# Navigation Assistance for the Visually Impaired Using RGB-D Sensor With Range Expansion

A. Aladrén, G. López-Nicolás, *Member, IEEE*, Luis Puig, and Josechu J. Guerrero, *Member, IEEE*

***Abstract*—Navigation assistance for visually impaired (NAVI) refers to systems that are able to assist or guide people with vision loss, ranging from partially sighted to totally blind, by means of sound commands. In this paper, a new system for NAVI is presented based on visual and range information. Instead of using several sensors, we choose one device, a consumer RGB-D camera, and take advantage of both range and visual information. In particular, the main contribution is the combination of depth information with image intensities, resulting in the robust expansion of the range-based floor segmentation. On one hand, depth information, which is reliable but limited to a short range, is enhanced with the long-range visual information. On the other hand, the difficult and prone-to-error image processing is eased and improved with depth information. The proposed system detects and classifies the main structural elements of the scene providing the user with obstacle-free paths in order to navigate safely across unknown scenarios. The proposed system has been tested on a wide variety of scenarios and data sets, giving successful results and showing that the system is robust and works in challenging indoor environments.**

***Index Terms*—Navigation assistance for visually impaired (NAVI), range and vision, RGB-D camera, visually impaired assistance, wearable system.**

## I. Introduction

OBTAINING structural layout of a scene and performing autonomous navigation is an easy task for anyone, but it is not a simple task for visually impaired people. According to the World Health Organization, in 2012 there were 285 million visually impaired people and 39 million were blind. In this framework, wearable systems referred to as navigation assistance for visually impaired (NAVI) can be useful for improving or complementing the human abilities in order to better interact with the environment. This paper is in the context of the project VINEA (wearable computer VIsion for human Navigation and Enhanced Assistance). The main goal of this project is the joint research of computer vision and robotic techniques in order to achieve a personal assistance system based on visual information. In particular, the goal is to design a system for personal assistance that can be worn by a person. This system will help people to navigate in unknown environments, and it will complement rather than replace the human abilities. Possible users of this system will range not only from visually impaired people but also to users with normal visual capabilities performing specific tasks, such as transport of merchandise that complicates the visibility or accessing to dark areas or environments with changing light conditions, such as fast light flickering or dazzling lights.

Different approaches for NAVI have been developed [1]. In general, they do not use visual information, and they need complex hardware systems, not only to equip the user but also the building where the navigation has to be accomplished. The system developed by Öktem *et al.* [2] used wireless communication technology. Another system is [3], where ultrasonic and GPS sensors are used.

Vision sensors play a key role in perception systems because of their low cost and versatility. An example of a system for indoor human localization based on global features that does not need 3-D reconstruction is presented in [4]. However, a disadvantage of monocular systems is that the global scale is not observable from a single image. A way to overcome this problem is by using stereo vision such as in [5], where a system for NAVI is developed by implementing a stereo vision system to detect the obstacles of the scene. The scale can be also obtained by measuring the vertical oscillation in the image during walking to estimate the step frequency, which was empirically related with the speed of the camera [6]. More recently, range information, which directly provides depth information, has been integrated in these systems. This information has been mainly used to find and identify objects in the scene [7]–[9]. One step ahead is to integrate range systems in the navigation task. Some examples are [10], where the task of NAVI is addressed using a Kinect camera, and [11], where range information is used to distinguish solid obstacles from wild terrain.

Fast corner detector and depth information for path planning tasks is used in [12], and a system that follows a colored navigation line that is set on the floor and uses radio-frequency identification technology to create map information is presented in [13]. A previous floor plan map of a building is used in [14] to define a semantic plan for a wearable navigation system by means of augmented reality.

A main initial stage for any autonomous or semiautonomous navigational system is the recognition of the structure of the environment. Most mobile robots rely on range data for obstacle detection [15]. Popular sensors based on range data are ultrasonic sensors, radar, stereo vision, and laser sensors. These

Manuscript received October 14, 2013; revised February 28, 2014; accepted April 16, 2014. This work was supported in part by the project VINEA DPI2012-31781 and in part by the Fondo Europeo de Desarrollo Regional.

A. Aladrén, G. López-Nicolás, and J. J. Guerrero are with the Instituto Universitario de Investigación en Ingeniería de Aragón, Universidad de Zaragoza, 50009 Zaragoza, Spain (e-mail: gonlopez@unizar.es).

L. Puig is with the General Robotics, Automation, Sensing and Perception Laboratory, University of Pennsylvania, Philadelphia, 19104-6228 PA USA.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSYST.2014.2320639

Fig. 1. Experimental setup with the range camera hanging from the user's neck. This device is light and easy to wear by the user.

devices measure the distance from the sensor to surrounding obstacles with different levels of accuracy [16]. Ultrasonic sensors are cheap and simple, but they provide with poor angular resolution and poor information of the scene. Radar systems perform better than ultrasonic sensors, but they are more complex and expensive and may suffer from interference problems with other signals inside buildings. Although laser sensors provide good and accurate information, they are expensive, heavy, and involve high-power requirements; thus, they are not the best option for human wearable applications. Recently, RGB-D sensors have been popularized due to the great amount of information they provide and to their low cost and good miniaturization perspectives. The RGB-D device provides range information from active sensing by means of infrared sensor and intensity images from a passive sensor such as a standard camera. This is the only sensor used in this paper, which benefits from both the range and visual information to obtain a robust and efficient system. This kind of sensor brings not only new opportunities but also new challenges to overcome. A picture of our experimental setup is shown in Fig. 1, where it can be seen that the range camera hangs from the user's neck, whereas the laptop is carried in a backpack.

In the field of computer vision, it is clear that image processing has made amazing advances in the last decades. Regarding the problem of floor-segmentation and road-segmentation tasks, several related works are the following. Ulrich and Nourbakhsh [17] presented a system that solves the floor-segmentation problem using hue and light information of the images. Li and Birchfield [18] proposed techniques related with lines extraction and thresholding. Adams and Bischof [19] showed a method for segmentation of intensity images based on seeded region growing techniques. Image segmentation has been also used for advanced driver assistance systems. Álvarez *et al.* [20] used a histogram-based road classifier. In [21], a method to find the drivable surface with appearance models is presented. In [22], it is shown that the fusion of information, in particular color and geometry information, improves the segmentation of the scene.

A man-made environment, which is essentially composed of three main directions orthogonal to each other, is usually assumed. This is usually denoted by the Manhattan world assumption. Taking this into account, a cubic room model is used to recognize surfaces in cluttered scenes in [23]. Straight lines are relevant features from structured environments, which can be used to impose geometrical constraints in order to find corner or relevant features such as parallelism or orthogonality between elements to generate plausible hypothesis of the scene's structure [24]. Scene understanding has been also considered by combining geometric and photometric cues [25] or from a single omnidirectional image [26], [27].

In this paper, we present a system that combines range information with color information to address the task of NAVI. The main contribution is the robust expansion of the range-based floor segmentation by using the RGB image. This system guides a visually impaired person through an unknown indoor scenario. Range information is used to detect and classify the main structural elements of the scene. Due to the limitations of the range sensor, the color information is jointly used with the range information to extend the floor segmentation to the entire scene. In particular, we use range information for closer distances (up to 3 m), and color information is used for larger distances (from 3 m to the rest of the scene). This is a key issue not only to detect near obstacles but also to allow high-level planning of the navigational task due to the longer range segmentation our method provides. An example of high-level planning is when there is an intersection of paths in the scenario, and the user would be able to decide the way he wanted in advance.

