

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Approaches to door identification for robot navigation

E. Jauregi, E. Lazkano and B. Sierra
University of Basque Country,
Donostia

1. Introduction

Even though great technical progress has been made in the area of mobile robotics, some fundamental control problems, such as autonomous navigation, remain unresolved. Many animals have shown that they are very good at navigating but autonomous navigation is still a complicated task for engineered robots. Therefore, research efforts have aimed at incorporating biologically inspired strategies into robot navigation models (Mallot and Franz, 2000; Trullier et al., 1997). Among these, *Local navigation strategies* allow the agent to choose actions based only on its current sensory input, while *way finding strategies* are responsible for driving the agent to goals out of the agent's perceptual range and require the recognition of different places and correlations among them. Acquiring the appropriate set of landmarks remains a challenge.

Many navigation tasks can be fulfilled by point to point navigation, door identification and door crossing (Li et al., 2004). Indoor semi-structured environments are full of corridors that connect different offices and laboratories where doors give access to many of those locations that are defined as goals for the robot. Hence, endowing the robot with the door identification ability would undoubtedly increase the navigating capabilities of the robot.

Doors in indoor environments are not necessarily uniform; door panels can be of different texture and colour and handles can vary in shape. Specific methods can be designed for each type of door. However, feature-based methods are more easily applicable to different object recognition tasks they are computationally expensive and therefore less attractive to be applied for real-time problems such as for mobile robot navigation. This chapter compares several approaches developed for door identification based on handle recognition, from specific methods based on colour segmentation techniques to more general ones that focus on feature extraction methods, like SIFT (Lowe, 2004) and (U)SURF (Bay et al., 2008). A new two-step multiclassifier that combines region detection and feature extraction is also presented. The objective of the approach is to extract the most relevant part of the image, the subimage that with high probability should contain the most interesting region of the image, and limit the application of the feature extraction techniques to that region. The developed algorithm improves the relevance of the extracted features, it reduces the superfluous keypoints to be compared at the same time that increases the efficiency by improving accuracy and reducing the computational time.

The developed approaches are first evaluated off-line and tested afterwards in a real robot/environment system using a a Peoplebot robot from *Mobilerobots*. The door recogni-

tion module is integrated in a behaviour-based control architecture (Brooks, 1986) that allows the robot to show a door-knocking behaviour.

The chapter is structured as follows: next section reviews the literature about door identification. Section 3 presents the first approach taken to solve the task: a three-stage procedure that relies mainly on colour segmentation. Section 4 summarises some feature extraction methods and shows the results obtained for the door recognition problem. Section 5 presents a new two-step algorithm that combines region detection and feature extraction that incorporate the applicability of the feature extraction methods to the performance of the environment specific ones. The obtained results are commented in section 6. Experiments performed on a real robot/environment system are described in section 7. Finally, conclusions and further work are outlined in section 8.

2. The door identification task

Several references can be found that tackle the problem of door identification. In (Kragic et al., 2002), where doors are located in a map, rectangular handles are searched for manipulation purposes using cue integration by consensus. However, most of the references we found are vision-based approaches (Seo et al., 2005) and focus on edge detection. Muñoz-Salinas et al. (2005) present a visual door detection system that is based on the Canny edge detector and Hough transform to extract line segments from images. Another vision-based system for detection and traversal of doors is presented in (Eberset et al., 2000). Door structures are extracted from images using a line based filtering parallel method, and an active tracking of detected door line segments is used to drive the robot through the door.

A different proposal for vision-based door traversing behaviour can be found in (Seo et al., 2005). Here, the PCA (Principal Component Analysis) pattern finding method is applied to the images obtained from the camera for door recognition. (Stella et al., 1996) presents SAURO, a robotic autonomous system oriented to transportation tasks in indoor environments. SAURO uses a vision-based self-localisation subsystem that allows the robot to determine its location to verify that the planned path is correctly followed. A door identification and crossing approach is also presented in (Monasterio et al., 2002), where a neural network based classification method was used for both, the recognition and crossing steps. More recently, in (Lazkano et al., 2007) a Bayesian Network based classifier was used to perform the door crossing task. Doors are assumed to be open, the opening is identified, and doors crossed using sonar sensor information.

But navigating in narrow corridors makes it difficult to identify doors by line extraction due to the inappropriate viewpoint restrictions imposed by the limited distance to the walls. The goal of the work presented in this chapter is to try to identify the doors by recognising the handles, extracting the necessary local features needed for robust identification. In the experiments here described the aim is to identify doors by means of the door handles in an environment with two type of handles: circular ones, located at doors with pladour (a type of laminated surface) and consequently, completely textureless blades; and rectangular ones located at wooden door blades. The problem is approached from two angles. The first one relies on the segmentation of the door blade surfaces using colour segmenters for each type of door. The second approach is based on feature extraction methods and aimed at finding a more general algorithm easily applicable to other types of doors. Both approaches are vision based and are tackled from a classification problem perspective: given an image, the goal is to determine whether the image contains or not a door handle.

Although there are tasks where images may not be restricted to a single object identification, and thus could contain more than one region to be detected and extracted, that is not the case in the problem described in this chapter, where it is assumed that doors contained a single handle.

3. Segmentation based identification

Monochrome image segmentation algorithms are based on two basic properties of grey levels: discontinuities and similarities. Discontinuity based segmentation means partitioning the image by detecting abrupt changes in grey levels. On the other hand, similarity based segmentation pretends to group regions with similar properties (Gonzalez and Woods, 1993). *Discontinuity detection* is performed by looking for three different types of discontinuities: isolated points, lines and edges. Points and lines are not too frequent for most practical applications. On the contrary, edges represent the frontiers among regions of different grey level properties and are the most widely applied segmentation technique. But discontinuity detection methods rarely characterise a border in a unique manner and therefore, additional local or global methods are needed to join pixels associated to the edges.

Among the methods that apply *similarity detection*, *thresholding* is probably the most important technique. But it is known that variant illumination affects the result of the segmentation. Thereby, several methods allow to change the threshold levels making it adaptive. On the other hand, *region growing* techniques consist of grouping pixels or regions into bigger regions. Alternatively, *region splitting* techniques start with an arbitrary number of regions in an image and try to group or divide them into regions that comply with certain properties.

It is noticeable the large number of grey level segmentation techniques. On the contrary, colour image segmentation requires more information about the objects to be identified. Colour image segmentation extracts one or more connected regions from an image and the extracted regions must satisfy certain uniformity criteria based on characteristics derived from the spectral components (Skarbek and Koschan, 1994).

Some colour segmentation techniques are based on mathematical models and some are algorithmic approximations. Basically, there are *histogram based* techniques, which try to identify peaks on the colour space and then use these extrema to classify pixels; *Clustering* techniques that group pixels according to the obtained cluster representatives; and *Fuzzy Clustering techniques* that use fuzzy functions to decide the membership of every pixel in each of the defined cluster.

3.1 The door identification process

In order to identify images containing a door handle a three-stage algorithm has been designed. This three-stage process proceeds as follows:

1. Region detection: this step is designed to approximate the handle area, if any, in the image. The most intuitive approach to detect circular handles seems the extraction of circular edges. The *Circular Hough Transform* is an appropriate method to achieve this objective. But handles can also be non circular. A more general method is needed as *blob* extraction also known as region detection or labelling.
2. Colour segmentation: using the position information of the previous step, the surroundings of the candidate object are segmented in order to see if the candidate handle is well surrounded by the appropriate door blade.

3. Statistical validation: measures of the segmented image are afterwards needed to classify an image as containing or not a handle.

3.1.1 Region detection

As mentioned before, some handles are assumed to be circular in shape. Therefore, circle detection can be performed to locate handles in the images taken by the robot. Although many circle extraction methods have been developed, probably the most well-known algorithm is the Circle Hough Transform (CHT). Moreover, Hough transform methods have shown to be robust enough to deal with noisy images (Ayala-Ramírez et al., 2006).

The use of the Hough Transform to detect circles was first outlined by Duda and Hart (1972) and then, Kimme et al. (1975) gave probably the first known application of the Hough transform to detect circles in real images. Later on, Yuen et al. (1990) investigated five circle detection methods which are based on variations of the Hough Transform. One of those methods is the *Two stage Hough Transform* and it is implemented in the OpenCV vision library (<http://www.intel.com/research/opencv>) used in the experiments described later on.

The circle finding function can identify a huge number of circles depending on the image background. From the returned list of circles, only the most probable one is considered. However, due to the local navigation strategies of the robot the images will be obtained within the same distance range and therefore, it is possible to know in advance the approximate radius of the handles. In this manner only the identified circumferences with a radius that lies within a known range would be considered as handle candidates.

