429.

[16] M. Nieuwenhuisen, *et al.*, "Mobile bin picking with an anthropomorphic service robot," in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 2327–2334, doi: 10.1109/ICRA.2013.6630892.

[17] J. Rojas-Quintero and M. Rodríguez-Liñán, "A literature review of sensor heads for humanoid robots," *Robotics and Autonomous Systems*, vol. 143, p. 103834, 2021, doi: 10.1016/j.robot.2021.103834.

[18] T. Foissotte, O. Stasse, A. Escande, and A. Kheddar, "A next-best-view algorithm for autonomous 3d object modeling by a humanoid robot," in *Humanoids 2008 - 8th IEEE-RAS International Conference on Humanoid Robots*, 2008, pp. 333–338, doi: 10.1109/ICHR.2008.4756001.

[19] T. Foissotte, O. Stasse, A. Escande, P. Wieber, and A. Kheddar, "A two-steps next-best-view algorithm for autonomous 3d object modeling by a humanoid robot," in *2009 IEEE International Conference on Robotics and Automation*, 2009, pp. 1159–1164, doi: 10.1109/ROBOT.2009.5152350.

[20] A. Andreopoulos, S. Hasler, H. Wersing, H. Janssen, J. K. Tsotsos, and E. Korner, "Active 3d object localization using a humanoid robot," *IEEE Transactions on Robotics*, vol. 27, no. 1, pp. 47–64, Feb 2011, doi: 10.1109/TRO.2010.2090058.

[21] B. Browatzki, V. Tikhanoff, G. Metta, H. H. Bülthoff, and C. Wallraven, "Active in-hand object recognition on a humanoid robot," *IEEE Transactions on Robotics*, vol. 30, no. 5, pp. 1260–1269, Oct 2014, doi: 10.1109/TRO.2014.2328779.

[22] K. Li, M. Q. -H. Meng, and X. Chen, "Robot aided object segmentation without prior knowledge," in *Proceedings of the 10th World Congress on Intelligent Control and Automation*, 2012, pp. 4797–4802, doi: 10.1109/WCICA.2012.6359387.

[23] E. Hidago-Peña, L. F. Marin-Urias, F. Montes-González, A. Marín-Hernández, and H. V. Ríos-Figueroa, "Learning from the web: Recognition method based on object appearance from internet images," in *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2013, pp. 139–140, doi: 10.1109/HRI.2013.6483540.

[24] S. Heinrich, *et al.*, "Object Learning with Natural Language in a Distributed Intelligent System: A Case Study of Human-Robot Interaction," in *Foundations and Practical Applications of Cognitive Systems and Information Processing*, F. Sun, D. Hu, and H. Liu, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 811–819.

[25] S. Nefti-Meziani, U. Manzoor, S. Davis, and S. K. Pupala, "3D perception from binocular vision for a low cost humanoid robot NAO," *Robotics and Autonomous Systems*, vol. 68, pp. 129 – 139, 2015, doi: 10.1016/j.robot.2014.12.016.

[26] V. Moya, E. Slawiñski, V. Mut, and B. Wagner, "Intercontinental bilateral-by-phases teleoperation of a humanoid robot," *IEEE Latin America Transactions*, vol. 20, no. 1, pp. 64–72, Jan. 2022, doi: 10.1109/TLA.2022.9662174.

[27] K. L. Flores-Rodríguez, F. Trujillo-Romero, and W. Suleiman, "Object recognition modular system implementation in a service robotics context," in *2017 International Conference on Electronics, Communications and Computers (CONIELECOMP)*, 2017, pp. 1–6, doi: 10.1109/CONIELECOMP.2017.7891833.

[28] J. Maver and R. Bajcsy, "Occlusions as a guide for planning the next view," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 5, pp. 417–433, 1993, doi: 10.1109/34.211463.

[29] J. E. Banta, Y. Zhien, X. Z. Wang, G. Zhang, M. T. Smith, and M. A. Abidi, "Best-next-view algorithm for three-dimensional scene reconstruction using range images," in *Intelligent Robots and Computer Vision XIV: Algorithms, Techniques, Active Vision, and Materials Handling*, Oct. 1995, vol. 2588, pp. 418–429, doi: 10.1117/12.222691.