We have also developed the user interface that sends navigation commands via sound map information and voice commands. In particular, the sound map is created using stereo beeps, which frequency depends on the distance from the user to an obstacle, and the voice commands provide high-level guidance along the free-obstacle paths. The proposed system has been tested with a user wearing the prototype on a wide variety of scenarios and data sets. The experimental results show that the system is robust and works correctly in challenging indoor environments. The proposal works on a wide variety of indoor environments, which can be characterized from small walking areas, such as a 15-$m^2$ room or huge walking areas such as the corridors or the hall rooms of a public building. The surrounding obstacles in the scene can also vary from no obstacles to a number of obstacles that prevent the user from walking without modifying his trajectory.

This paper is organized as follows. In Section II, the new algorithm for the scene segmentation providing obstacle-free paths is presented. In the first stage, range data are processed to obtain an initial layout of the scene, and then, the method extends the range information with the color information, resulting in robust and long-range layout segmentation. Section III presents the user interface that guides the user and informs about scene's obstacles. Section IV reports the experimental results obtained with a wide variety of real scenarios. Finally, Section V draws the conclusions.

## II. Obstacle-Free Path Segmentation

Here, we present an algorithm that extracts the floor from the scene in order to detect obstacles. The first stage of the algorithm only uses range information to detect planes of the scene. Then, this extraction is improved and extended to the whole scene using color information.

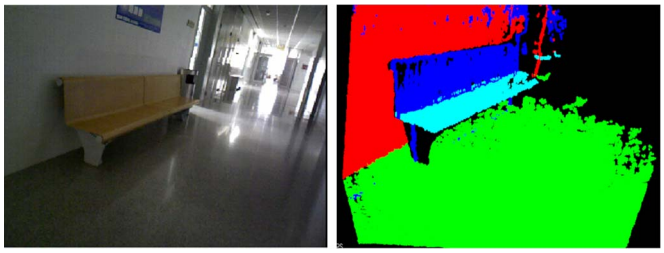

Fig. 2. Example of plane segmentation in a corridor scene using range information. The left image is the acquired image, and the right image is the 3-D plane segmentation result.

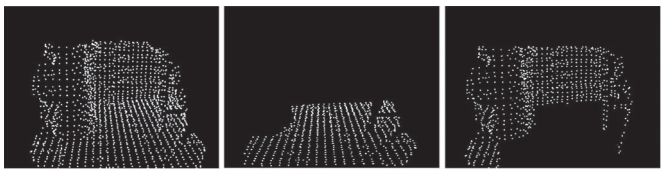

Fig. 3. Left image shows the filtered point cloud. Center and right images show the first detected plane and the point cloud after first plane extraction, respectively. This last point cloud will be processed again in order to extract the next plane of the scene, and so on.

### A. Floor Segmentation Using Range Data

The first step of any navigation system is to distinguish the floor from the obstacles and walls of the scene. Here, we present the processing of the range information captured by the RGB-D device. Since we are only interested in the main structure of the environment, we reduce the amount of data to be processed by downsampling the point cloud. In particular, we use a voxel-based filter creating cubes that are placed in the existing surfaces of the point cloud. All points contained in the same cube become a single point, i.e., the centroid of the corresponding cube. To give an idea of the advantage in terms of efficiency, the point cloud before filtering can contain around 300 000 points and around 7000 points after filtering without losing representative information. The next step is to identify the most representative planes of the scene from the point cloud. An example of the output of this part is presented in Fig. 2, where the 3-D plane segmentation of a corridor scene is shown.

The algorithm used for plane detection is RANdom SAmple Consensus (RANSAC) [28], [29] by using the plane model. The RANSAC algorithm provides a robust estimation of the dominant plane parameters, performing a random search in the space of solutions. The number of computed solutions $m$ is selected to avoid possible outliers in the random selection of the three points, which define each plane

$$m = \frac{\log(1-P)}{\log\left(1-(1-\varepsilon)^p\right)} \tag{1}$$

where $P$ is the probability of not failing the computation because of outliers, $p$ is the dimension of the model, and $\varepsilon$ is the overall percentage of outliers.

In order to perform the detection of the most representative planes of the scene, we use Algorithm 1. A graphical result of the procedure is shown in Fig. 3, with the input filtered point cloud, the first detected plane, and the point cloud after removing the first plane's points. This process is repeated until the number of points contained in a candidate plane is less than a certain value.

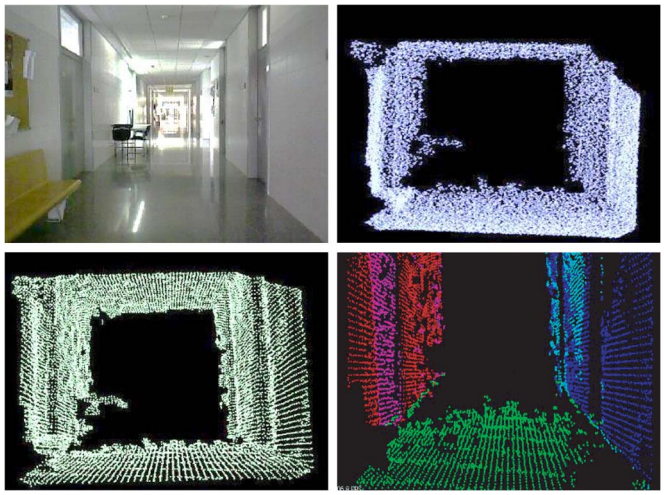

Fig. 4. Summary of the segmentation algorithm based on range information. (Top left) Original RGB image. (Top right) Acquired point cloud. (Bottom left) Filtered point cloud. (Bottom right) Resultant range-based segmentation of points where the segmented floor is labeled as obstacle-free.

**Algorithm 1** Range-based plane extraction

$Min\ number\ of\ points = constant$;
$i = 1$;
$\Pi_i = Extract\ range\ plane\ (filtered\ point\_cloud)$
**while** $Points\ of\ plane \geq Min\ number\ of\ points$ **do**
  $i = i + 1$;
  $\Pi_i = Extract\ range\ plane\ (filtered\ point\_cloud)$
**end while**

Once all the planes in the scene have been detected, we need to identify them. We consider that the scene follows a Manhattan world model, which assumes that the environment has three main directions that are orthogonal between them. The identification of each plane is carried out by analyzing the normal vector of each plane. Eventually, we obtain the classification of the scene planes as floor and obstacles. These are labeled as range floor and range obstacles, respectively, and will be used in the following section for the floor hypothesis expansion. A summary of the phase segmentation algorithm based on range information is shown in Fig. 4. The presented method works properly at indoor scenes, and it is robust to lighting changes. However, it has some limitations: It is susceptible to sunlight, and the maximum distance it can accurately measure is up to 3.5 m. In particular situations, range measurements could be also obtained up to 7 m but with low accuracy. These limitations are overcome by expanding the obtained floor segmentation with color information as explained next.

### B. Path Expansion Using Color Information

Up to now, we have only used the depth information of the RGB-D device. As commented above, range information has limitations, and it would be enough for reactive obstacle avoidance but not enough for applications, such as path planning, in which information of the whole scene is required.

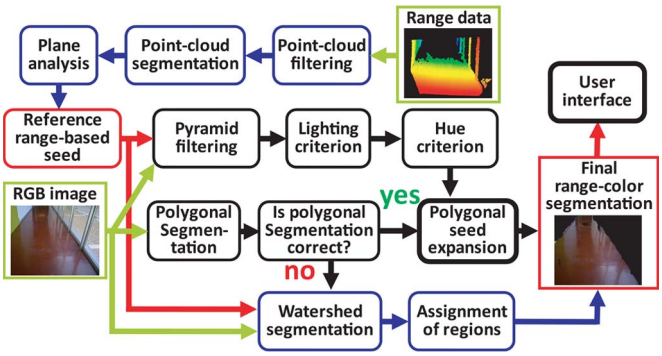


Fig. 5. Overall diagram of the proposed method to expand the range-based results to the entire scene by means of color information.