The CHT detects circular shapes; hence, an alternative method is needed in order to first approximate the non circular handles. The method can be generalised by scanning the image for continuous connected regions or *blobs*. A blob (binary large object) is an area of touching pixels with the same logical state. Blob extraction, also known as region detection or labelling, is an image segmentation technique that categorises the pixels in an image as belonging to one of many discrete regions. The process consists of scanning and numbering any new regions that are encountered, but also merging old regions when they prove to be connected on a lower row. Therefore, the image is scanned and every pixel is individually labelled with an identifier which signifies the region to which it belongs (see (Horn, 1986) for more details).

Blob detection is generally performed on the resulting binary image from a thresholding step. Instead, we apply the SUSAN (Smallest Univalued Segment Assimilating Nucleus) edge detector (Smith and Brady, 1997), a more stable and faster operator.

Again, the blob extraction process can give many different regions for a single image. In order for a blob to be confirmed as a candidate, the result of the blob detection process should be filtered and false positives should be discarded. Different filtering techniques can be used. For instance, in (Ye and Zhong, 2007) location-related pixel information is used for blob discrimination where the aim is to count persons in images taken by a surveillance system. A similar approach is used in our proposal where blobs that are not consistent with the defined size restrictions are discarded. The size restrictions depend on the distance the images are taken from.

3.1.2 Colour segmentation

The region extraction step is not reliable enough, neither to confirm, nor to reject, the presence of a handle in an image. Therefore, it needs to be endowed with some complementary processes in order to improve its performance. The approach used here is to use colour information around the location of the detected shape for door recognition. The objective of

this second step is to segment the pixels belonging to the door blades in an image. Candidate handles not surrounded by the proper blades should help to reject false positives.

Within the robot environment, a circular handle is always associated to a pladour door and rectangular ones are on wooden doors presenting different tonalities according to lighting conditions –e.g. presence or absence of electric or natural lighting.

3.1.2.1 Segmenting pladour door blades

A supervised machine learning approach has been selected for this goal. To build the classifier we chose *Oc1* (Oblique Classifier 1) (Murthy et al., 1994), a decision tree induction system well suited for applications where the instances have numeric feature values. *Oc1* builds decision trees containing linear combinations of one or more attributes at each internal node. The trees built in this way partition the instance space with both oblique and axis-parallel hyperplanes. Images taken by the robot are represented in RGB colour space and thereby, each pixel is a three component vector, each component taking a value that ranges from 0 to 255.

In every classification problem, a training set is required to get a model to be later used when a new instance is presented to the model. To get the training set, we firstly constructed a database of positive instances (those associated to pladour doors), as well as negative instances (those not associated to pladour doors). The size of these databases was about two million pixels, obtained from about sixty images taken from the robot camera in a corridor. From these databases we extracted, 80,000 pixels randomly, 40,000 of them labelled as *pladour* and the remaining 40,000 as *not pladour*. Then, these 80,000 pixels were presented to the *Oc1* tree generation procedure, to get the decision tree used in the segmentation of the images taken by the robot camera. The obtained performance after applying 10 fold crossvalidation to this database was 93.61%. Fig. 1 shows several examples of this segmentation process. The first row contains images with handles, although the variant lighting conditions affect the pladour colour recognition process and therefore, the segmentation process. Notice for example the upper right corner of the fourth image. The bottom row shows images without handles. Original images contain the found circles detected by the Hough transform, in spite of the radius exceeds the preset threshold, for sake of clarity. It can be observed that the segmentation of these images does not imply the existence of any handle.

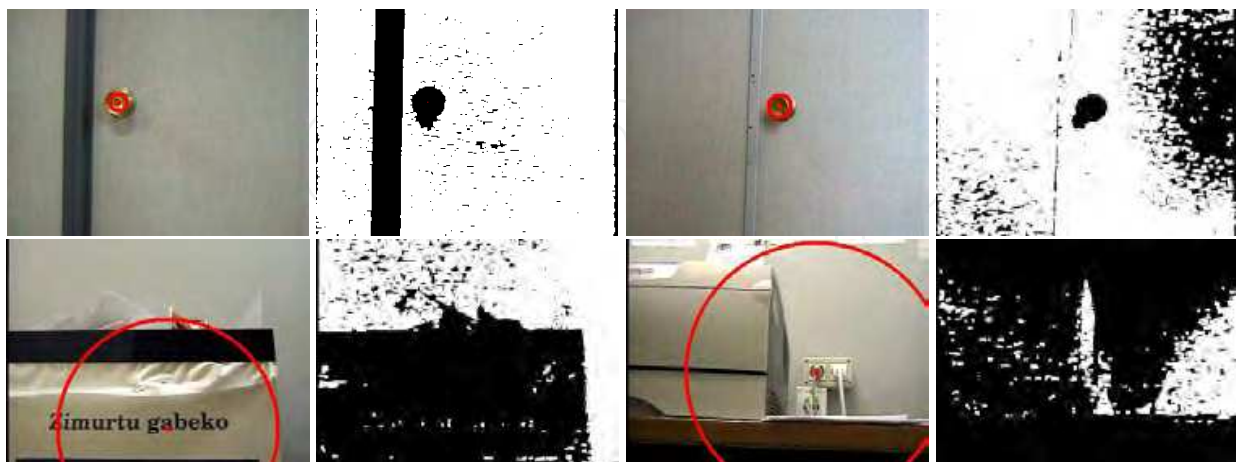


Fig. 1. Pladour segmentation examples

3.1.2.2 Segmenting wooden door blades

The goal now is to classify pixels belonging to wooden surfaces. To segment wooden surfaces a specific thresholding algorithm has been designed. First of all, the following reference values are selected from an image containing uniquely wooden coloured pixels using the normalised RGB colour space:

- R_{min} : minimum value of the red channel.
- G_{min} : minimum value of the green channel.
- B_{min} : minimum value of the blue channel.
- $Rdiff_{min}$: value of the minimum difference among the red and the maximum of the green and blue.
- $Gdiff_{min}$: value of the minimum difference among the green and the maximum of the red and blue.
- $Bdiff_{min}$: value of the minimum difference among the blue and the maximum of the green and red.

Then, given a new image, a pixel will be labelled as belonging to a wooden surface according to the algorithm described in figure 2. The process is performed using the normalised RGB colour space.

```

wood_segementer(R, G, B,
                R_min, Rdiff_min, G_min, Gdiff_min, B_min, Bdiff_min)
{
    Rdiff = R - max(G, B)
    Gdiff = G - max(R, B)
    Bdiff = B - max(R, G)
    if (R > R_min && G > G_min && B > B_min &&
        Rdiff > Rdiff_min &&
        Gdiff > Gdiff_min &&
        Bdiff > Bdiff_min)
        return 1;
    return 0;
}

```

Fig. 2. Wooden surface segmentation algorithm

Figure 3 shows some examples of this segmentation process.

3.1.3 Statistics for decision taking

So far we are provided with two procedures to recognise the handle and its surroundings. However, both procedures are prone to errors due to noise, lighting conditions and other objects in the environment (printers, dustbins, tables, and so on). We cannot rely in the accuracy of any of them separately. The aim of this third step is to analyse segmented pixels of the candidate handle and those surrounding the circle in order to definitively confirm or reject the handle candidate. The segmentation process yields a black-and-white image, where white

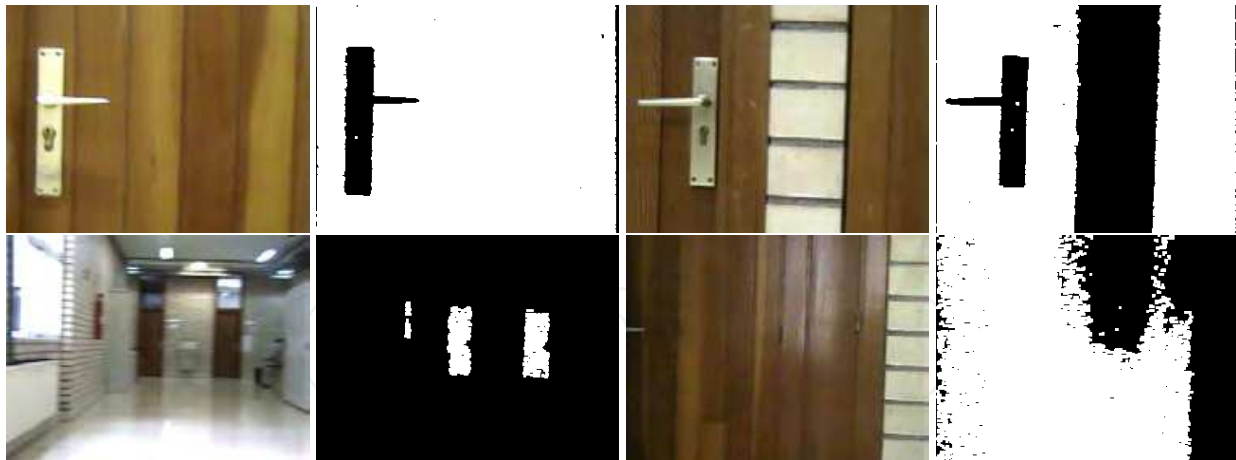


Fig. 3. Segmentation of wooden surfaces

points are those classified as *door blade*, and black ones are those classified as not belonging to the *door blade*. To analyse the surroundings of the prospective handle, and using the information about the detected region, the centre (x_0, y_0) is obtained and then, the following values are calculated (see Figure 4):

