

Segmentation in Corridor Environments: Combining floor and ceiling detection

Sergio Lafuente-Arroyo, Saturnino Maldonado-Bascón, Hilario Gómez-Moreno,
and Cristina Alén-Cordero

GRAM, Department of Signal Theory and Communications,
University of Alcalá, Alcalá de Henares, Spain

sergio.lafuente@uah.es

<http://agamenon.tsc.uah.es/Investigacion/gram>

Abstract. Automatic segmentation from indoor images has several applications for mobile platforms. We address the problem of corridor segmentation and propose an approach by combining floor and ceiling detection. However, different difficulties may limit the accuracy of the system. To overcome these difficulties, a strategy is used in this paper to evaluate the degree of consistency of ceiling and floor guidelines. The method is based on computing the disparity between the hypothesized vanishing points by intersecting the boundaries par-wise. The approach is evaluated in a novel dataset. Our experimental validation confirms that the integration of floor and ceiling detection with the consistency model performs effectively and robustly. Because of the simplicity of the method, the image processing is quite fast and robust.

Keywords: Semantic segmentation · Corridor structure · Vanishing point.

1 Introduction

In indoor environments mobile platforms have to navigate along corridors to reach a room in order to perform a specific task. For this reason, the extraction of visual information can provide rich knowledge. The importance of scene understanding is a core computer vision problem for robot navigation in corridors. By inferring labels each pixel is associated with the class of its enclosing region (floor, ceiling or wall). A fundamental part of the indoor segmentation process is the floor detection step. However, several difficulties appear in visual floor detection associated to common specular reflections. The reflections on the floor may come from the ceiling lights, outdoor lighting from windows and doors, that even make it difficult sometimes for a human observer to distinguish the floor area. We tackle these problems as a challenge to propose a model based on the complementary detection of the floor and the ceiling in order to ensure a valid model for the corridor structure. How can we efficiently integrate both detections into the model? This is the question we want to answer with this work.

In man-made environments, such as corridors, sets of parallel lines intersect at points at infinity. Their projections in an image are called vanishing points (VPs).

In this paper we use a priori knowledge about the 3D scene in the sense that the corridor guidelines (wall-floor and wall-ceiling boundaries) intersect in the image at the vanishing point, which is located somewhere along the horizon line. Our objective is to estimate the common image intersection for the four corridor guidelines. Due to the noisy detection the imaged boundaries will generally not intersect in a unique point and the VP can be computed by intersecting the boundaries pair-wise. The disparity of these points is an indicator of the degree of validation of the detected boundaries.

The paper is structured as follows. In Section 2, a review of related works is introduced. Section 3 addresses our proposed method of detecting floor and ceiling. Section 4 discusses an algorithm to verify the consistency of candidate corridor guidelines. Section 5 demonstrates the experimental results with real images and Section 6 draws the final conclusions about the research performed in this paper.

2 Related Work

Existing literature contains several works on indoor floor segmentation based on computer vision, which can be easily classified depending on whether they use a purely appearance or geometric/homographic standpoint or those which combine both. In those approaches based on appearance, multiple visual clues from the environment are used for detection. In [1], a combination of color and gradient histograms to distinguish free navigable space is used. Due to over reliance on color based descriptors, the approach fails in homogeneous environments. A different approach in [2] uses a combination of vertical edges, thresholding and segmentation to approximate the wall-floor boundaries and then classify horizontal edges that lie on that boundary. This approach gives good results, robustly dealing with specular reflection on floor which is common in indoor environments. Nevertheless, it fails either when vertical edges are missed in the lower half of the image or when side walls are close to the robot.

Different approaches based on geometry exploit the ground plane constraint and focus on just finding the ground plane [3–6]. In [3], the motion between two images is modeled by a homography constraint as a criterion for ground plane detection. Optical flow is also used in [4] for ground plane detection. Both researches [3, 4] used a monocular camera, while dense point correspondences relied on stereo homographies in [5]. More recently, in [6] a combination of sparse optical flow and planar homography for ground plane detection is used. Aforementioned methods are computationally intense.