In order to extend the segmentation given by the depth sensor, we use the monocular camera information. Obtaining the whole surface of the ground is essential to compute the obstacle-free path. Here, we present two methods to segment the floor of the entire image: the polygonal floor segmentation and the watershed floor segmentation. The appropriate method is automatically selected by the algorithm depending on the type of the scene we are dealing with. The RGB and Hue, Saturation, and Intensity color spaces are used to fulfill the task, as well as image geometrical features. We also deal with shadows and reflections, which are common phenomena in these environments. A diagram of the different stages of the floor expansion method is depicted in Fig. 5. The corresponding explanations for each step are provided next.

*1) Polygonal Floor Segmentation:* In this method, we initialize a seeded region growing algorithm, where the seed belongs to the floor's plane given by the range segmentation. As mentioned before, our algorithm uses hue, lighting, and geometry image features. Based on this information, we define similarity criteria, if a pixel satisfies these criteria, it is labeled as floor-seed.

*a) Pyramid filtering:* The first step of this algorithm is to homogenize as much as possible the image data of the scene. The method used for this purpose is the shift mean algorithm over a pyramid of images [30]. The main elements of this method are image pyramids and the mean-shift filter.

The image pyramid consists of a collection of images, all arising from a single original image, which are successively downsampled until some desired stopping point is reached. There are two common kinds of image pyramids: Gaussian pyramid and Laplacian pyramid. The first one is used to downsample images, whereas the second one is used to upsample images from an image with less resolution.

In the case of Gaussian pyramid, the image at layer $i$ is convolved with a Gaussian kernel, and then, every even-numbered row and column are removed. The resulting image at layer $i+1$ will be exactly one-quarter the area of its predecessor. The Laplacian pyramid is the opposite case. In this case, the image at layer $i+1$ in the Laplacian pyramid is upsized to twice the original image in each dimension with the new even rows and columns filled with zeros. Then, a convolution with the same Gaussian kernel (multiplied by 4) to approximate the values of the missing pixels is performed.

Fig. 6. Example of use of the shift mean algorithm over pyramid images. Left: Original image. Right: Result obtained using shift mean algorithm over pyramid images.

The mean-shift filter works as follows. Given a set of multi-dimensional data points, whose dimensions are $(x, y, blue, green, red)$, mean shift can find the highest density clumps of data in this space by scanning a window over the space. Notice, however, that the spatial variables $(x, y)$ have very different ranges from the color magnitude ranges $(blue, green, red)$. Therefore, mean shift needs to allow for different window radii in different dimensions. At every pixel $(X, Y)$ of the input image the function executes mean-shift iterations, that is, the pixel $(x, y)$ neighborhood in the joint space-color hyperspace is considered

$$\begin{aligned} (x,y) : X - sp &\leq x \leq X + sp, \\ Y - sp \leq y &\leq Y + sp, \\ \|(R,G,B) - (r,g,b)\| &\leq sr \end{aligned} \tag{2}$$

where $(R, G, B)$ and $(r, g, b)$ are the vectors of color components at $(X, Y)$ and $(x, y)$, and $sp$ and $sr$ are the spatial and color window radius.

All pixels that have been traversed by the spatial and color filter windows and converge at a same certain value in the data, will become connected and will build a region in the image. An example of the result obtained is shown in Fig. 6 where the left image shows the original scene, and the right image shows the result obtained using this algorithm. Notice that the spotted floor has been smoothed, obtaining a more homogeneous floor. See for instance that the socket placed on the right wall has been also removed. On the other hand, the boundaries between the floor and all obstacles of the scene have been respected. Hence, this algorithm allows us to remove unnecessary details, whereas the relevant ones are not affected.

*b) Seed lighting criteria:* The next step is to compare the lighting of the homogenized image with the floor-seed. Being $H_1$ and $H_2$, the histograms of the homogenized image and the floor-seed, respectively, we compare the histograms of the lighting channel of both images

$$d(H_1, H_2) = \frac{\sum_I \left(H_1(I) - \overline{H_1}\right)\left(H_2(I) - \overline{H_2}\right)}{\sqrt{\sum_I \left(H_1(I) - \overline{H_1}\right)^2 \sum_I \left(H_2(I) - \overline{H_2}\right)^2}} \tag{3}$$

where

$$\overline{H_k} = \frac{1}{N} \sum_J H_k(J).$$

Pixels satisfying the lighting similarity criterion will pass to the next step.

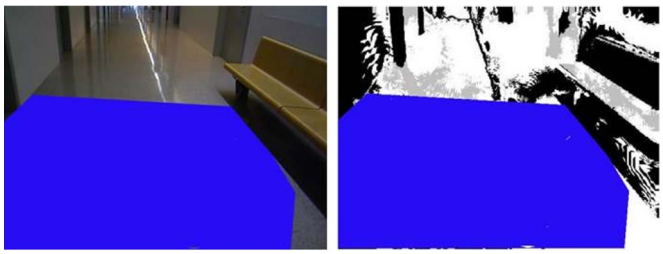

Fig. 7. Example of the hue comparison algorithm. Left image shows the original scene with the range floor-seed in blue. Right image shows the final result where white regions are those that have the highest probability of being part of the floor.

Fig. 8. Example of the polygonal-based segmentation algorithm. Left image shows the extracted lines forming the polygonal regions. Right image shows the corresponding polygonal-based floor segmentation.

*c) Seed hue criteria:* The next comparison is related to the hue channel. As we mentioned before, the floor is not homogeneous; thus, the floor-seed will have different hue values. Then, we compare each region of the image that satisfies the previous criterion with each hue value of the floor-seed.

We carry this task out with a back-projection algorithm. This algorithm is a way of recording how well the pixels of a given image fit the distribution of pixels in a histogram model. Fig. 7 shows an example of how this comparison algorithm works. The left image shows the original scene with the range floor-seed (blue region), and the right image shows the obtained result. The first step is to segment the range floor-seed in its different hue levels and to obtain their respective histograms. After that, we go over the region of the image, which does not belong to the range floor-seed. For each pixel, we obtain its value and frequencies in the floor-seed histograms. The right image of Fig. 7 shows the frequencies obtained for each pixel. White regions are those regions that have the highest probability to be part of the floor. Pixels that satisfy this criterion are considered as new additional image floor-seeds.

*d) Polygonal segmentation:* Once we have all the floor-seeds of the image, it is time to mark out the region growing of each floor-seed. The borderline between the floor and the rest of obstacles is usually defined by lines. Here, we propose an algorithm that segments the scene in different regions in order to detect these borderlines between accessible and nonaccessible zones of the scene. Because of the reason commented above, we have decided to create these regions with polygons. These polygons are generated by detecting the most representative lines of the original image. In particular, we apply the Canny edge detector [31], followed by the probabilistic Hough line transform [32] to extract the line segments from the image. Then, these lines are extended to the image borders; thus, we obtain a scene segmented by irregular polygons. Fig. 8 shows an example of the result obtained with the polygonal-segmentation algorithm.

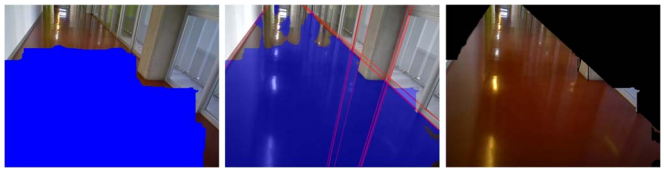

Fig. 9. Example of polygonal-based expansion of floor-seeds. Left image: Initial floor-seeds provided by range segmentation algorithm. Center image: Final floor-seeds provided by color segmentation algorithm and polygonal segmentation. Right image: Final result of floor segmentation.