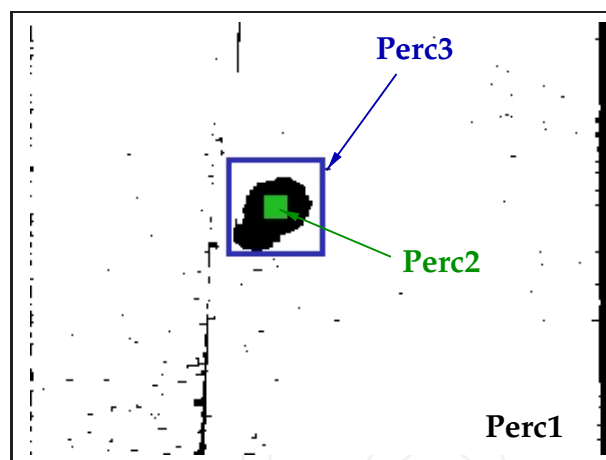


Fig. 4. Zones from which *Perc2* and *Perc3* are computed; *Perc1* is computed over the whole image

- *Perc1*: Percentage of white points in the whole image. When the robot is close to a door, a huge percentage of the pixels in its visual field are likely to be segmented as white. So, the bigger *Perc1*, the more likely the robot is in front of a pladour door.
- *Perc2*: The pixels around the centre should belong to the handle, not to the pladour class. This value represents the percentage of black points in the 5×5 grid centred at (x_0, y_0) . Therefore, the bigger *Perc2*, the more likely the robot is in front of a handle.
- *Perc3*: When the procedure that returns the region has really located a handle, the farther surroundings of the centre are expected to be segmented as white, as they do not fall in the handle, but in the door blade. We define:

- S_1 : set of points in the squared area centred at (x_0, y_0) and of size length $2 \times r$.
- S_2 : set of points in the squared area centred at (x_0, y_0) and of size length $2 \times (r + d)$, where d represents the desired shift from circle perimeter.

Hence, $S = S_2 - S_1$ defines the closest surroundings of the handle and $Perc3$ represents the percentage of white points (*pladour*) in S . Again, the bigger $Perc3$, the more likely the robot is in front of a handle.

The CHT gives the radius of each detected circle, together with some more information, whereas using region labelling r is defined as the maximum value between the width and the height of the obtained region.

The combination of these percentages (weighted mean) give us a measure of confidence (cl_{handle}) of being in front of a door and that the region recognition procedure has really recognised a circular handle.

3.2 Results

At this stage of the experimentation, the handle recognition was performed for each class of handle, using separate databases for each one.

All the images (learning and testing DBs) were taken while the robot followed the corridors using its local navigation strategies and therefore, they were taken at distances at which the robot is allowed to approach the walls. Both databases contained almost equal positive and negative cases.

The databases characteristics were the following:

1. For circular handles, the database contained about 3000 entries
2. For rectangular handles, the size was bigger with about 5000 images

The experiments were executed for a maximum allowed radius of 20 pixels ($r_{max} = 20$), a shift of 15 pixels from the circle perimeter ($d = 15$), and the confidence level of being in front of a handle should rise above 0.6 to confirm the door ($cl_{handle} > cl_{TH} = 0.6$).

In order to evaluate the improvement introduced by the three-stage process, the classification accuracy was computed for the sequence and compared with the accuracy that would be obtained if only region detection was used.

However, accuracy is considered a fairly crude score that does not give much information about the performance of a categoriser. The F1 measure, also known as F-score or F-measure, is a measure of a test's accuracy and can be viewed as a weighted average of the precision and recall (see equation 1), where an F1 score reaches its best value at 1 and worst score at 0. F1 measure combines both precision and recall into a single metric and favour a balanced performance of the two metrics (Chen, 1996).

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (1)$$

Table 1 shows the obtained classification accuracies and the corresponding F1 measure values. These results show how both performance measures, the classification accuracy and the F1 value are highly increased using the three stage approach for both types of handles. Surprisingly, the blob detector shows to be more appropriate for the circular handles. The robot's viewpoint affects the shape of the handle in the sense that the circular handle distorts and often does not look like a circle but more like an ellipsoid making it more difficult to be identified by the CHT.

Method	Circular		Rectangular	
	Acc.	F1	Acc.	F1
CHT	30.6	0.319	–	–
CHT 3-stage	85.1	0.685	–	–
Blobs	91.1	0.867	61.0	0.36
Blobs 3-stage	96.7	0.946	93.6	0.72

Table 1. Experimental results

The door identification process should be performed in real time by the robot. Hence, the time needed to compute the identification must be analysed. Table 2 shows the time requirements of the developed three-stage algorithm.

Method	Circular	Rectangular
CHT 3-stage	0.043	–
Blobs 3-stage	0.050	0.057

Table 2. Computational payload (s) for processing an image applying the 3-stage procedure

4. Feature extraction based identification

The approach described above showed to be too specific to the robot's particular environment and not easily generalisable. Although the region detector is robust enough, the segmentation processes are specific and new ones should be developed in order to detect handles located in different surfaces.

Feature extraction in images is an important issue in mobile robotics, as it helps the robot to understand its environment and fulfil its objectives. There are several applications where it is mandatory to recognise objects or scenes: image retrieval (Ledwich and Williams, 2004), mobile robot localisation (Gil et al., 2005);(Tamimi et al., 2006) and SLAM (Se et al., 2002), physical sign recognition (Roduner and Rohs, 2006) and automatic guidance of vehicles (Primdahl et al., 2005). Although different sensors can be used, vision turns out to be the most appropriate one due to the rich information that can be extracted from images. However, object identification becomes a complex task specially because of varying environmental conditions and changes in object scale and camera viewpoint.

Objects can be identified in images extracting local image descriptors. These descriptors should be distinctive and invariant to image transformations. The idea is to detect image regions co-variant to a class of transformations, which are then used as support regions to compute invariant descriptors (Mikolajczyk and Schmid, 2005). While complete invariance has yet to be achieved, features which are robustly resilient to most image transforms have been proposed by D. G. Lowe (Lowe, 2004).

Several invariant feature extraction techniques exist in the literature. SIFT (*Scale-invariant feature transform*) (Lowe, 2004) attempts to extract distinctive features from an image in order to find correspondences among different images or to identify objects. SIFT keypoints are nowadays the most common descriptors used. The features extracted by the algorithm are invariant to scale and rotation, and partially invariant to changes in illumination.

SIFT features are extracted according to the following procedure:

1. Detect scale-space extrema, searching over all scales and image locations. Potential interest points invariant to scale and orientation are efficiently computed using a DoG (differential of Gaussian) function.
2. Localise keypoints, detecting local extrema and removing low contrast points or candidates located in edges.
3. Assign orientations to the candidate keypoints based on local image gradient directions.

After keypoints are localised, for each keypoint a descriptor is computed by calculating a histogram of local oriented gradients around the interest point and storing the bins in a 128 dimensional vector. These descriptors can be then compared with stored ones in a reference database for object recognition purposes. But SIFT suffers from high computational payload and for on-line applications, each one of the three steps (detection of local extrema, keypoint description computation and keypoint matching) should be computed faster.

SURF (Bay et al., 2006) descriptors are computed in two steps. First, a reproducible orientation is found based on the information of a circular region around the interest point. This is performed using Haar-wavelet responses in the x and y directions at the scale where the interest point was detected. The dominant orientation then is estimated by calculating the sum of all responses within a sliding window. Next, the region is split up into smaller square subregions and some simple features are computed (weighted Haar wavelet responses in both directions, sum of the absolute values of the responses). This yields a descriptor of length 64, half size of the original SIFT descriptor and hence, offers a less expensive computationally matching process.