While purely appearance based approaches fail under homogeneity of appearance, geometric methods are robust enough at detecting features that define the floor. However, geometry based approaches need extra hints to segment the boundary that include the floor features. The approach of [7], which applies one of the methods that encompasses both geometry and appearance, is able of developing geometrical reasoning by searching the best fitting model, which is transformed into a full 3D model. Other interesting strategy [8] creates valid

box layout hypotheses by using detected line segments and virtual rays from orthogonal vanishing points. With the same approach of considering geometric backgrounds to improve scene interpretation, a method for supporting relations of indoor scenes from an RGBD (Red-Green-Blue-Depth) image is proposed in [9].

Recently, more and more computer vision tasks such as image classification have been solved by Convolutional Neural Network (CNN). As an example of application to indoor environments, we can find the work of Hazirbas [10]. This technique broadly surpasses other conventional approaches in terms of accuracy but it usually needs more processing time and memory. Therefore, further effort is required to explore new architectures in order to make semantic segmentation more efficient.

3 Floor and ceiling detection

This section describes a completely automated process of floor and ceiling detection in the image. The stages include detection of line segments, clustering and detection of boundaries.

3.1 Detection of line segments

Edges convey essential information for distinguishing separations. The popular Canny detector is used for this purpose in this work. Probabilistic Hough transform is then applied to the resulting edge image.

Unlike some previous work [2], which establishes an unique set of values for parameters, the proposed line extraction consists of two detectors. If no lines are extracted with the first detector, a new detection is applied with more flexible conditions. The second detector focuses on no remarkable lines. We tune the Canny threshold, t , in order to maintain a compromise between accuracy and robustness to noises and outliers. As the value of this parameter decreases, no remarkable lines can be detected but instead outliers may appear. The existence of outliers provoke a major dispersion of crossing points between pairs of lines. In figure 1 we plot the dispersion of crossing points and the percentage of images without detection and as functions of the Canny threshold. In terms of compromise, this parameter has been set to $t_1 = 30$ for the first detector and $t_2 = 15$ for the second one. By applying the above procedure, we obtain a set of line segments $\mathcal{L} = \{\mathbf{l}_1, \mathbf{l}_2, \dots, \mathbf{l}_N\}$ defined by their two endpoints.

Man-made environments include a lot of regularities due to their intrinsic structure. Detected lines in corridor images can be grouped into three categories regarding to the angle range: vertical lines, transversal lines and horizontal lines. The last ones are discarded for our purpose. Vertical lines correspond, in general, to walls boundaries such as doors and windows whereas transversal lines identify, in general, wall-floor and wall-ceiling boundaries. The angle ranges have been established by considering the appearance of imaged wall-floor and wall-ceiling boundaries with the camera fixed at different heights from the ground.

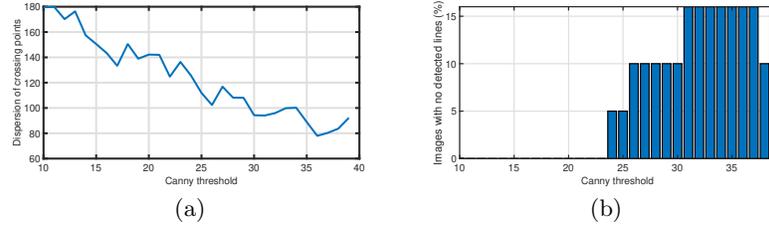


Fig. 1. The impact of Canny threshold on the accuracy of line detectors. We show (a) the average dispersion of crossing points between pairs of lines and (b) the percentage of images without detection as a function of the Canny threshold.

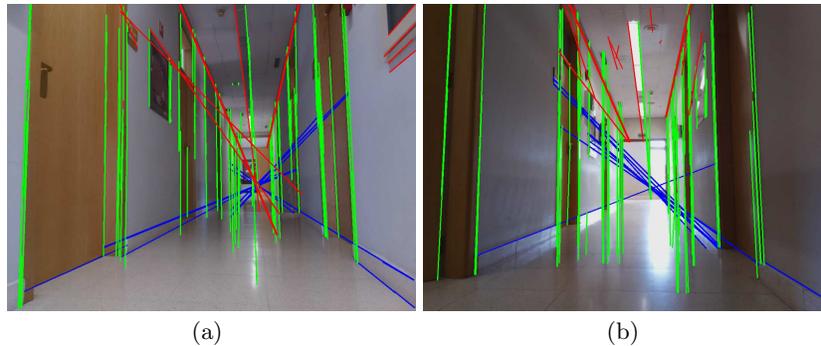


Fig. 2. Line extraction and classification: vertical (green), floor (blue) and ceiling (red).