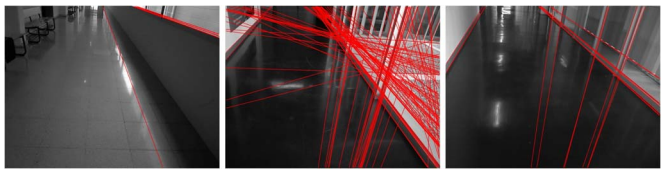

Fig. 10. Example of homogeneous and no homogeneous line distribution. Left and center images show cases of bad distribution. Right image is an example of satisfactory polygonal segmentation due to lines distribution.

Once we have the image segmented into different regions, we use the reference floor (Range floor's plane) to determine the regions that belong to the floor. All regions that have at least one or more floor-seeds of the reference floor and do not belong to a range obstacle are labeled as floor. An example showing the result of our procedure based on watershed segmentation is shown in Fig. 9.

*2) Watershed Segmentation:* The polygonal segmentation has satisfactory results if the lines extracted with probabilistic Hough line transform are representative lines of the scene. However, given the wide variety of possible scenarios, this is not always the case. We propose an alternative algorithm of floor segmentation, named watershed segmentation, to be used in these situations when the number of extracted lines is either too low or too high, or the extracted lines have a heterogeneous distribution. The system automatically decides if the polygonal segmentation is correct or if the watershed segmentation is needed. The specific criterion of the algorithm selection is based on the number of extracted lines and their spatial distribution in the scene. Hence, if the number of extracted lines is over a certain threshold and they are homogeneously distributed, the system will choose the polygonal segmentation. In the rest of cases watershed segmentation will be the best option. An example of homogeneous and nonhomogeneous distributions of lines is shown in Fig. 10. The first two images are examples of useless polygonal segmentation, due to the low number of lines (left) or an excessive number of them (center). The image on the right shows an example of adequate segmentation, where lines are distributed over the whole scene.

The algorithm we propose is based on watershed segmentation [33]. The input to this algorithm is a binary image given by the Canny edge detector. This binary image contains the "marks" and boundaries between regions, which will be used later on. The watershed algorithm converts lines in an image into "mountains" and uniform regions into "valleys" that can be used to segment a region of an image. This algorithm first takes the gradient of intensity image; this has the effect of forming valleys or basins, where there is no texture and of forming mountains or ranges where there are dominant lines in

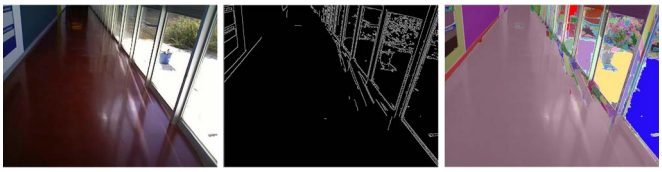

Fig. 11. Example of watershed segmentation using a binary image of marks created with Canny edge detector. Left: Original image. Center: Canny edge detector output. Right: Segmentation result.

the image. It then successively floods basins until these regions meet. Regions that merge across the marks are segmented as belonging together as the image "fills up." This way, the basins connected to the marker point become "owned" by that marker. After that, the image is segmented into the corresponding marked regions, and the result is similar to a superpixel image.

Once we have the image segmented into different regions, we use the reference floor (Range floor's plane) to determine the regions that belong to the floor. All regions that have at least one or more floor-seeds of the reference floor are labeled as floor. An example showing the result of our procedure based on watershed segmentation is shown in Fig. 11.

## III. User Interface

Different user interfaces have been investigated for this type of applications. For example, in [34], a user interface for indoor scenarios was designed using wireless technology and a compass in order to locate and guide the user along the scenario. Audio systems have been combined with white canes, as the one developed by the Central University of Michigan [35]. This device uses ultrasonic sensor technology in order to detect possible obstacles. When an obstacle is detected, the system provides information to the user in such a way that it can avoid the obstacle. A different system is presented in [36], where a vibration system in addition to audio interface is used. The vibration system takes responsibility of giving information of the scenario to the user through vibrations.

In this paper, we propose to create a simple interface that gives information to the user according to the results provided by the presented algorithms. Thus, we substitute the vibration system with a sound map in order to make it more simple and wearable. Therefore, our user interface provides audio instructions and sound map information. Audio instructions will be used only for high-level commands, available free-path information, or in dangerous situations, where the user could collide with an obstacle. In this case, the system will warn about the situation and will give instructions in order to avoid the obstacle in a satisfactory way. In the rest of cases, the sound map will send stereo beeps, whose frequency depends on the distance from the obstacle to the person. We have defined the safety area from the user to any obstacle as 2 m. A known drawback of audio systems is that they may block other natural sounds. However, our system does not provide constant audio instructions or beeps; thus, the possible blocking of natural sounds will only appear sporadically. The user may also regulate the volume of the system so that he could hear natural sounds and audio instructions at the same time.

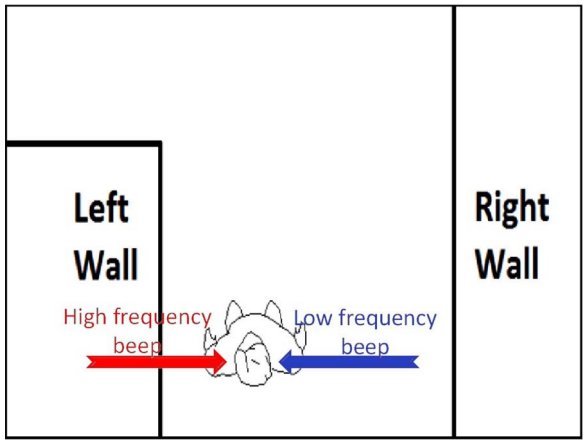


Fig. 12. Top view diagram of an example with the user in a corridor. The sound map information informs the user about the nearer obstacles.

TABLE I
Audio Instructions That the System Provides to the User

| Condition | Audio instruction |
|---|---|
| Obstacle placed in front of the user with no avoidance option. | *Attention, obstacle in front of you. You should turn left.* |
| Wall placed on the left and obstacle placed in front of the user with no avoidance option | *Attention, obstacle in front of you. You should turn right* |
| Walls placed on both sides and obstacle placed in front of the user with no avoidance option. | *Attention, there's no way* |
| Obstacle in front of the user with avoidance option on the left. | *Attention, obstacle in front of you. Step to the left.* |
| Obstacle in front of the user with avoidance option on the right. | *Attention, obstacle in front of you. Step to the right.* |
| Obstacle in front of the user but no avoidance needed. | *Attention, obstacle in front of you. Go straight.* |
| Free path placed on both sides and in front of the user. | *Attention, available way is left, ahead and right.* |
| Free path placed on the left and in front of the user. | *Attention, available way is left and ahead.* |
| Free path placed on the right and in front of the user. | *Attention, available way is ahead and right.* |
| Free path on both sides of the user. | *Attention, available way is left and right.* |
| Free path in front of the user. | *Attention, available way is ahead.* |