[30] A. García-Moreno and J. González-Barbosa, "Reconstrucción virtual tridimensional de entornos urbanos complejos," *Revista Iberoamericana de Automática e Informática industrial*, vol. 17, no. 1, pp. 22–33, 2020, doi: 10.4995/riai.2019.11203.

[31] R. Pito, "A solution to the next best view problem for automated surface acquisition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 10, pp. 1016–1030, 1999, doi: 10.1109/34.799908.

[32] S. Zhang, G. D. Sullivan, and K. D. Baker, "The automatic construction of a view-independent relational model for 3-d ob-

ject recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 6, pp. 531–544, 1993, doi: 10.1109/34.216723.

[33] Y. Ye and J. K. Tsotsos, "Where to look next in 3d object search," in *Proceedings of International Symposium on Computer Vision - ISCV*, 1995, pp. 539–544, doi: 10.1109/ISCV.1995.477057.

[34] S. D. Roy, S. Chaudhury, and S. Banerjee, "Isolated 3d object recognition through next view planning," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 30, no. 1, pp. 67–76, 2000, doi: 10.1109/3468.823482.

[35] S. Dutta Roy, S. Chaudhury, and S. Banerjee, "Recognizing large 3-d objects through next view planning using an uncalibrated camera," vol. 2, 02 2001, pp. 276–281 vol.2, doi: 10.1109/ICCV.2001.937636.

[36] S. Chen and Y. Li, "Automatic sensor placement for model-based robot vision," *IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics : a publication of the IEEE Systems, Man, and Cybernetics Society*, vol. 34, pp. 393–408, 03 2004, doi: 10.1109/TSMCB.2003.817031.

[37] J. Hernandez-Vicen, S. Martinez, and C. Balaguer, "Principios básicos para el desarrollo de una aplicación de bi-manipulación de cajas por un robot humanoide," vol. 10, no. 11, p. 1354, Jun. 2021, doi: 10.3390/electronics10111354.

[38] M. Tsuru, A. Escande, A. Tanguy, K. Chappellet, and K. Harad, "Online object searching by a humanoid robot in an unknown environment," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2862–2869, 2021, doi: 10.1109/LRA.2021.3061383.

[39] J. Denzler and C. M. Brown, "Information theoretic sensor data selection for active object recognition and state estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, pp. 145–157, 2002, doi: 10.1109/34.982896.

[40] S. Wenhardt, B. Deutsch, J. Hornegger, H. Niemann, and J. Denzler, "An information theoretic approach for next best view planning in 3-d reconstruction," in *18th International Conference on Pattern Recognition (ICPR'06)*, vol. 1, 2006, pp. 103–106, doi: 10.1109/34.982896.

[41] S. Kriegel, T. Bodenmüller, M. Suppa, and G. Hirzinger, "A surface-based next-best-view approach for automated 3d model completion of unknown objects," in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 4869–4874, doi: 10.1109/ICRA.2011.5979947.

[42] S. Kriegel, C. Rink, T. Bodenmüller, A. Narr, M. Suppa, and G. Hirzinger, "Next-best-scan planning for autonomous 3d modeling," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 2850–2856, doi: 10.1109/IROS.2012.6385624.

[43] S. Kriegel, M. Brucker, Z. Marton, T. Bodenmüller, and M. Suppa, "Combining object modeling and recognition for active scene exploration," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 2384–2391, doi: 10.1109/IROS.2013.6696691.

[44] M. Nieuwenhuisen, *et al.*, "Mobile bin picking with an anthropomorphic service robot," in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 2327–2334, doi: 10.1109/ICRA.2013.6630892.

[45] S. Peipei, L. Wenyu, Y. Ningjia, and D. Feng, "An intelligent vision localization system of a service robot nao," in *2015 34th Chinese Control Conference (CCC)*, 2015, pp. 5993–5998, doi: 10.1109/ChiCC.2015.7260577.

[46] A. Athar, A. M. Zafar, R. Asif, A. A. Khan, F. Islam, Yasar, and O. Hasan, "Whole-body motion planning for humanoid robots with heuristic search," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 4720–4727, doi: 10.1109/IROS.2016.7759694.