At this point the transversal lines are classified in four sets: C_l and C_r includes, respectively, the line segments which correspond to potential left and right ceiling boundaries whereas the sets F_l and F_r include, respectively, the potential left and right floor boundaries. Figure 2 shows two examples of line detection, where the color of each line represents its category. It is worth noting that, in general, detected transversal lines include spurious edges. The sets F_l and F_r may contain line segments as effect of reflections, shadows and tile joints. Beside this, the sets C_l and C_r may include segments from the structure of the ceiling, upper doorframes and ceiling lights.

3.2 Clustering

Once detected the meaningful line segments, an agglomerative clustering scheme is used for each one of the four sets. Clustering is based on two features: slope and bias of each segment. In order to compute the bias it is considered the intersecting points of the detected segments with the borders of the image. Specifically, intersections of floor segments with the bottom image border and, on the other

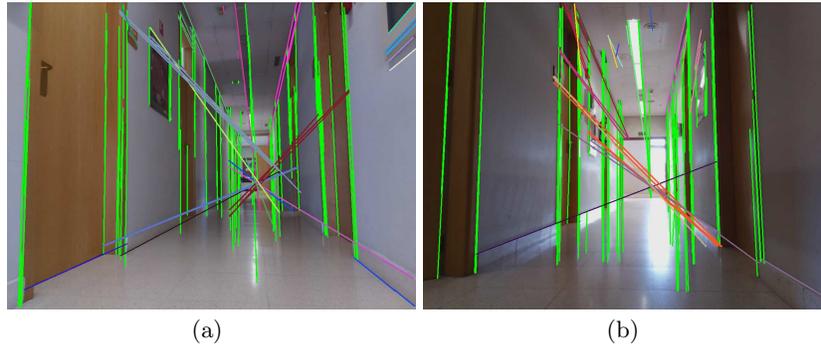


Fig. 3. Illustration of the clustering process where each cluster is depicted with a different color.

hand, intersections of ceiling segments with the upper image border. Both features slope and bias are integrated in a function to compute pairwise distances between two clusters C_m and C_n as:

$$d_{C_m, C_n} = \frac{1}{N_{C_m}} \frac{1}{N_{C_n}} \sum_{i \in C_m} \sum_{j \in C_n} \sqrt{\left(\frac{\Delta\theta_{ij}}{\pi}\right)^2 + \left(\frac{\Delta x_{ij}}{W}\right)^2} \quad (1)$$

where $\Delta\theta_{ij}$ is the angle difference between each pair of line segments \mathbf{l}_i and \mathbf{l}_j with range of values $[-\pi, \pi]$, Δx_{ij} is the X-coordinate step between intersection points of \mathbf{l}_i and \mathbf{l}_j with the image borders and W is the width of the image. The parameters $\frac{1}{N_{C_m}}$ and $\frac{1}{N_{C_n}}$ are the number of points of C_m and C_n .

The two clusters with the smallest distance are merged in each iteration and the operation is repeated until the distance d_{C_m, C_n} between the two closest clusters is larger than a certain threshold, which has been adjusted experimentally. Thus, the final clusters $\mathcal{C} = \{C_1, C_2, \dots, C_N\}$ are obtained and each cluster C_j is characterized by a prototype line \mathbf{t}_j . In order to compute each cluster prototype, we give more relevance to longer lines. The slope and the bias of each \mathbf{t}_j are computed as the weighted average of the segments of the cluster C_j . Figure 3 shows the resulting clusters for two examples, where lines of each cluster are depicted with the same color. The prototype lines of the different clusters become the candidate floor and ceiling boundaries for the following stage (see Figure 4).