The upright version of SURF, named **USURF**, skips the first step of the descriptor computation process, resulting in a faster version. USURF is proposed for those cases where rotation invariance is not mandatory.

In all the three methods, the matching process requires a matching criteria and a reference database where the keypoints of the interesting objects must be stored.

4.1 Extracting features from door images

As mentioned previously, in our robot's environment circular and rectangular handles are distinguished. Circular ones are located into pladour blades. These have the advantage that almost every keypoint is located at the door handle and only a few of them appear at the handle surroundings (see Figure 5(a)). On the other hand, rectangular handles are located into wooden door blades that are not textureless and therefore, keypoints appear outside the handle (see Figure 5(b)).

In order to see the adequateness of the feature extraction techniques for the handle identification task, the same databases used for the previous experiments were used, albeit a few extra images were taken to use as reference objects (reference DBs). It must be noted that the test cases were collected in a different environment from where reference cases were taken. Therefore, the test databases did not contain images of the handles in the reference database. The databases characteristics were the following:

1. For circular handles, the testing database contained about 3000 entries and the reference database contained 32 images.
2. For rectangular handles, the testing database contained about 5000 images and the reference database contained 19 images.

The keypoint matching criteria used in these experiments is the 1-NN, the same proposed in (Lowe, 2004).

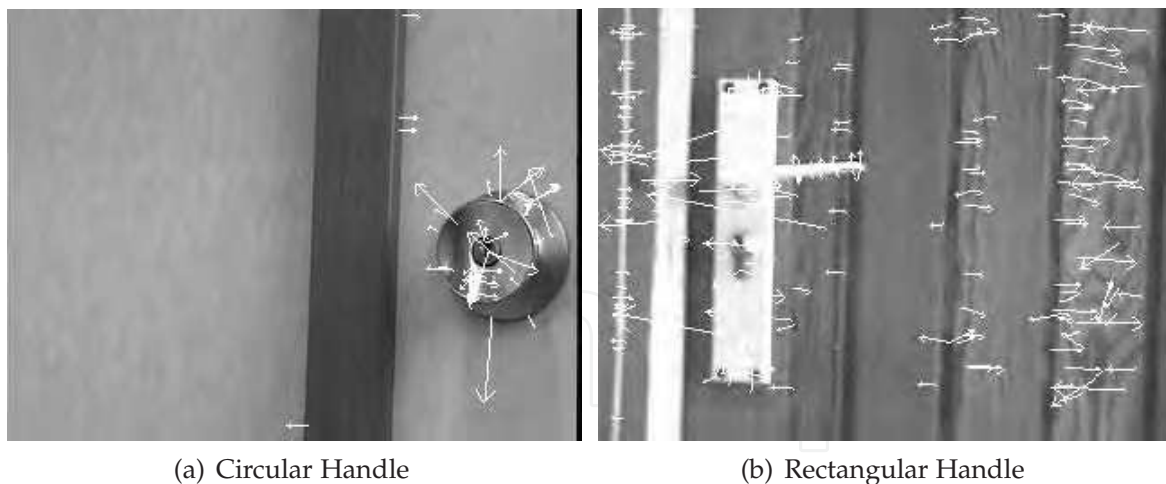


Fig. 5. SIFT keypoints in 320×240 images

4.2 Results

Table 3 shows the obtained results. Again, accuracy and F1 measure are calculated. USURF outperforms SIFT in both, accuracy and F1 measure in the case of circular handles, but its performance degrades for rectangular ones.

	Circular		Rectangular	
	Acc.	F1	Acc.	F1
SIFT	62.39	0.592	55.41	0.362
SURF	72.5	0.691	42.34	0.324
USURF	72.5	0.691	47.0	0.344

Table 3. Performance of the keypoint extraction methods

These values are disappointing. The feature extraction approach does not give reasonable performance values to be applied on the robot for navigation purposes. Comparing these results with the ones shown in table 1, both the classification accuracy and the F1 measure are much lower than the values obtained using the three-stage procedure previously described.

5. Combining region extraction and feature extraction

This section presents a new two-step algorithm that aims at reducing the superfluous keypoints obtained by methods like SIFT and (U)SURF by firstly extracting the region of the image where the object or objects to be identified are likely to be located.

As mentioned before, instead of computing the invariant features of the whole image, the approach presented here aims to reduce the size of the image to be processed by extracting the portion of the image that, will have a high probability of containing crucial features. Here on, we will call that portion the region of interest (ROI) of the image.

Several methods can be used for extracting the ROI but the experiments performed while developing the segmentation based door identifier showed that blob extraction gives robust results for the handle identification goal.

Once the most interesting blob is located, blob's length and width values are used to find its centre and afterwards, a square subimage is extracted. The size of the square is determined by the maximum value between the length and width of the candidate blob. The subimage is then scaled to a fixed size to obtain the portion of the image with a high probability of containing the object of interest, i.e a ROI. Then, the keypoint extraction and matching procedure is performed to the extracted ROI. Figure 6 summarises the process.

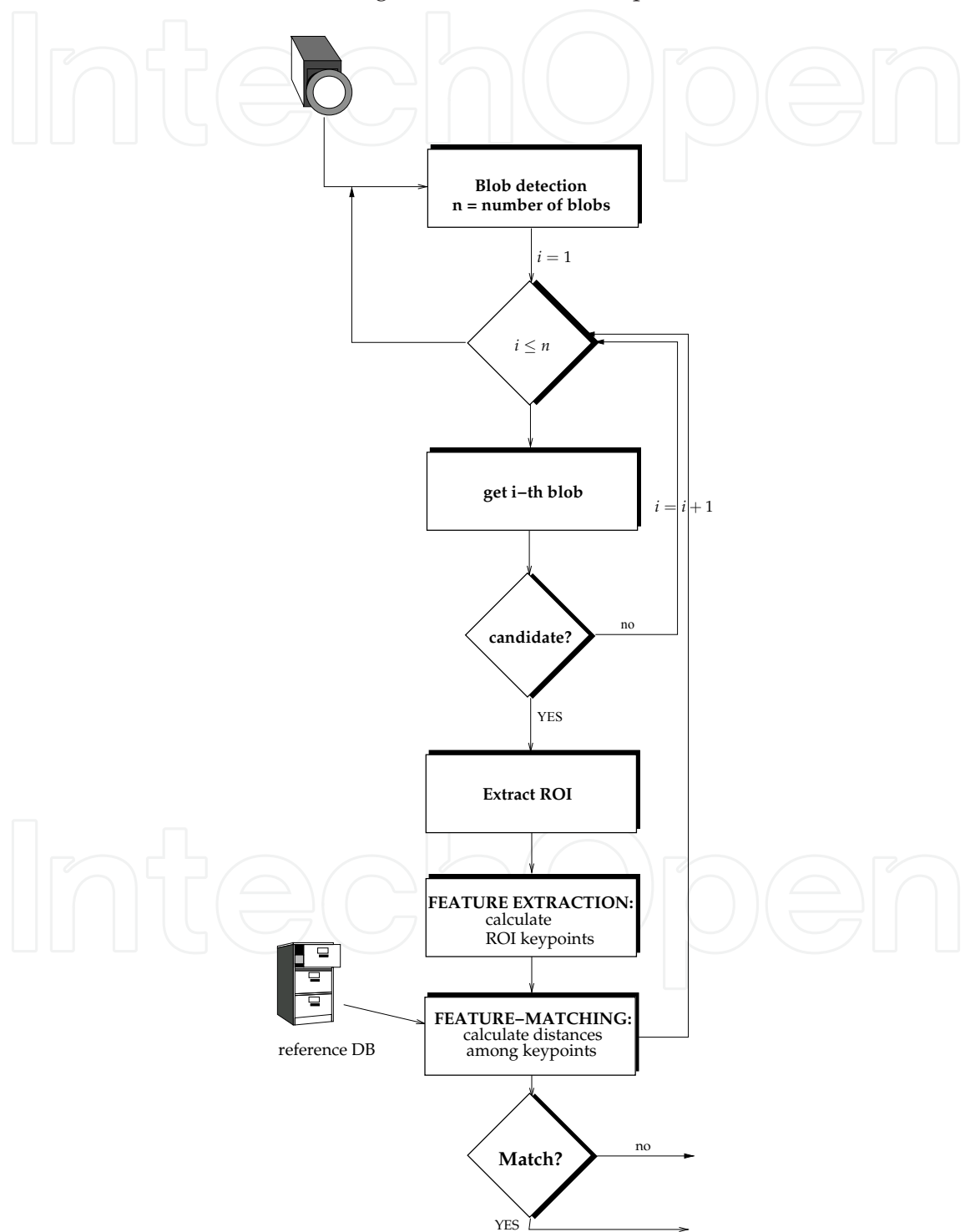


Fig. 6. Flow diagram of the new algorithm

Although it is not the case for the problem stated in this paper, the algorithm is drawn in a general form where more than one object can be searched in an image and hence, more than one ROI should be extracted and afterwards, processed for identification purposes.