[47] C. Li and X. Wang, "Visual localization and object tracking for the nao robot in dynamic environment," in *2016 IEEE International Conference on Information and Automation (ICIA)*, 2016, pp. 1044–1049, doi: 10.1109/ICInfA.2016.7831973.

[48] L. Zhang, H. Zhang, H. Yang, G.-B. Bian, and W. Wu, "Multi-target detection and grasping control for humanoid robot nao," *International Journal of Adaptive Control and Signal Processing*, vol. 33, no. 7, pp. 1225–1237, Jul. 2019, doi: 10.1002/acs.3031.

[49] D. Paulius, K. S. P. Dong, and Y. Sun, "Functional object-oriented network: Considering robot's capability in human-robot collaboration," 2019. [Online]. Available: http://arxiv.org/abs/1905.00502

[50] I. Ofodile, A. Helmi, A. Clapés, E. Avots, K. Peensoo, S.-M. Valdma, A. Valdmann, H. Valtna-Lukner, S. Omelkov, S. Escalera, C. Ozcinar, and G. Anbarjafari, "Action recognition using single-pixel time-of-flight detection," *Entropy*, vol. 21, no. 4, p. 414, Apr. 2019, doi: 10.3390/e21040414.

[51] L. Zhang, H. Liu, C. Luo, G.-B. Bian, and W. Wu, "Target recognition of indoor trolley for humanoid robot based on piecewise fitting method," *International Journal of Adaptive Control and Signal Processing*, vol. 33, no. 8, pp. 1319–1327, Aug. 2019, doi: 10.1002/acs.2994.

[52] K. Othman and A. Rad, "An indoor room classification system for social robots via integration of cnn and ecoc," *Applied Sciences*, vol. 9, no. 3, p. 470, Jan. 2019, doi: 10.3390/app9030470.

[53] P. Alcantarilla, J. Nuevo, and A. Bartoli, "Fast explicit diffusion for accelerated features in nonlinear scale spaces," in *Procedings of the British Machine Vision Conference 2013*, 2013, pp. 13.1-13.11, doi: 10.5244/C.27.13.

[54] P. F. Alcantarilla, A. Bartoli, and A. J. Davison, "Kaze features," in *Computer Vision – ECCV 2012*, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 214–227.

[55] B. Fritzke, "Growing cell structures - a self-organizing network for unsupervised and supervised learning," *Neural Networks*, vol. 7, no. 9, pp. 1441–1460, Jan. 1994, doi: 10.1016/0893-6080(94)90091-4.

BIOGRAPHIES OF AUTHORS

Karen Lizbeth Flores Rodríguez Karen's professional training includes a Bachelor's Degree in Mechatronics from the Polytechnic University of Sinaloa in 2012. She completed a master's degree in Robotics at the Technological University of Mixteca, Oaxaca, in 2017. She is currently a Ph.D. student in Advanced Technology, in the area of image analysis at CICATA-IPN Querétaro. Her areas of interest include computer vision, robotics, automation, artificial intelligence, and education. She can be contacted at email: kfloresr1800@alumno.ipn.mx.

Felipe Trujillo-Romero received the Ph.D. degree in Informatics Systems from the Institut National Polytechnique de Toulouse France in 2008. At present, he is a full-time professor at the Department of Electronics Engineering, Universidad de Guanajuato. His research interests include Computer vision, Evolutionary Algorithms, Robotics, and Parallel Computing. He can be contacted at email: ftrujillo@mixteco.utm.mx.

José-Joel González-Barbosa has two master degree, the first one was received in Electrical Engineering from the University of Guanajuato, Mexico in 1998, and the second one was received in Signal, Image and, Acoustics from National Polytechnic Institute of Toulouse, France in 2000. He received his PhD. degree in Computer Science and Telecommunications from National Polytechnic Institute of Toulouse, France, in 2004. He is currently an Associate Professor at the CICATA-IPN, Mexico, where he teaches courses in Computer Vision, Image Processing, Pattern Recognition and Scientific Computing. His research interests include Multicamera Systems, 3D Computer Vision, Panoramic Vision, Object Recognition, Robotics, Augmented and Virtual Reality. He is affiliated with IEEE member. He can be contacted at email: jgonzalezba@ipn.mx.