3.3 Detection of boundaries

Given a set of candidate boundaries $\{\mathbf{t}_j\}_{j=1}^N$ generated by the clustering step, the goal is to find a function which estimates the strength of each candidate. In order to deal with this problem, it is proposed a weighted sum of scores based on two individual visual aims for each prototype line \mathbf{t}_i of F_l and F_r :

$$\phi(\mathbf{t}_i) = w_1\phi_1(\mathbf{t}_i) + w_2\phi_2(\mathbf{t}_i) \quad (2)$$

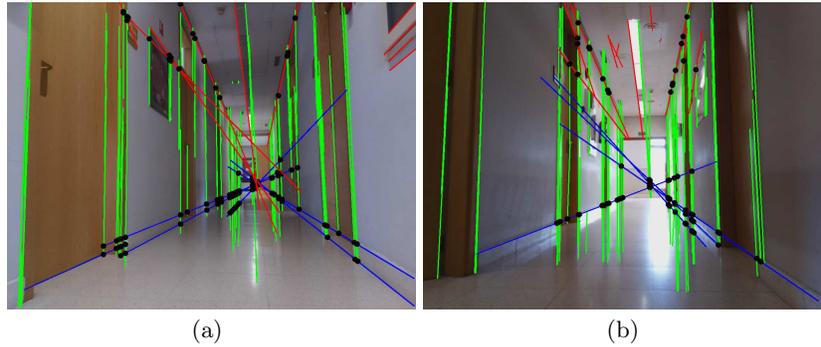


Fig. 4. Prototype lines and their intersection points with vertical lines.

where w_1 and w_2 are the weights and ϕ_1 and ϕ_2 are the individual scores which we describe in the following items:

- Intersections with vertical lines (ϕ_1). It is inspired by the fact that low endpoints of vertical lines delimit theoretically the floor boundaries in the image. Thus, the algorithm favours transversal lines with a higher number of intersections with vertical lines in the proximity of their low endpoints. As intersections of transversal lines with vertical lines may extend beyond the floor boundaries, due to reflections or shadows, a tolerance margin for intersections needs to be defined.
- Length of the prototype line (ϕ_2). Intuitively, very short line segments are frequently noisy, hence, their contribution should be constrained by comparison with long segments. This score favours longer line segments as candidates to floor boundaries.

In a similar way, the weighted sum of scores for the sets C_l and C_r introduces an extra visual aim (ϕ_3) corresponding to the maximum angle. The objective of ϕ_3 is to give more relevance to those line segments with higher slope. The slope of ceiling boundaries in the image is higher than the corresponding to those spurious line segments from upper doors or windows sides in the walls. Figure 4 shows the prototype lines of the different clusters and their intersection points with vertical lines. The output of this stage returns an array of weights associated to each one of the four sets (C_l , C_r , F_l and F_r). Then, it can be defined the boundary \mathbf{b} as the candidate line whose weighted sum $\phi(\mathbf{t}_j)$ is the highest among all candidates being also higher than certain threshold T :

$$\mathbf{b} = \mathbf{t}_j^* = \operatorname{argmax}_{\phi(\mathbf{t}_j) > T} (\phi(\mathbf{t}_j)). \quad (3)$$

This process is applied independently for each one of the four sets C_l , C_r , F_l and F_r . Then the left and right floor boundaries, denoted, respectively, as \mathbf{b}_1 and \mathbf{b}_2 , and the left and right ceiling boundaries, denoted, respectively, as \mathbf{b}_3 and \mathbf{b}_4 , are obtained. Figure 5 illustrates the four detected boundaries.



Fig. 5. Examples of detection of wall-floor and wall-ceiling boundaries.

4 Consistency of boundaries

Ideally, assuming perfect imaging condition and line segment extraction, parallel lines should intersect at a dominant VP as is shown in Figure 6(a). However, in the real world, there are pixel noise, image distortion, discretization errors, and line segment extraction errors, which make the problem much more challenging. In addition, an incorrect detection of floor and ceiling boundaries could cause disparity over all possible VP locations (see Figure 6(b)). From the four boundaries $\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3, \mathbf{b}_4$, we can compute the VP and use it for validating the floor and ceiling boundaries. Each pair of boundaries \mathbf{b}_i and \mathbf{b}_j defines a hypothesis VP as $\mathbf{v}_{ij} = \mathbf{b}_i \times \mathbf{b}_j$. Then, six VP estimations can be determined, denoting the set as $\mathcal{V} = \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_6\}$.