5.1 Application to handle identification: results

Figure 7 shows examples of the result of extracting the ROI for both handle types. Obviously, the rectangular handle identification problem is more difficult due to the non symmetrical property of the handle itself and the featured blades where they are located.

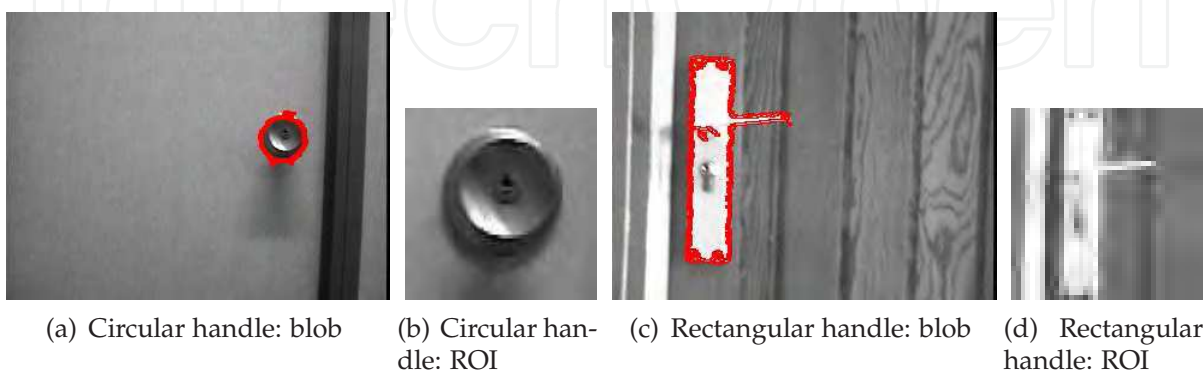


Fig. 7. Blob extraction and ROI scaling

Table 4-a) shows the improvement obtained applying the method proposed in the paper. Again, USURF seems to be the most adequate feature extraction method for circular handles, whereas SIFT is clearly the best approach to be applied to the obtained ROI when non circular handles must be identified. The accuracy is increased in almost 30% for SIFT and about 20 – 22% for SURF and USURF respectively for the case of circular handles. The improvement is better for rectangular handles, with an increase of 35% for SIFT and about 40% for SURF. Notice that this improvement is also reflected in the F1 measure, proving the adequateness of the methodology.

	Circular			Rectangular		
	acc.	size	F1	acc.	size	F1
SIFT	91.82	150	0.863	91.52	80	0.669
SURF	92.94	100	0.890	87.77	40	40.6
USURF	94.35	150	0.911	87.77	40	0.406

a) Region labelling + feature extraction

	Circular			Rectangular		
	acc.	size	F1	acc.	size	F1
SIFT	94.48	240	0.909	92.67	80	0.699
SURF	95.70	80	0.931	89.87	100	66.84
USURF	96.09	150	0.936	89.71	100	0.666

b) Region labelling + feature extraction + MR

Table 4. Best results

As mentioned in (Pfeifer and Bongard, 2006), vision may take advantage of the physical interaction of the agent with its environment. Taking into account the robot's morphology and the environmental niche, more specifically, the height at which the camera is mounted on the robot, and the height at which the handles are located on the doors, the handles should always appear at a specific height on the image. The improvement introduced by this *morphological restriction* (MR) is also showed in Table 4-b).

6. Discussion

Figures 8 and 9 show the evolution of the performance, for circular and rectangular handles. Both, the classification accuracy and the F1 measure are plotted when increasing the ROI size. These figures correspond to the proposed two-step method together with the morphological restriction previously mentioned. It can be appreciated how USURF is the best approach for circular handles and, on the contrary, SIFT improves speeded-up variants when rectangular handles need to be detected, even for larger ROI sizes.

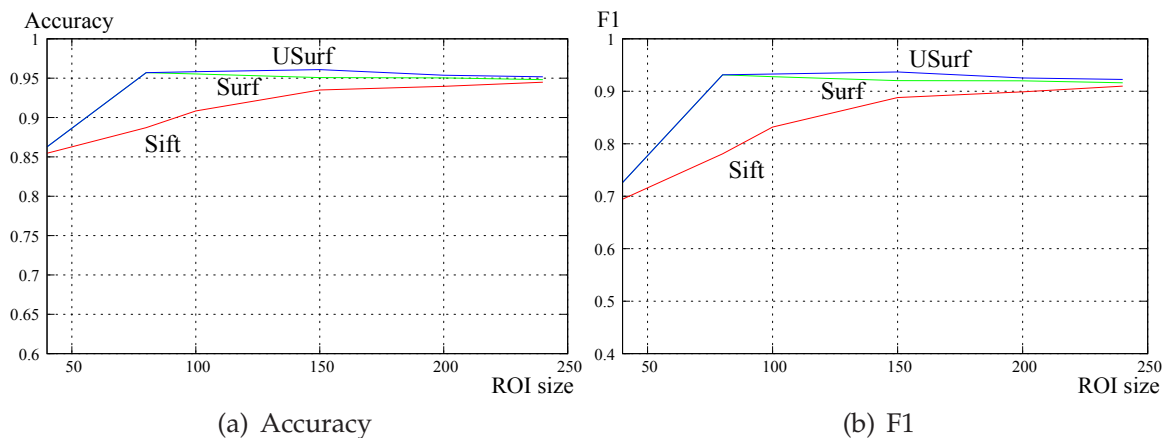


Fig. 8. Performance on circular handles while varying the ROI size

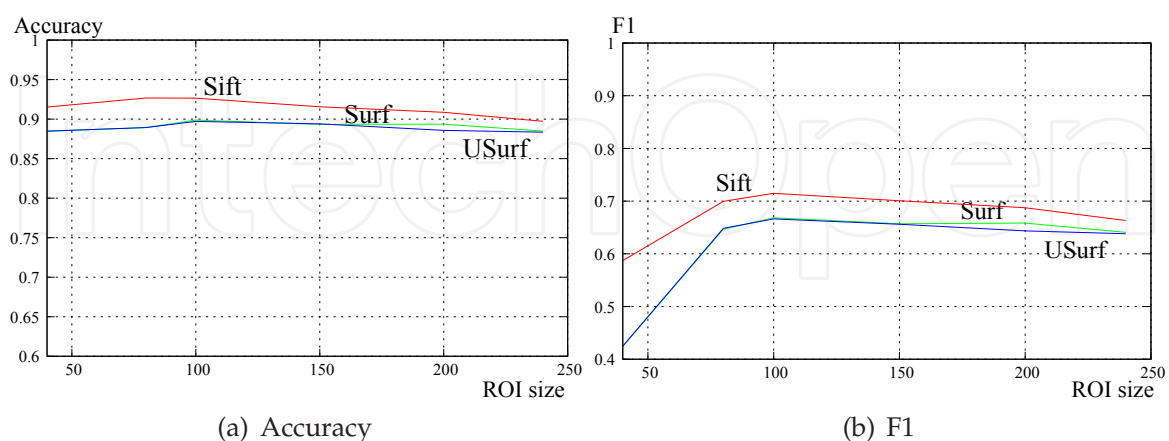


Fig. 9. Performance on rectangular handles while varying the ROI size

Performances are comparable to those obtained with the three-stage approach. Table 5 reflects how the average number of keypoints increases together with the ROI size. These values were calculated using the reference databases used in each experimental trial

and hence, are only tentative values. Although the number of keypoints increases together with the ROI size, stable keypoints are found on small image sizes. Increasing the scale of the ROI gives a higher number of keypoints but the repeatability of these new keypoints does not seem good.

Ref. DB	40	80	100	150	200	320 × 240
Circular	10.09	21.43	41.5	74.03	105.53	49.28
Rectangular	5.95	23.47	38.79	72.10	97.78	139.89

Table 5. Average number of keypoints

Table 6 shows the average time needed to process a single image using SIFT (blob location, keypoint extraction and matching against the reference database). Note that the time needed is sensibly higher for larger ROI sizes due to the larger number of keypoints that occur in those images. Moreover, the computational requirements for processing a single image is larger than the 20 fps attainable with the three-stage process; the same frequency can be reached only using ROI size of 40.