TABLE II
Frequency and Ear, Which Will Receive the Beeps Depending on the Type of Obstacle and Its Distance to the User

| Obstacle location | Distance to the user | Frequency | Ear |
|---|---|---|---|
| Left | $d < 30cm$ | 1500 Hz | Left ear |
| | $30cm < d < 1m$ | 700 Hz | |
| | $1m < d < 1.5m$ | 650 Hz | |
| | $1.5m < d < 2m$ | 600 Hz | |
| | $2m < d < 2.5m$ | 550 Hz | |
| | $2.5m < d < 3m$ | 500 Hz | |
| | $3m < d < 3.5m$ | 450 Hz | |
| Right | $d < 30cm$ | 1500 Hz | Right ear |
| | $30cm < d < 1m$ | 700 Hz | |
| | $1m < d < 1.5m$ | 650 Hz | |
| | $1.5m < d < 2m$ | 600 Hz | |
| | $2m < d < 2.5m$ | 550 Hz | |
| | $2.5m < d < 3m$ | 500 Hz | |
| | $3m < d < 3.5m$ | 450 Hz | |
| Front | $d < 30cm$ | 1500 Hz | Both ears |
| | $30cm < d < 1m$ | 700 Hz | |
| | $1m < d < 1.5m$ | 650 Hz | |
| | $1.5m < d < 2m$ | 600 Hz | |
| | $2m < d < 2.5m$ | 550 Hz | |
| | $2.5m < d < 3m$ | 500 Hz | |
| | $3m < d < 3.5m$ | 450 Hz | |

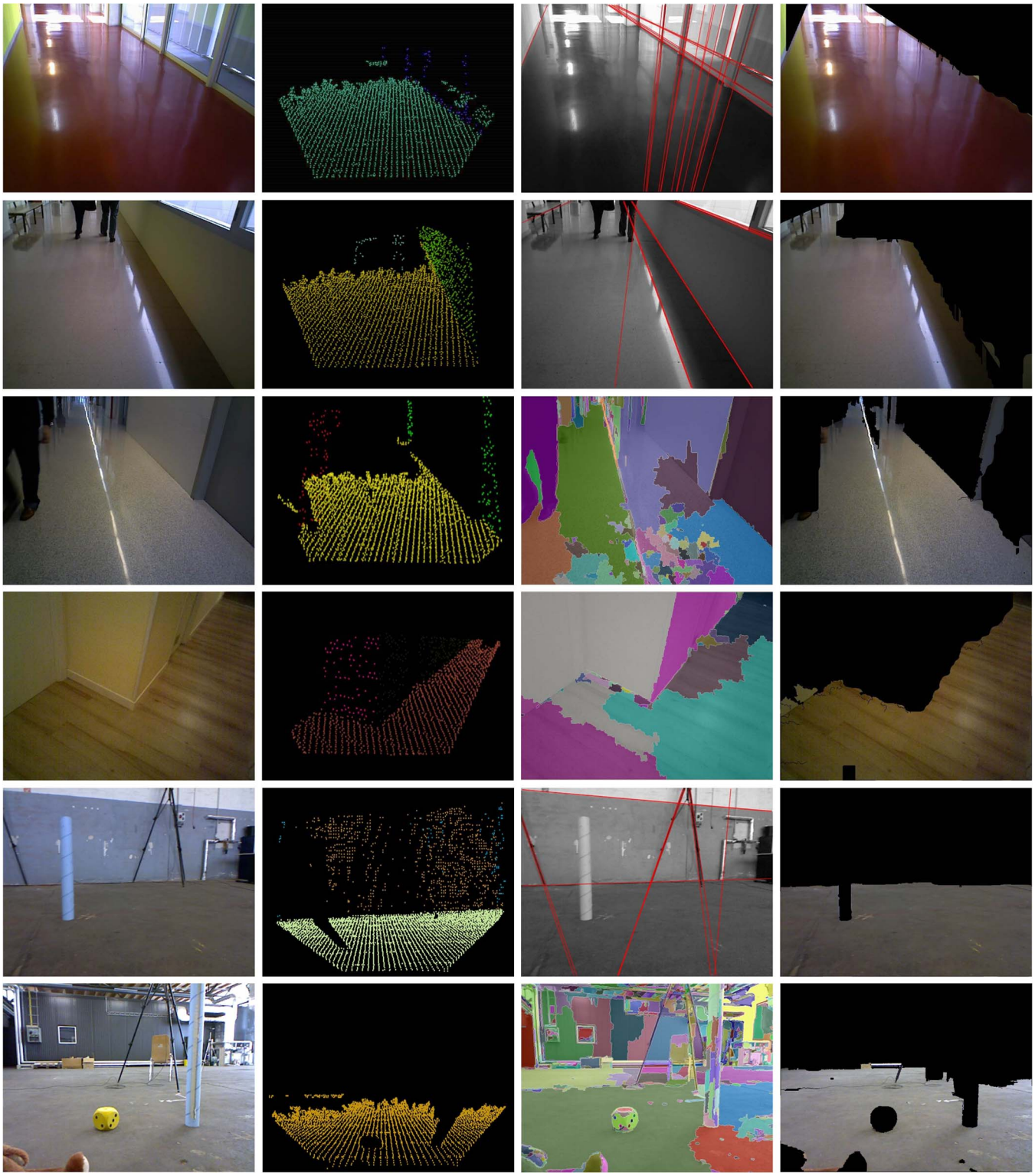

Fig. 13. Results of 3-D reconstruction and floor expansion. Each row shows a different example. First column corresponds to the original images of the scenes, second column shows the range-based plane segmentation results, third column shows the segmentation algorithm automatically selected, and fourth column shows the final floor image segmentation. Note the challenging floor reflectivity and hard lighting conditions.

In Fig. 12, we show an example of the types of sounds produced by our system with respect to the distance from the user to the obstacle. If the left wall is closer to the user than the right one, the user will hear a high-frequency beep in his left ear and a low frequency beep in the right ear. If the wall is placed in front of the person, the beep will be heard in both ears. These beeps allow the user to understand the environment. With this user interface, the user will be able to navigate through an unknown scenario, as well as being able to avoid obstacles with no risk of collision. Tables I and II show the different kind of audio instructions and beeps that the system provides to the user depending on the type of obstacle and its position and distance to the user.