It is proposed a new strategy which models the impact of an incorrect boundary detection and is able to correct it. That is, the method detects whether one of the four boundaries does not fit well to the candidate VP. The strategy is based on building four partition sets $\{S_1, S_2, S_3, S_4\}$, where each one of them does not take into account one of the four boundaries. Thus, we define $S_1 = \{\mathbf{b}_2, \mathbf{b}_3, \mathbf{b}_4\}$, $S_2 = \{\mathbf{b}_1, \mathbf{b}_3, \mathbf{b}_4\}$, $S_3 = \{\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_4\}$ and $S_4 = \{\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3\}$. For each one of the four sets the algorithm determines three VP hypotheses for each pair of boundaries (see Figure 6(c)-6(f)). Let $\{d_1, d_2, d_3, d_4\}$ be the sum of distances among the three VP hypotheses associated respectively with the partitions $\{S_1, S_2, S_3, S_4\}$. In case of having a boundary \mathbf{b}_j not parallel to the other ones, its impact is included in all the partitions except in the partition S_j . Due to the method relies on distances to identify the incorrect boundary, if all distances $\{d_i\}_{i=1}^4$ with $i \neq j$ are greater than a fixed fraction of d_j , the corresponding boundary \mathbf{b}_j would be replaced by the prototype line with the second highest weighted sum ϕ . We apply recursively the procedure until reaching a dominant VP. The method ensures that the four boundaries that define the corridor structure are parallel.

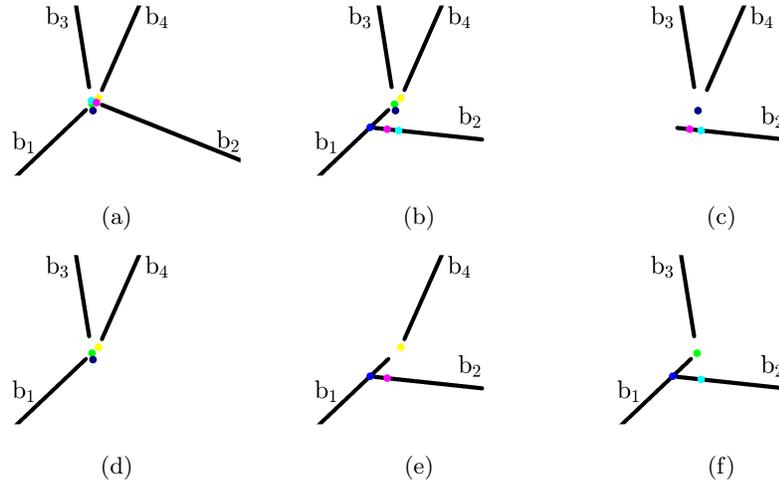


Fig. 6. Hypothesis VP based on detected boundaries. (a) Example with parallel boundaries. (b) Example with a not parallel boundary respect to the remaining ones. (c) Partition S_1 . (d) Partition S_2 . (e) Partition S_3 . (f) Partition S_4 .

5 Results

In order to check the performance of the proposed approach, we have generated a test dataset from different locations within the Politechnic School of the University of Alcalá. The dataset consists of 106 frames distributed in three sequences through the corridors of the building. For the camera, a 1024x768 resolution is selected.

Figure 7 shows the output results of some images from our dataset. Each corridor image is segmented into four possible regions corresponding to floor, ceiling and walls (left and right). Each one of these regions is enclosed by two boundaries and represented by a polygon of three vertexes: the points in which both boundaries intersect with the borders of the image and the vanishing point. Different colors are used to identify these regions. The black point is the detected vanishing point in which converge the four boundaries. We test the algorithm under different conditions. Thus, Figure 7(e)-(f) shows the robustness of the algorithm to changes of perspective. The performance of the system has been tested with partial occlusions in presence of persons (Figure 6(g)-(h)). It is noticeable that even when the ceiling-wall boundary is occluded partially, the weight of the occluded boundary is even greater than the corresponding to remaining candidates. Figure 7(m)-(n) includes two images with different distances to the end of the corridor. In addition, we have tested the robustness of the system to images captured with different camera heights and figure 7(m)-(n) shows two examples. On the other hand, some isolated images in different