Database	40	80	100	150	200	No ROI
Circular	0.06	0.18	0.22	0.46	0.75	0.31
Rectangular	0.06	0.13	0.17	0.42	0.58	0.89

Table 6. Computational payload (s) for processing an image using SIFT

To summarise, although the off-line experimental step showed a degraded accuracy for the ROI of size 40 extracted using SIFT for circular handles, the short time needed to compute the identification and the better performance of SIFT for the rectangular handle identification problem that proved more difficult, makes it more appealing for the real time problem stated in this paper.

7. Experiments with the robot

Tartalo is a PeopleBot robot from MobileRobots, provided with a Canon VCC5 monocular PTZ vision system, a Sick Laser, several sonars and bumpers and some other sensors. Player-Stage (Gerkey et al., 2003) is used to communicate with the different devices and the software to implement the control architecture is SORGIN (Astigarraga et al., 2003), a specially designed framework that facilitates behaviour definition. To evaluate the robustness of the handle identification system developed, it has been integrated in a behaviour-based control architecture that allows the robot to travel across corridors without bumping into obstacles. When the robot finds a door, it stops, turns to face the door and knocks it with its bumpers a couple of times asking for the door to be opened and waiting for someone to open it. If after a certain time the door is still closed, *Tartalo* turns back to face the corridor and carries on, looking for a new handle. On the contrary, if someone opens the door the robot detects the opening with its laser and activates a door crossing behaviour module that allows it to enter the room. All the computation is carried out in its on-board Pentium (1.6GHz).

The overall control architecture is composed of nine threads that communicate and activate/deactivate each other when appropriate (see Figure 10). These nine threads can be grouped as follows:

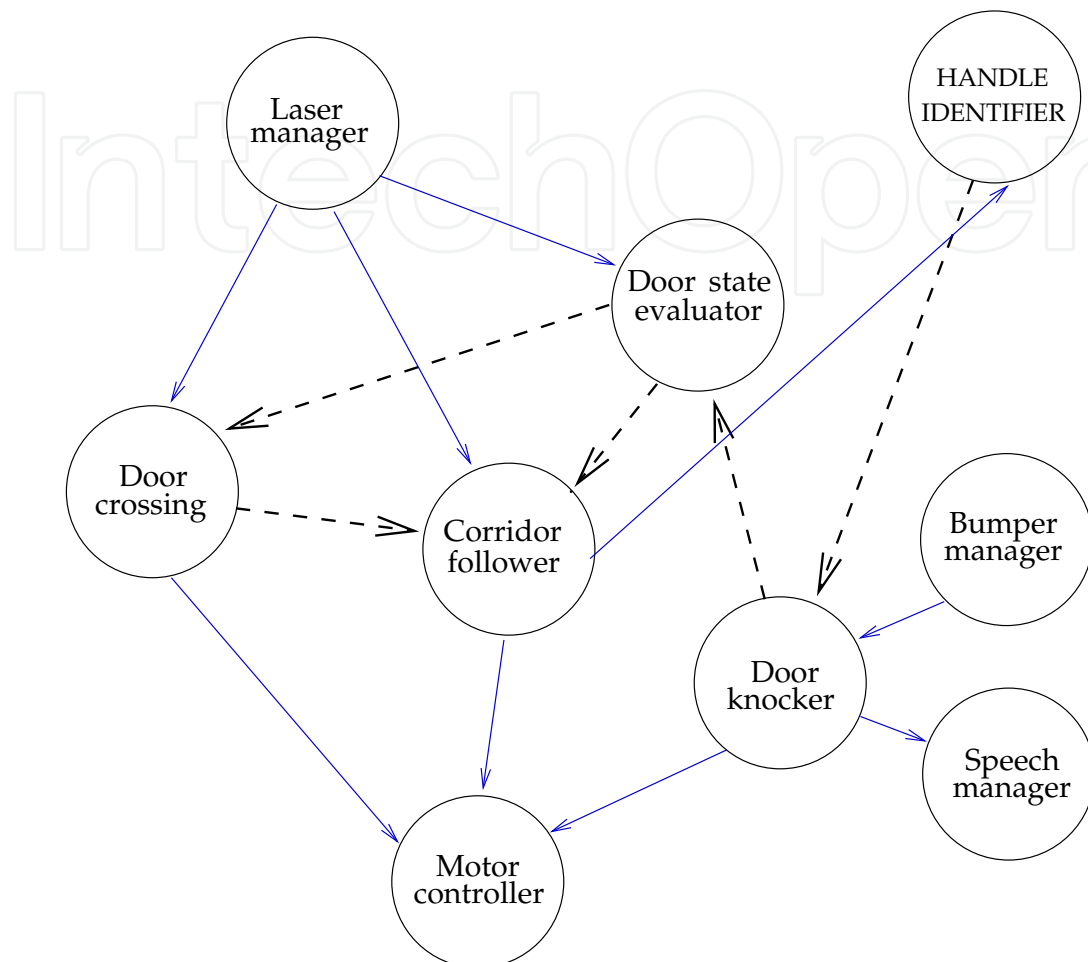


Fig. 10. Behaviour modules and communication

- Four device managers: these modules are not behaviour modules themselves, but they are needed to get/send up-to-date data from/to physical devices. These managers are needed for laser, bumper, speech and motor control.
- Three local navigation strategies: *corridor following*, *door crossing* and *door knocker* behaviours.

The *corridor following* behaviour represents the main local navigation strategy and its aim is to balance free space on its sides, in such a way that corridors are travelled in smooth trajectories, while avoiding obstacles at the same time.

- Two landmark identification subsystems: *handle identification* and *door state evaluator* subsystems. This last module calculates the difference between the reading taken just after the robot stops for the waiting period and the current reading, and decides if the door has been opened according to that difference.

Dashed lines in Fig.10 represent behaviour activation/deactivation links, while thin links are data communication connections among modules. Only most relevant connections are drawn for sake of clarity.

As mentioned previously, experiments within the real robot/environment system were performed using a ROI size of 40 and applying the SIFT feature extraction method. Also, to make the behaviour more robust, instead of relying on a single image classification, the robot will base its decision upon the sum of the descriptor matches accumulated for the last five consecutive images. Experiments were carried out in three different environments.

7.1 Environment 1: circular handles

Figure 11 shows the environment together with the evolution of the sum of the matching keypoints over time. The horizontal line represents the value at which the threshold was fixed. The 18 doors present in the environment were properly identified and no false positives occurred.

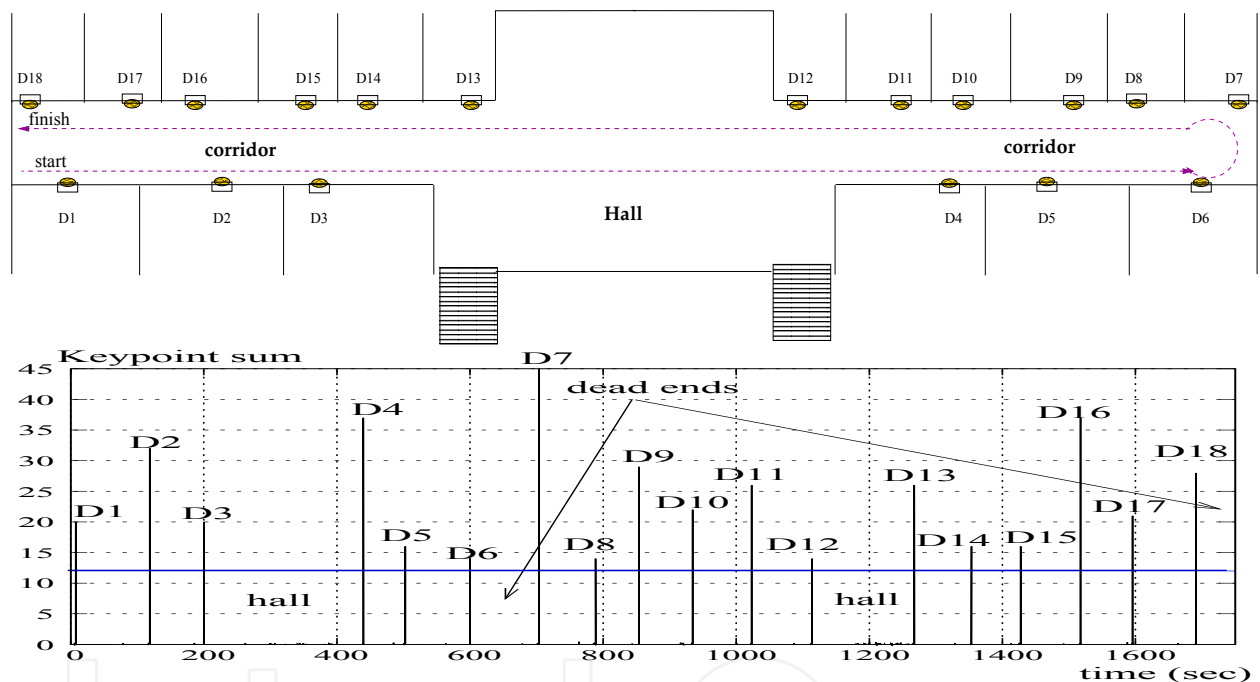


Fig. 11. First floor results

7.2 Environment 2: rectangular handles

Second floor of the building, lower corridor. 39 rectangular handles were consecutively to be identified. Figure 12 shows the original environment and the evolution of the keypoint sum over time.