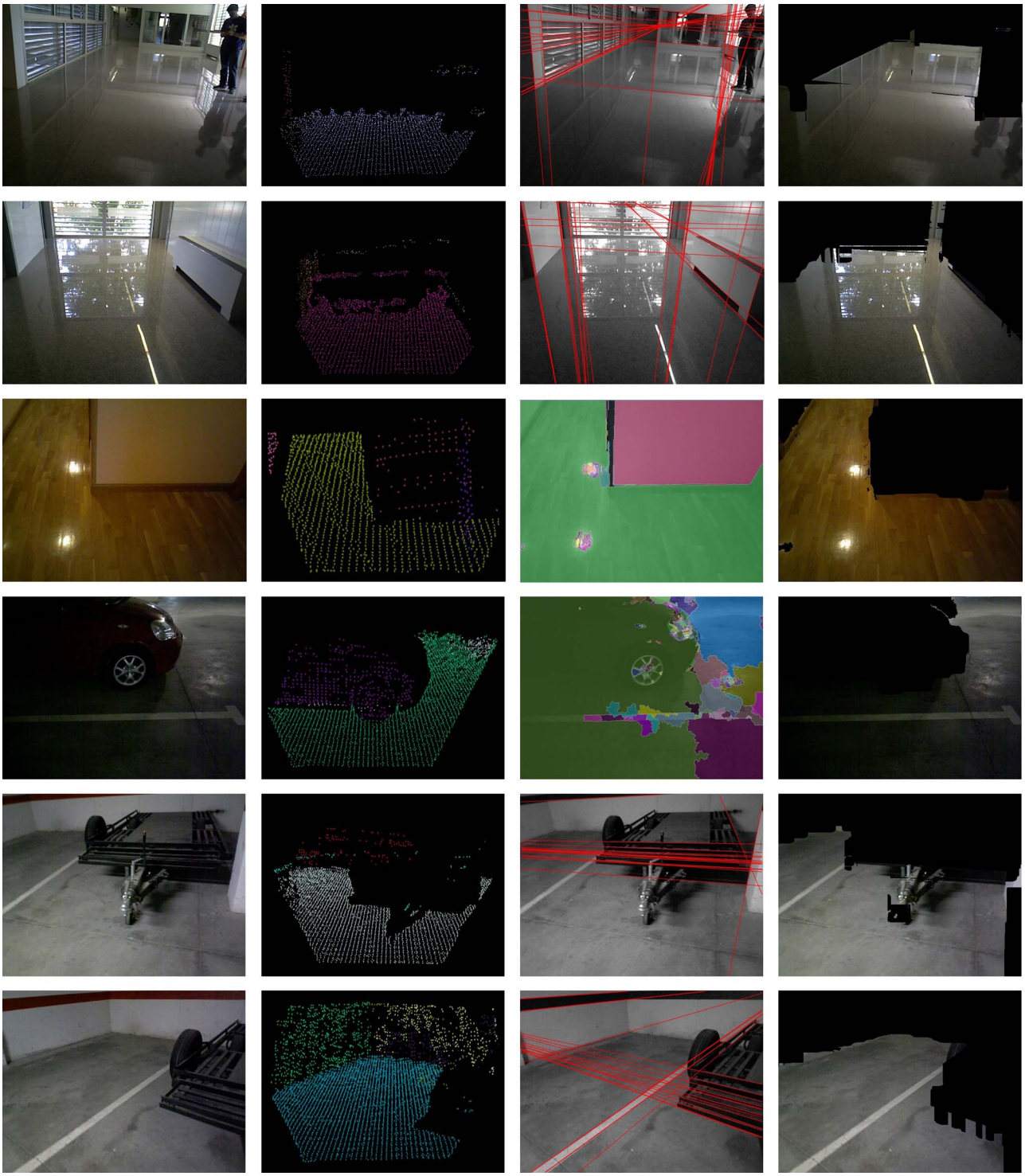

Fig. 14. More results of 3-D reconstruction and floor expansion in scenarios with more cluttered areas and scenes with extreme brightness. First column: original image. Second column shows the range plane segmentation. Third column: segmentation algorithm used, and fourth column shows the floor image segmentation. Note that despite of extreme reflections or presence of bizarre obstacles, the system provides good results that enables the appropriate navigation through the unknown scenario.

## IV. Experiments

The performance of the proposed algorithm has been evaluated in different types of real scenarios exhibiting a wide variety of different visual characteristics and lighting conditions. We have tested the algorithm in public and private buildings. The public ones are placed in University of Zaragoza (Spain) and they are as follows: Ada Byron building, Torres Quevedo building, and I + D building, where the Institute of Engineering Investigation of Aragón (I3A) is located. The private buildings are examples of houses and a garage. Since the number of data sets to test approaches for NAVI is almost nonexistent, we have released our data set,[1] which collects data used in our

[1]http://webdiis.unizar.es/%7Eglopez/dataset.html

experiment to be available to the research community. We have also evaluated our system using the data set of the Technische Universität München.[2]

As mentioned before, the presented system is worn by the user, and all the hardware required is an RGB-D device (Asus Xtion Pro live camera), a laptop, and headphones. As observed in Fig. 1, the experimental setup consists of the camera that hangs from the user's neck, and the laptop carried in a backpack. We have chosen this position of the camera because it has a high social acceptability, and its body motion is small [37]. The RGB-D device will be slightly tilted toward the ground in order to detect the closest obstacles. The parameters used in the floor segmentation using range data and according to (1) are such that the system computes 100 iterations $(m)$, which gives a probability of not failing the computation $(P)$ of 99.87% if we consider a ratio of outliers $(\varepsilon)$ of 60%.

The RGB-D data captured by the Asus Xtion Pro live camera are processed by an algorithm implemented in the C++ programming language using Robot Operating System, OpenCV library, and Point-Cloud Library (PCL) on a 2.53-GHz Core 5 processor (HP EliteBook 8440p laptop). Range segmentation algorithm runs at approximately 2 frames/s. The algorithm (range data processing, RGB image processing, and user interface generation) runs at approximately 0.3 frames/s. The implementation of the algorithm is not yet optimized; thus, this frame rate could be improved to work at higher frame rates. Additionally, the new versions of the PCLs, which are the libraries used for range information processing, take advantage of multiple processing cores; thus, they could be used to program parallel data processing in order to improve performance.

In Fig. 13, we present the results of our algorithm on some typical corridor images available in our data set and on some scenes of the data set of the University of Münich. The first column images from the left column display the original image of several different examples, and the second column images show the point-cloud segmentation. The third column images display the polygonal segmentation or the watershed segmentation depending on the automatic choice algorithm, and the last column images show the final floor expansion result. In Fig. 14, we show some additional examples where there is either extreme brightness or reflections that make more difficult the analysis of the scene. Although the difficult conditions, we obtain good results.

In order to quantitatively evaluate our proposal, we have manually labeled a representative sample of 150 images to have a ground truth. Table III shows the percentages of the floor's segmented area of pixels obtained just with range data and the result obtained with the whole system combining range and color information. We have calculated the precision, recall, and F1 statistic, according to the floor's segmented area. The recall confidence interval is also computed in the last column at the 95% confidence level. According to the results obtained for this table, we can analyze the scenarios into three groups: scenarios that have no solar light incidence, scenarios that have medium-low solar light incidence, and scenarios with high solar light incidence. The precision obtained with range data is 100% in all scenarios. These perfect precisions are caused by of short-range hardware limitations and because the range sensor is unable to obtain range data of regions that are closed to an object's boundary, producing conservative results. On the other hand, recall has low values due to these limitations.

TABLE III
QUANTITATIVE RESULTS OF THE SEGMENTED AREAS COMPARED WITH THE GROUND TRUTH IN FIVE DIFFERENT SCENARIOS. THE SEGMENTED AREAS ARE MEASURED IN PIXELS

| **Percentages of floor-segmentation with range data** | | | | |
|---|---|---|---|---|
| **Scenario** | **Precision** | **Recall** | **F1** | **Recall interval** |
| I3A building | 100% | 78,62% | 87,87% | 78,62 ± 4,79 % |
| Ada Byron bldg. | 100% | 84,23% | 91,43% | 84,23 ± 1,08 % |
| Torres Quevedo | 100% | 78,95% | 88,10% | 78,95 ± 3,51 % |
| Garage | 100% | 87,63% | 93,38% | 87,63 ± 1,68 % |
| München dataset | 100% | 54,54% | 69,01% | 54,54 ± 6,82 % |
| **Percentages of floor-segmentation with range and color data** | | | | |
| **Scenario** | **Precision** | **Recall** | **F1** | **Recall interval** |
| I3A building | 98,94% | 96,74% | 97,81% | 97,00 ± 1,20 % |
| Ada Byron bldg. | 98,97% | 95,22% | 97,04% | 95,00 ± 1,30 % |
| Torres Quevedo | 99,26% | 97,38% | 98,30% | 97,00 ± 1,00 % |
| Garage | 99,62% | 93,62% | 96,49% | 94,00 ± 1,82 % |
| München dataset | 99,09% | 96,23% | 97,62% | 96,00 ± 1,60 % |

TABLE IV
CONTRIBUTIONS TO THE FINAL RESULT OF THE RANGE-BASED PART OF THE ALGORITHM AND THE COLOR-BASED SEGMENTATION. THE SEGMENTED AREAS ARE MEASURED IN METRIC UNITS

| **Scenario** | **Range segmentation** | **Color segmentation** |
|---|---|---|
| I3A building | 26,53% | 73,47% |
| Ada Byron building | 43,34% | 56,66% |
| Torres Quevedo building | 54,92% | 45,08% |
| Garage | 74,22% | 25,78% |
| München dataset | 52,62% | 47,38% |

In clear evidence, scenarios where there is no solar light are ideal cases for range data, where a high percentage of the floor's pixels, approximately 85%, are segmented. Color data segment those regions that are farther away and represented by a less number of pixels than those regions that are closer to the camera. In the rest of cases, which are more common scenarios day-to-day, the advantages of sensor fusion are shown. Range segmentation is limited due to solar light. The floor segmentation is lower than 80% of pixels and in the case of the data set of München is reduced to 55%. In those situations where range data fail in providing good recall numbers, color data has a high index of segmentation, where a great part of pixels are segmented (from 20% to 41%), providing satisfactory final results.

In order to show the advantages of the fusion and expansion of range segmentation with color segmentation, we have calculated (see Table IV) the contribution of both parts of the algorithm to the final floor result. In order to obtain a fair comparison in metric units, we need to project the image's floor without projective distortion to have a top view of it in real magnitude. Otherwise, the farther the segmented region is in the projective image, the less number of pixels it contains (despite representing a similar metric area than closer regions). We have calculated the homography from the image to the floor, and

[2]http://vision.in.tum.de/data/datasets/rgbd-dataset

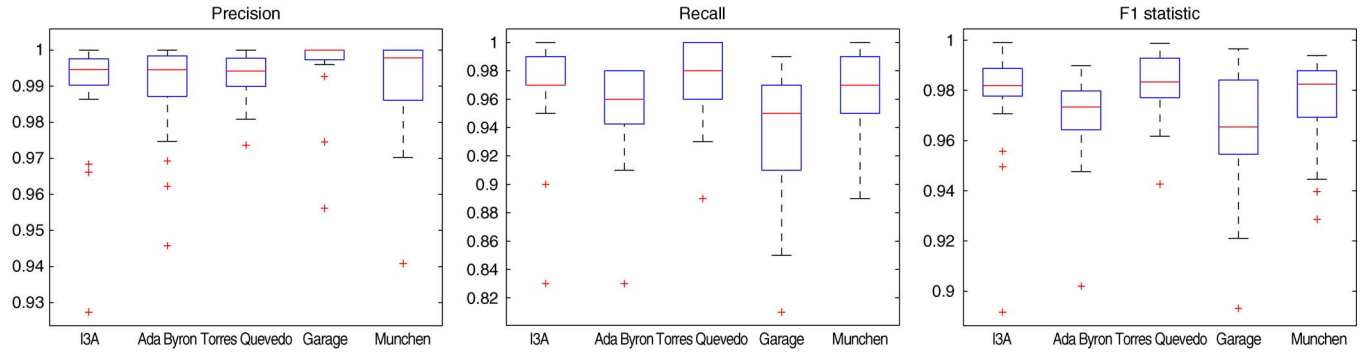


Fig. 15. Box plots obtained for the 150 RGB-D images with ground truth used for the system's evaluation in the five different scenarios. For each scenario, it is shown the median value, Q1 and Q3 quartiles, upper and lower limits, and outliers of the confidence intervals.

we have obtained the number of squared meters segmented by range and color algorithms. Table IV shows that the expansion of the range segmentation with color segmentation is essential in all scenarios. According to the three different scenarios we have defined before, those scenarios where there is no solar light incidence have the highest contribution of range segmentation. In spite of that, we almost obtain the 26% of the floor area from the RGB segmentation. For those scenarios that have a medium-low solar light incidence, we obtain a contribution of 50% approximately with both kinds of segmentations; thus, improvement of the results with sensor fusion is clearly shown. In addition, those scenarios where the presence of solar light is really high, color segmentation has the highest contribution where more than 70% of the segmented floor is obtained with this algorithm, and the limitations of range segmentation are drastically reduced.

Finally, the whole system considering range and visual information has segmented a medium value of 95, 83% of the floor's pixels with a medium precision of the 99, 18%. Fig. 15 shows more detailed quantitative results for each scenario. In the precision box plot, we can see that median values are over 99% and Q1 quartile over 98%, which corroborates a very low false positive index. In the recall box plot, we also obtain very good values. The recall is over 95%, and the Q1 quartile is over 90%; thus, the segmentation percentage is really high in most of cases. Despite the good values of precision and recall, the system does not provide perfect results, as shown in Figs. 13 and 14. However, the resultant error of the system is really small in comparison with and negligible for the addressed task.

Additional results are presented in the **video** attached as additional material. Two scenarios are shown, i.e., the I3A building and garage. The frames are divided in four pictures: The top left shows the RGB input image. The bottom left shows the range-based result. The bottom right shows the final result using or range data with image-based expansion. Moreover, the top right illustrates the sound map and commands sent to the user.

## V. Conclusion

In this paper, we have presented a robust system for NAVI, which allows the user to safely navigate through an unknown environment. We use a low-cost RGB-D system, from which we fuse range information and color information to detect obstacle-free paths. Our system detects the main structural elements of the scene using range data. Then, this information is extended using image color information. The system was evaluated in real scenarios and with a public data set providing good results of floor segmentation with 99% of precision and 95% of recall. The results show that this algorithm is robust to lighting changes, glows, and reflections. We have also released the data set used in this paper to provide a benchmark to evaluate systems for NAVI using RGB-D data.

## References

[1] D. Dakopoulos and N. Bourbakis, "Wearable obstacle avoidance electronic travel aids for blind: A survey," *IEEE Trans. Syst., Man, Cybern. C*, vol. 40, no. 1, pp. 25–35, Jan. 2010.

[2] R. Öktem, E. Aydin, and N. Cagiltay, "An indoor navigation aid designed for visually impaired people," in *Proc. IEEE IECON*, 2008, pp. 2982–2987.

[3] C. S. S. Guimaraes, R. V. Bayan Henriques, and C. E. Pereira, "Analysis and design of an embedded system to aid the navigation of the visually impaired," in *Proc. ISSNIP BRC*, 2013, pp. 1–6.

[4] J. Liu, C. Phillips, and K. Daniilidis, "Video-based localization without 3D mapping for the visually impaired," in *Proc. IEEE CVPRW*, Jun. 2010, pp. 23–30.

[5] F. Wong, R. Nagarajan, and S. Yaacob, "Application of stereovision in a navigation aid for blind people," in *Proc. Joint Conf. 4th Int. Conf. Inf., Commun. Signal Process., 4th Pacific Rim Conf. Multimedia*, 2003, vol. 2, pp. 734–737.

[6] D. Gutiérrez-Gómez, L. Puig, and J. J. Guerrero, "Full scaled 3D visual odometry from a single wearable omnidirectional camera," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2012, pp. 4276–4281.

[7] S. Gupta, P. Arbelaez, and J. Malik, "Perceptual organization and recognition of indoor scenes from RGB-D images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 564–571.

[8] H. Takizawa, S. Yamaguchi, M. Aoyagi, N. Ezaki, and S. Mizuno, "Kinect cane: Object recognition aids for the visually impaired," in *Proc. 6th Int. Conf. HSI*, Jun. 2013, pp. 473–478.

[9] Z. Wang, H. Liu, X. Wang, and Y. Qian, "Segment and label indoor scene based on RGB-D for the visually impaired," in *MultiMedia Modeling*, C. Gurrin, F. Hopfgartner, W. Hurst, H. Johansen, H. Lee, and N. O'Connor, Eds. Cham, Switzerland: Springer International Publishing, 2014, ser. Lecture Notes in Computer Science, pp. 449–460.

[10] B. Peasley and S. Birchfield, "Real-time obstacle detection and avoidance in the presence of specular surfaces using an active 3D sensor," in *Proc. IEEE WORV*, 2013, pp. 197–202.

[11] H. Schafer, A. Hach, M. Proetzsch, and K. Berns, "3D obstacle detection and avoidance in vegetated off-road terrain," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2008, pp. 923–928.

[12] Y. H. Lee and G. Medioni, "A RGB-D camera based navigation for the visually impaired," presented at the RGB-D: Advanced Reasoning With Depth Cameras Workshop Conjunction With RSS, Los Angeles, CA, USA, 2011.

[13] T. Seto and K. Magatani, "A navigation system for the visually impaired using colored navigation lines and RFID tags," in *Proc. IEEE Annu. Int. Conf. Eng. Med. Biol. Soc.*, Sep. 2009, pp. 831–834.

[14] S. L. Joseph *et al.*, "Semantic indoor navigation with a blind-user oriented augmented reality," in *Proc. IEEE Int. Conf. SMC*, Oct. 2013, pp. 3585–3591.

[15] A. Ray, L. Behera, and M. Jamshidi, "Sonar-based rover navigation for single or multiple platforms: Forward safe path and target switching approach," *IEEE Syst. J.*, vol. 2, no. 2, pp. 258–272, Jun. 2008.
[16] R. Volpe and R. Ivlev, "A survey and experimental evaluation of proximity sensors for space robotics," in *IEEE Int. Conf. Robot. Autom.*, 1994, vol. 4, pp. 3466–3473.
[17] I. Ulrich and I. Nourbakhsh, "Appearance-based obstacle detection with monocular color vision," in *Proc. 17th Nat. Conf. Artif. Intell. Twelfth Conf. Innov. Appl. Artif. Intell.*, 2000, pp. 866–871.
[18] Y. Li and S. Birchfield, "Image-based segmentation of indoor corridor floors for a mobile robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2010, pp. 837–843.
[19] R. Adams and L. Bischof, "Seeded region growing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 6, pp. 641–647, Jun. 1994.
[20] J. Alvarez, A. Lopez, and R. Baldrich, "Illuminant-invariant model-based road segmentation," in *Proc. IEEE Intell. Veh. Symp.*, 2008, pp. 1175–1180.
[21] H. Dahlkamp, A. Kaehler, D. Stavens, S. Thrun, and G. Bradski, "Self-supervised monocular road detection in desert terrain," in *Proc. Robot. Sci. Syst.*, Aug. 2006, [Online]. Available: http://www.roboticsproceedings.org/rss02/p05.html
[22] C. Dal Mutto, P. Zanuttigh, and G. Cortelazzo, "Fusion of geometry and color information for scene segmentation," *IEEE J. Sel. Topics Signal Process.*, vol. 6, no. 5, pp. 505–521, Sep. 2012.
[23] D. Lee, M. Hebert, and T. Kanade, "Geometric reasoning for single image structure recovery," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2009.
[24] V. Hedau, D. Hoiem, and D. Forsyth, "Recovering the spatial layout of cluttered rooms," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2009, pp. 1849–1856.
[25] A. Flint, D. Murray, and I. Reid, "Manhattan scene understanding using monocular, stereo, and 3D features," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2228–2235.
[26] J. Omedes, G. López-Nicolás, and J. J. Guerrero, "Omnidirectional vision for indoor spatial layout recovery," in *Frontiers of Intelligent Autonomous Systems*. New York, NY, USA: Springer-Verlag, 2013, pp. 95–104.
[27] G. López-Nicolás, J. Omedes, and J. J. Guerrero, "Spatial layout recovery from a single omnidirectional image and its matching-free sequential propagation," *Robot. Autonom. Syst.*, Apr. 5, 2014, to be published. [Online]. Available: http://dx.doi.org/10.1016/j.robot.2014.03.018
[28] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981.
[29] R. Raguram, O. Chum, M. Pollefeys, J. Matas, and J. Frahm, "USAC: A universal framework for random sample consensus," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 2022–2038, Aug. 2013.
[30] G. Bradski and A. Kaehler, "Computer vision with the OpenCV library," in *Learning OpenCV*. Sebastopol, CA, USA: O'Reilly Media, 2008.
[31] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, no. 6, pp. 679–698, Nov. 1986.
[32] N. Kiryati, Y. Eldar, and M. Bruckstein, "A probabilistic Hough transform," *Pattern Recog.*, vol. 24, no. 4, pp. 303–316, 1991.
[33] F. Meyer, "Color image segmentation," in *Proc. Int. Conf. Image Process. Appl.*, 1992, pp. 303–306.
[34] M. Al-Qutayri, J. Jeedella, and M. Al-Shamsi, "An integrated wireless indoor navigation system for visually impaired," in *Proc. IEEE Int. SysCon*, 2011, pp. 17–23.
[35] W. Martin, K. Dancer, K. Rock, C. Zeleney, and K. Yelamarthi, "The smart cane: An electrical engineering design project," in *Proc. Amer. Soc. Eng. Edu. North Central Section Conf.*, 2009, pp. 1–9.
[36] M. Zöllner, S. Huber, H.-C. Jetter, and H. Reiterer, "NAVI: A proof-of-concept of a mobile navigational aid for visually impaired based on the Microsoft Kinect," in *Proc. INTERACT*, vol. 6949, *Lecture Notes in Computer Science*, P. Campos, N. Graham, J. Jorge, N. Nunes, P. Palanque, and M. Winckler, Eds., 2011, pp. 584–587, Springer Berlin Heidelberg.
[37] W. Mayol-Cuevas, B. Tordoff, and D. Murray, "On the choice and placement of wearable vision sensors," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 39, no. 2, pp. 414–425, Mar. 2009.

**A. Aladrén** received the M.S. degree in industrial engineering from the University of Zaragoza, Zaragoza, Spain, in 2013.

He collaborates with the Robotics, Perception and Real-Time Group, University of Zaragoza, Zaragoza, and the Instituto Universitario de Investigación en Ingeniería de Aragón. He has worked in the development of systems for navigation assistance for the visually impaired. His current research interests are also focused on computer vision, 3-D visual perception, and autonomous robot navigation.

**G. López-Nicolás** (M'08) received the M.S. and Ph.D. degrees in industrial engineering from the University of Zaragoza, Zaragoza, Spain.

He is a member of the Instituto Universitario de Investigación en Ingeniería de Aragón, University of Zaragoza. He is currently an Associate Professor with the Department of Computer Science and Systems Engineering, University of Zaragoza. He has organized two editions of the workshop ViCoMoR (Visual Control of Mobile Robots) and a Special Issue in the journal *Robotics and Autonomous Systems*. His current research interests are focused on visual control, autonomous robot navigation, multirobot systems, and the application of computer vision techniques to robotics.

**Luis Puig** received the undergraduate degree in computer science from the Autonomous University of Puebla, Puebla, Mexico, in 2005. He received the Ph.D. degree in computer science and systems engineering from the University of Zaragoza, Zaragoza, Spain, in 2011.

Since 2013, he has been a Postdoctoral Fellow with the General Robotics, Automation, Sensing and Perception Laboratory, University of Pennsylvania, Philadelphia, PA, USA. His research interests are in the areas of computer vision, focusing on omnidirectional vision, visual odometry, image matching, localization, and their applications to robotics.

**Josechu J. Guerrero** (M'10) received the M.S. and Ph.D. degrees in electrical engineering from the University of Zaragoza, Zaragoza, Spain, in 1989 and 1996, respectively.

He is currently an Associate Professor and Deputy Director with the Department of Computer Science and Systems Engineering, University of Zaragoza. His research interests are in the areas of computer vision, particularly in 3-D visual perception, photogrammetry, visual control, omnidirectional vision, robotics, and vision-based navigation.