Fig. 7. Output segmentation results (Best viewed in color): floor (red), ceiling (blue), left wall (green) and right wall (orange). First and third rows are the original images. Second and fourth rows are the output images. (a), (b), (e), (f) Different perspectives. (c), (d), (g), (h) Occlusions of boundaries due to the presence of persons. (i), (j), (m), (n) Different distances to the end of the corridor. (k), (l), (o), (p) Images captured with different heights of the camera from the floor.

environments, such as hospitals, have been tested with the same approach. Figure 8 illustrates some results.

When the robot is close to side walls (Figure 9(a)-(d)), the ceiling region is reduced or does not exist in the image. It may cause errors as in Figure 9(e)-(f)), where several guidelines are incorrectly detected. However, two criteria have been established in order to validate the ceiling detection: 1) ceiling area in the image must be more reduced than floor area and 2) ceiling boundaries must extend to both sides of the vertical line that passes through the vanishing point. If the ceiling detection does not follow one of both conditions, ceiling detection is not valid. In these cases, the system only depicts the floor region, as we can see in Figure 9(g)-(h).

In order to obtain quantitative results, we have labelled the ground truth of floor and ceiling in the images of the dataset. For this purpose we have developed an application Python that allows to define manually the ceiling and floor boundaries. In the test phase a mask is generated for each image by comparison of both manual and automatic segmentation, as we can see in Figure 10. Blue pixels represent those ones that were classified correctly, red correspond to misclassified pixels and green are not detected pixels with respect to the ground truth. On average, 93.3% of ceiling and floor pixels labelled manually were classified correctly in our dataset and only 4.15% of the pixels of the image were incorrectly segmented.

All processing steps have been implemented in Python, making use of the Numpy package and the OpenCV library. In order to decrease the processing time, we have reduced the image size to 512×384 while keeping the same results reported above. On an Intel Core i5-7500 CPU, the implementation takes only 30.78 ms on average per image.

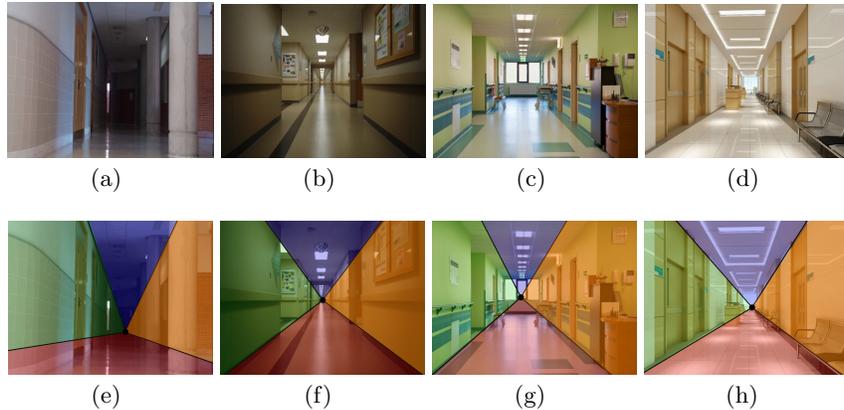


Fig. 8. Output segmentation results in different corridor environments. Top row: original frames. Second row: segmentation results.

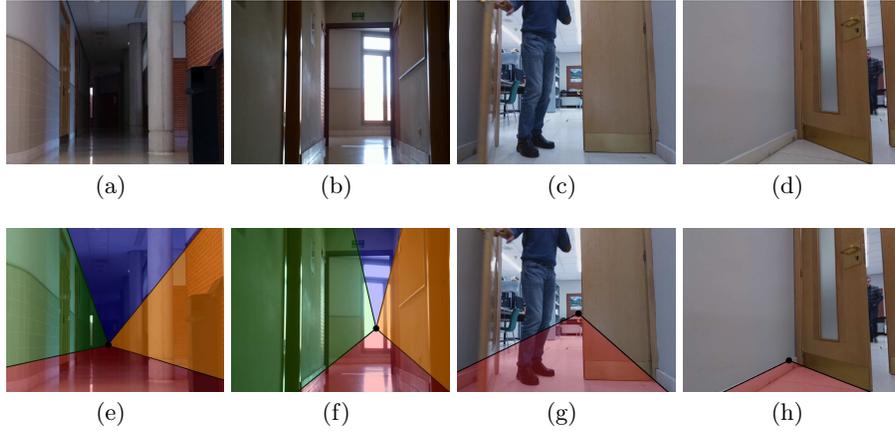


Fig. 9. Examples of views close to side walls. Top row: original frames. Second row: segmentation results.