The robot started on the left side of the corridor, with its camera pointing to its right and travelled all the way along the corridor successfully fulfilling the sequence of 6+7+7 handle identification, and then turned at the dead end. In its way back, only one of the handles of the central sector of the corridor, the one marked with a circle in the upper image of figure 12, was not recognised (6+6+6) and again, no false positives were given. Hence, a success ratio of 0.97 was achieved.

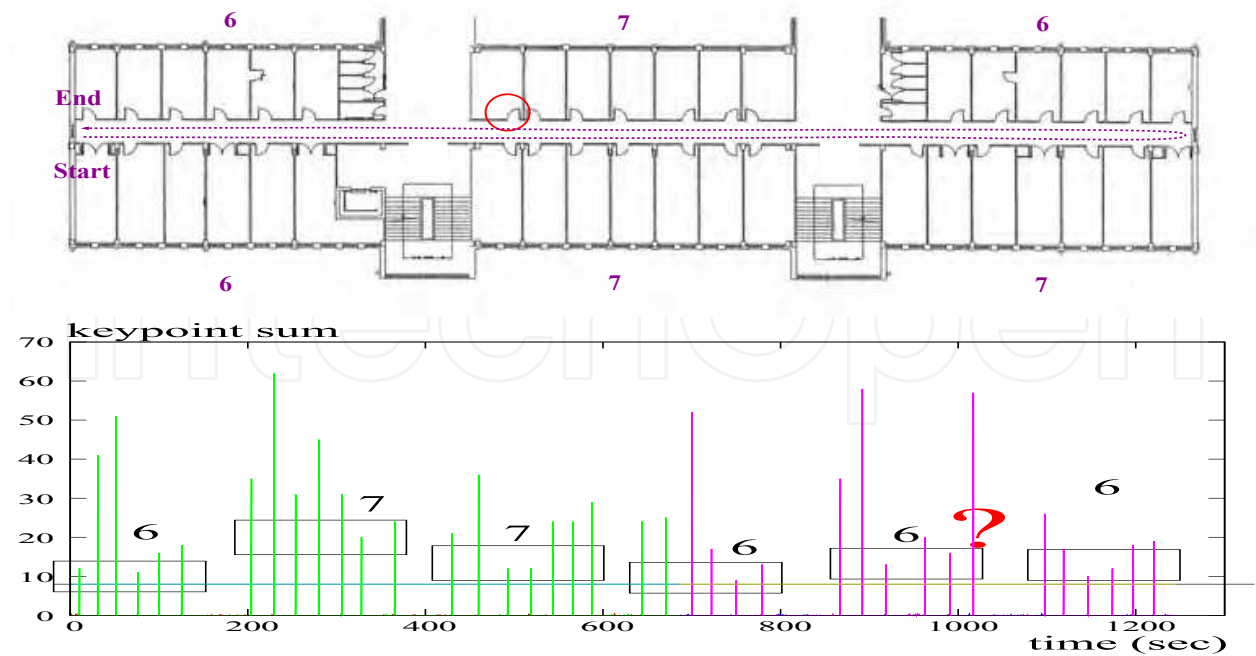


Fig. 12. Second floor results

7.3 Environment 3: mixed identification

Mixed handle identification over time. Experiments were performed on a part of the third floor of the faculty (figure 13). Three circular handles and three rectangular handles were to be identified. The robot was left running for three rounds in the corridor.

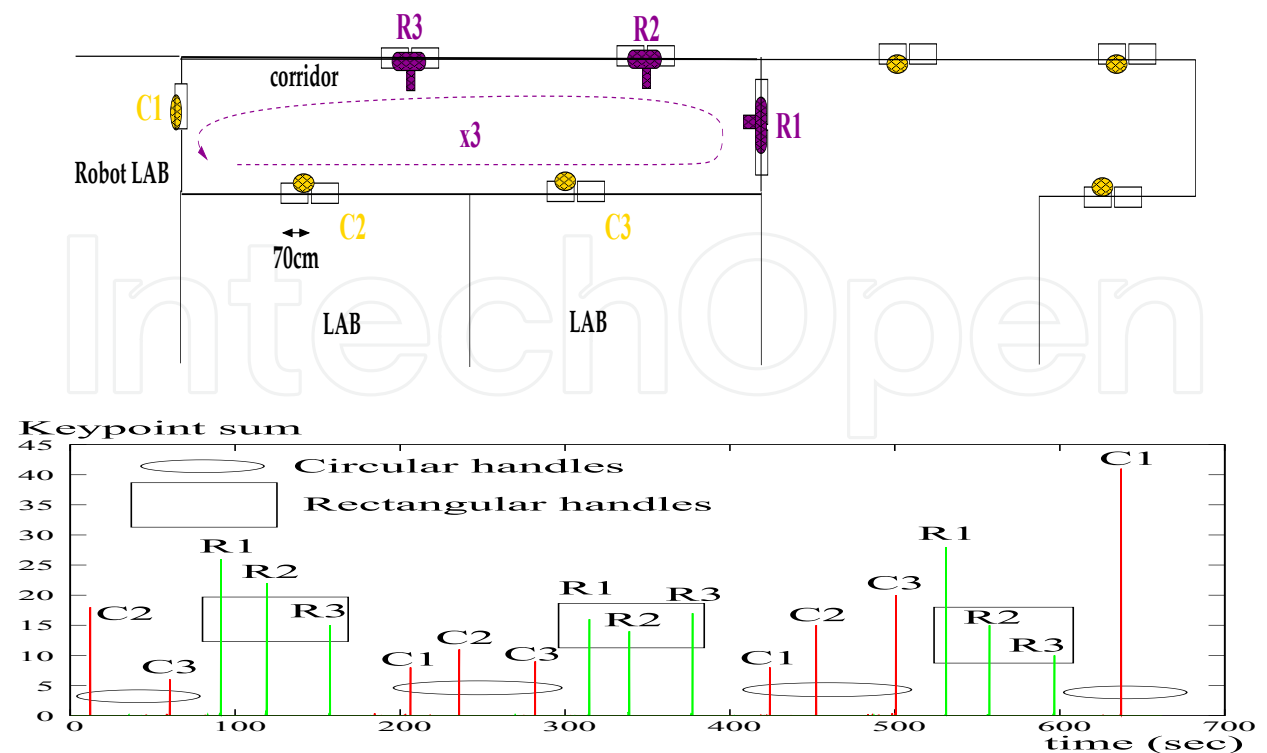


Fig. 13. Mixed environment

In each round, three circular handles and three rectangular ones were positively identified and no false positive was given by the system. The total amount of handles correctly recognised was 18.

Summing up, during the experiments performed within the real robot/environment system, 74 handles were identified, achieving a success of 98.66% and not a false positive occurred.

8. Conclusions and further work

The chapter summarises several approaches aimed at identifying doors for robot navigation purposes. Environmental specific methods showed a good performance. However, the segmentation based approach is not easily applicable to other environments since new segmenters need to be developed. Feature extraction methods were easy to apply because they only required building a reference database containing the objects to be identified. However, the classification performance and the computational payload needed to process a single image make these methods unsuitable to be used during robot navigation. A new two step algorithm was presented based on feature extraction that aimed at improving the extracted features to reduce the superfluous keypoints to be compared at the same time that it increased its efficiency by improving accuracy and reducing the computational time.

ROI extraction improves handle identification procedure and depending on the ROI size, the computational time to classify an image can be considerably reduced. The system showed a very low tendency to give false positives while providing a robust identification. The authors' opinion is that the presented approach is appropriate for situations where the background is so varied that its characteristics are either irrelevant for object identification or just add noise to the recognition process.