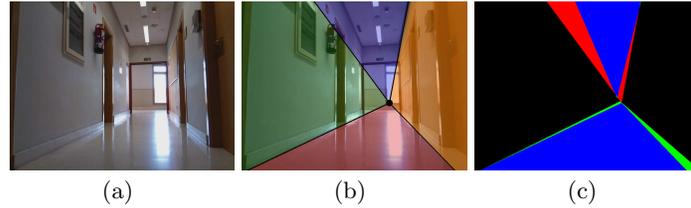


Fig. 10. Example of segmentation mask. (a) Original image. (b) Segmentation. (c) Output mask: correct segmentation (blue), incorrect segmentation (red), not detected (green).

6 Conclusions

In this paper we have presented a corridor segmentation algorithm based on a combined floor and ceiling detection. An important advantage over traditional floor-based detectors is the fact that ceiling detection can help to correct floor detection in some cases. The disparity of vanishing points hypotheses is used as indicator of the degree of validation of the boundaries that define the corridor structure. The quantitative experiments in a novel and challenging dataset are conducted to validate the effectiveness of the proposed method.

Acknowledgements. This work has been supported by project PREPEATE, with reference TEC2016-80326-R, of the Spanish Ministry of Economy, Industry and Competitiveness, and by the project with reference CCGP2017-EXP/054 of the Government of the Community of Madrid.

References

1. Lorigo, L., Brooks, R., Grimsou, W. Visually-guided obstacle avoidance in unstructured environments. In: Intelligent Robots and Systems IROS'97. vol.1, pp. 373-379 (1997)
2. Yinxiao, L., Birchfield, S.T. Image-based segmentation of indoor corridor floors for a mobile robot. In: 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems. Taipei, Taiwan (2010)
3. Zhou, J., Li, B. Robust ground plane detection with normalized homography in monocular sequences from a robot platform. In: IEEE International Conference on Image Processing, pp. 3017-3020 (2006)
4. Cui, X.N., Kim, Y.G. Floor segmentation by computing plane normals from image motion fields for visual navigation. In: International Journal of Control, Automation and Systems, vol. 7(5), pp. 788-798 (2009)
5. Fazl-Ersi, E., Tsotsos, J. Region classification for robust floor detection in indoor environments. In: IEEE Image Analysis and Recognition, pp. 717-726 (2009)
6. Kumar, S., Dewan, A., Madhava Krishna, K.: A Bayes filter based adaptive floor segmentation with homography and appearance cues. In: 8th Indian Conference on Computer Vision Graphics and Image Processing (ICVGIP), pp. 54. ACM, (2012)
7. Lee, D., Hebert, M., Kanade, T. Geometric reasoning for single image structure recovery. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2136-2143 (2009)
8. Baligh-Jahromi, A., Sohn G. Geometric context and orientation map combination for indoor corridor modeling using a single image. In: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLI-B4. XXIII ISPRS Congress, pp. 295-302, Prague (2016)
9. Silberman, N., Hoiem, D., Kohli, P., Fergus, R. Indoor segmentation and support inference from RGBD images. In: Fitzgibbon A., Lazebnik S., Perona P., Sato Y., Schmid C. (eds) Computer Vision – ECCV 2012. LNCS, vol 7576. Springer, Heidelberg (2012)
10. Hazirbas, C., Ma, L., Domokos, C., Cremers, D.: FuseNet: Incorporating Depth into Semantic Segmentation via Fusion-Based CNN Architecture Asian Conference on Computer Vision (ACCV). LNCS, vol. 10111, pp. 213-228. Springer, Heidelberg (2016)