The area of mobile robotics needs general methods for several reasons. The use of algorithms specifically designed to work in a concrete environment makes the comparison among methods difficult as well as the application of the approach in different robot/environment systems. The developed system outperforms the performances obtained without extracting the ROIs, and experiments carried out in a real robot-environment system showed the adequateness of the approach. Opposite to the segmentation based method, the two-step method can easily be generalised to other types of handles. The same blob extraction method could be applied to obtain the region of interest, adapting the reference database to contain the correct reference keypoints and, adjusting the size restrictions applied to the region detector results –which depend on the distance the images are taken from by the robot.

Currently, the aim is to extend the proposed method to other applications such as face recognition and road signal identification. Still, the keypoint matching criteria has to be analysed more deeply. More sophisticated and efficient algorithms remain to be tested and the performance of different distance measures still needs to be studied. Obviously, the proposed method is opened to any other feature extraction method and to any improvements that could be made.

Acknowledgements

This work has been supported by the Basque Country Government under Research Team Financiation, and by the ROBAUCO Ministry project.

9. References

- Astigarraga, A., Lazkano, E., Rañó, I., Sierra, B., and Zarautz, I. (2003). SORGIN: a software framework for behavior control implementation. In *CSCS14*, volume 1, pages 243–248.
- Ayala-Ramírez, V., Garcia-Capulin, C., Perez-Garcia, A., and Sanchez-Yanez, R. R. (2006). Circle detection on images using genetic algorithms. *Pattern Recognition Letters*, 27(6):652–657.
- Bay, H., Ess, A., Tuytelaars, T., and Gool, L. V. (2008). SURF: Speeded up robust features. *Computer Vision and Image Understanding*, 110(3):346–359.
- Bay, H., Tuytelaars, T., and Gool, L. V. (2006). SURF: Speeded up robust features. In *Proceedings of the 9th European Conference on Computer Vision*.
- Brooks, R. A. (1986). A robust layered control system for a mobile robot. *IEEE Journal of robotics and automation*, RA-26:14–23.
- Chen, S.-H. (1996). Elements of information theory : Tomas m. cover and joy a. thomas, (john wiley & sons, new york, ny, 1991). *Journal of Economic Dynamics and Control*, 20(5):819–824.
- Duda, R. and Hart, P. E. (1972). Use of Hough transform to detect lines and curves in pictures. *Communications of the ACM*, 15(1):11–15.
- Eberset, C., Andersson, M., and Christensen, H. I. (2000). Vision-based door-traversal for autonomous mobile robots. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 620–625.
- Gerkey, B. P., Vaughan, R. T., and Howard, A. (2003). The Player/Stage project: tools for multi-robot and distributed sensor systems. In *Proc. of the International Conference on Advanced Robotics (ICAR)*, pages 317–323.
- Gil, A., Reinoso, O., Vicente, A., Fernández, C., and Payá, L. (2005). Monte Carlo localization using SIFT features. *Lecture Notes in Computer Science*, 3522:623–630.
- Gonzalez, R. C. and Woods, R. E. (1993). *Digital image processing*. Addison Wesley.
- Horn, B. K. P. (1986). *Robot Vision*. MIT Press.
- Kimme, C., Ballard, D., and Sklansky, J. (1975). Finding circles by array accumulators. *Communications of the ACM*, 18(2):120–122.
- Kragic, D., Petersson, L., and Christensen, H. I. (2002). Visually guided manipulation tasks. *Robotics and Autonomous Systems*, 40(2-3):193–203.
- Lazkano, E., Sierra, B., Astigarraga, A., and Martínez-Otzeta, J. M. (2007). On the use of bayesian networks to develop behavior for mobile robots. *Robotics and Autonomous Systems*, 55(3):253–265.
- Ledwich, L. and Williams, S. (2004). Reduced sift features for image retrieval and indoor localisation. In *Australian Conference on Robotics and Automation*.
- Li, W., Christensen, H. I., and Orebäck, A. (2004). An architecture for indoor navigation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1783–1788.
- Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–100.
- Mallot, H. A. and Franz, M. A. (2000). Biomimetic robot navigation. *Robotics and Autonomous System*, 30:133–153.
- Mikolajczyk, K. and Schmid, C. (2005). A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630.

- Monasterio, I., Lazkano, E., Rañó, I., and Sierra, B. (2002). Learning to traverse doors using visual information. *Mathematics and Computers in Simulation*, 60:347–356.
- Muñoz-Salinas, R., Aguirre, E., and García-Silvente, M. (2005). Detection of doors using a genetic visual fuzzy system for mobile robots. Technical report, University of Granada.
- Murthy, S. K., Kasif, S., and Salzberg, S. (1994). A system for induction of oblique decision trees. *Journal of Artificial Intelligence Research*, 2:1–33. <ftp://blaze.cs.jhu.edu/pub/oc1>.
- Pfeifer, R. and Bongard, J. (2006). *How the body shapes the way we think. A new view of intelligence*. MIT Press.
- Primdahl, K., Katz, I., Feinstein, O., Mok, Y., Dahlkamp, H., Stavens, D., Montemerlo, M., and Thrun, S. (2005). Change detection from multiple camera images extended to non-stationary cameras. In *Proceedings of Field and Service Robotics*, Port Douglas, Australia.
- Roduner, C. and Rohs, M. (2006). Practical issues in physical sign recognition with mobile devices. In Strang, T., Cahill, V., and Quigley, A., editors, *Pervasive 2006 Workshop Proceedings (Workshop on Pervasive Mobile Interaction Devices, PERMID 2006)*, pages 297–304, Dublin, Ireland.
- Se, S., Lowe, D. G., and Little, J. (2002). Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *International Journal of Robotics Research*, 21(9):735–758.
- Seo, M. W., Kim, Y. J., and Lim, M. T. (2005). *LNAI*, chapter Door Traversing for a Vision Based Mobile Robot using PCA, pages 525–531. Springer-Verlag.
- Skarbek, W. and Koschan, A. (1994). Colour image segmentation – a survey. Technical report, Polish Academy of Sciences.
- Smith, S. M. and Brady, J. M. (1997). SUSAN: a new approach for low level image processing. *International Journal of Computer Vision*, 23(1):45–78.
- Stella, E., Cirelli, G., Attolico, G., and Distante, A. (1996). Sauro: An autonomous mobile vehicle for indoor environment. In *MELECON*, volume 2, pages 1145–1150.
- Tamimi, H., Halawani, A., Burkhardt, H., and Zell, A. (2006). Appearance-based localization of mobile robots using local integral invariants. In *In Proc. of the 9th International Conference on Intelligent Autonomous Systems (IAS-9)*, pages 181–188, Tokyo, Japan.
- Trullier, O., Wiener, S. I., Berthoz, A., and Meyer, J. A. (1997). Biologically-based artificial navigation systems: review and prospects. *Progress in Neurobiology*, 51:483–544.
- Ye, W. and Zhong, Z. (2007). Robust people counting in crowded environment. In *Proceedings of the 2007 IEEE International Conference on Robotics and Biomimetics*, pages 1133–1137.
- Yuen, H. K., Princen, J., Illingworth, J., and Kittler, J. (1990). Comparative study of Hough transform methods for circle finding. *Image and Vision Computing*, 8(1):71–77.

IntechOpen

IntechOpen



Mobile Robots Navigation

Edited by Alejandra Barrera

ISBN 978-953-307-076-6

Hard cover, 666 pages

Publisher InTech

Published online 01, March, 2010

Published in print edition March, 2010

Mobile robots navigation includes different interrelated activities: (i) perception, as obtaining and interpreting sensory information; (ii) exploration, as the strategy that guides the robot to select the next direction to go; (iii) mapping, involving the construction of a spatial representation by using the sensory information perceived; (iv) localization, as the strategy to estimate the robot position within the spatial map; (v) path planning, as the strategy to find a path towards a goal location being optimal or not; and (vi) path execution, where motor actions are determined and adapted to environmental changes. The book addresses those activities by integrating results from the research work of several authors all over the world. Research cases are documented in 32 chapters organized within 7 categories next described.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

E. Jauregi, E. Lazkano and B. Sierra (2010). Approaches to Door Identification for Robot Navigation, Mobile Robots Navigation, Alejandra Barrera (Ed.), ISBN: 978-953-307-076-6, InTech, Available from: <http://www.intechopen.com/books/mobile-robots-navigation/approaches-to-door-identification-for-robot-navigation>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen