Detecting Doors Edges in Diverse Environments for Visually Disabled People

Mohamed Ibrahim Habib,
mhabib@seu.edu.sa
Computer Science Department, College of Computing and Informatics, Saudi Electronic University,
Riyadh 11673, Saudi Arabia,
Faculty of Engineering, Port Said University, Port Said, 42526 Egypt,

Summary

It is a challenge for visually impaired people to access unfamiliar environments independently, hence the quality of life is reduced, and safety of life is compromised. An accurate and reliable door detection system comprising of way finding and indoor navigation can be beneficial for a large number of autonomous and mobile applications for visually impaired people. This paper illustrates an image-based door detection scheme for visually impaired people using stable features (edges and corners) including color averaging and image resizing. Simulation results show that the proposed scheme shows a significant improvement when compared with existing scheme.

Key words:
Stable features, Contrast adjustments, Color averaging and Image resizing, Visual impairment.

1. Introduction

According to the World Health organization (WHO) 285 million people are estimated to be visually impaired around the world in 2011. Visual impairment has two subcategories blindness and low vision. Among the figure presented, 39 million were blind and 246 million had low vision [1]. It is a challenge for visually impaired people to access unfamiliar environments independently, hence the quality of life is reduced, and safety of life is compromised. Indoor environments often contain well-regulated artificial objects with a particular size, shape and location. Doors act as important landmarks for navigation, they work as a channel to connect one place to another. The challenge of a computer vision system is to be capable of recognizing natural structures and providing the necessary semantic interpretations of their environment. An accurate and reliable door detection system comprising of way finding and indoor navigation can be beneficial for a large number of autonomous and mobile applications for visually impaired people.

Visually impaired people have to deal with a challenging task of way finding in unfamiliar environments every day. X. Yang, and Y. Tian [2] presented a general geometric model for door shape characterization that utilizes the general and stable features of doors (i.e. edges and corers). In this method the computational time for corners detection was directly proportional to the number detected corners, for large number of detected corners, the computational time increases. X. Yang et al. [3] presented a context based indoor object detection model that combines edges and corners for general geometric shaped door detection. This model further classifies the doors by extracting character information. However, the major problem this model confronts during automatic door detection is that many doors do not have contextual information present.

R. Sekkal et al. [4] developed a framework for tracking and door detection that exploits geometric features of corridors. This framework detects visual features (such as lines and vanishing points) that are joined together to differentiate the floor and wall planes and then it identifies doors inside the image sequences. Detected doors initialize an edge based 2D door tracker for tracking. Mahmood and Kunwar [5] have proposed a model that uses a self-organizing neural scheme that utilizes features-based classification and Kohonen [6] Self-Organizing Map (SOM) for door detection. This model extracts the edges and group them into line segments using line detection. This model also extracts generic and stable features (such as concavity, bottom door gap, distance between vertical lines etc.) and feed them to SOM classifier as input for training and classification of doors.

However, door gap and concavity are not always visible in every case. Hensler et al. [7] presents a door detection technique based on the use of camera and laser. In this model AdaBoost algorithm takes weak classifiers (color, texture, door gap, door frame, bottom gap etc.) as input to detect a door in an image. However, some of the weak classifiers (texture, door gap, bottom gap, etc.) are not always visible/present in every situation. Sturm et al. [8] discussed a detection, tracking and learning articulation models for cabinet doors and drawers without any use of artificial marker. The model uses the depth images acquired from an active stereo system to detect rectangles by sampling. This strategy segments the point clouds into planes, and then iteratively fits rectangles to each plane separately [8].

Manuscript received May 5, 2021
Manuscript revised May 20, 2021
https://doi.org/10.22937/IJCSNS.2021.21.5.2

A two-layer navigation model for autonomous indoor exploration has been presented by Zhang et al. [9]. A stereo vision-based algorithm (high-level) and an improved dynamic window approach have been used in this model for door detection and to make a robot avoid obstacles (low-level). In case of several doors' detection, the closest door is selected, and the goal point is determined by a heuristic method according to the distance of the door. Hierarchical Composition Model in [10] is based upon the incorporation of context information. The process of door detection depends on the standard size range. The distance between the camera and the object is estimated by calibrating the pixels of the object in image with the real size of doors. However, the model is limited for certain environments, where contextual information is available. Hinkenjann et al. [11] discusses a real-time approximate simulation of some camera errors and their effects on common computer vision algorithms for robots.

They have presented a software framework for real-time post processing of image data. This framework is useful to tune robotic algorithms to make them more robust to imperfections of real cameras. Borgsen et al. [12] extraction the parts of doors from point clouds based on constraint region growing. Gaussian probabilities are used to give these parts a weight that are combined to create an overall probability measure. Crossing a door is usually a necessary action for an indoor surveillance robot, however, it is difficult for a large robot to cross narrow doors. Dai et al. [13] presents a model for detecting, locating, and crossing a door for an indoor surveillance robot with Kinect.

The process used in [13] to detect and locate doors is based on RGB-D data collected form Kinect. This model forms a connection between the robot coordinate and the world coordinate system and design a nonlinear adaptive controller so that the robot can move perpendicularly through the door. The obstacle detection system, to assist the visually impaired in [14], focuses on indoor environments and classification of typical hindrances utilizing a 3D sensor. The system reconstructs the point clouds by means of both depth and color data. Point clouds are further filtered and are passed to the segmentation stage. Eventually, plane-based segmentation is followed by the segmentation of the remaining objects [14].

A recent approach using deep learning model was presented in [15]. An indoor signage detection system model was implemented to detect four types of signage: exit, wc, disabled exit and confidence zone. The effectiveness and high precision of the proposed recognition scheme was demonstrated by the results of the experiments. Results were able to achieve a recognition rate of 99.8%. Trabelsi et al. [16] proposed a multi-modal methodology to assist recognizing object and doors. A complex-valued neural networks (CVNNs) and multi-label learning techniques was implemented. The model builds one CVNN for each label, with the expected labels being aggregated later to create the multi-label vector. Extensive tests on two newly obtained multi-labeled RGBD datasets have shown the efficacy of the proposed techniques.

The rest of the paper has been organized into four sections. The next section discusses our proposed scheme and algorithms in details. Section three demonstrates and discusses the experimental results. Section four concludes this work by presenting conclusion and future directions.

2. Proposed Scheme

Consider the input image (I_i) from the camera, having the dimensions $x * y * \beta$, where $x = [1, 2,... X]$, $y = [1, 2,... Y]$ and $\beta = [R, G, B]$. Images having larger dimensions consume more time. I_{in} is down sampled to reduce the computational cost and converted to gray scale. Figure 1 and Figure 2 show the down sampled and gray scale input image respectively.

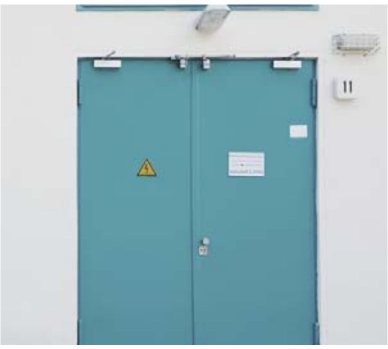

Fig. 1. Down sampled input image

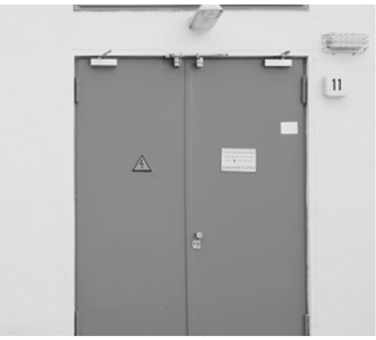

Fig. 2. Gray scale input image

Since, the whole technique is based on edges and corners, canny edge detector is being used for the extraction of the edges, and the edge map determines the number of corners. To improve the computational cost a formula has been developed to assign threshold values, this formula generates minimum number of corners but at the same time enough for accurate detection of a door in an image. In the proposed method, first step is down sampling and when the image has been down sampled, standard deviation of the whole image is calculated to find out the high threshold (H_{th}) of canny edge detector [17] using equation (1).

$$H_{th} = \alpha \times \sqrt{Std} \quad (1)$$

We have experimented with different range of values for H_{th}. Threshold values for canny edge detector has been assigned from different ranges based on the standard deviation such as:

$$H_{th} = \begin{cases} 0.1 & if\ Std < 10 \\ 0.15 & if\ 10 < Std < 15 \\ 0.2 & if\ 15 < Std < 20 \end{cases} \quad (2)$$

This particular range setting worked pretty well. Hence, a formula was derived (see eq. 1) that would automatically find H_{th}. The constant α (α = 0.066) in equation 1 brings the value closer to those values that have been set empirically for experimentation. Once, the value of H_{th} is determined, low threshold can be calculated by:

$$L_{th} = 0.4 \times H_{th} \quad (3)$$

This process does not only produce a good edge map but also reduces the computational cost comprehensively at the same time. The second step in the process is to extract the edges. Threshold values obtained in the previous step are fed to the canny edge detector to produce an edge map (by using the following method).

$$I_{ed} < -edge(I_{in}, H_{th}, L_{th}) \quad (4)$$

In the proposed scheme, we have experimented with different edge detectors, but canny edge detector produces the better results (see fig. 3) than others for the current problem.

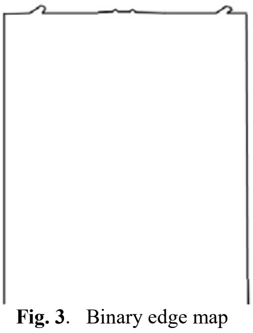

Fig. 3. Binary edge map

The proposed system applies a corner detector based on global and local curvature properties for corner detection [18]. The system extracts the edge contours from the edge map to fill the gaps in the edge map, finds T-junctions and marks them as T-corners. Then, the curvature value of each pixel of the contour is computed using equation 6.

$$Con_{ex} < -I_{ed} \quad (5)$$

$$Cur_{Con}(\mu,\epsilon) = \frac{\dot{M}(\mu,\epsilon)\ddot{N}(\mu,\epsilon)-\ddot{M}(\mu,\epsilon)\dot{N}(\mu,\epsilon)}{(\dot{M}(\mu,\epsilon)^2+\dot{N}(\mu,\epsilon)^2)^{1.5}} \quad (6)$$

Where,

$$\dot{M}(\mu,\epsilon) = m(\mu) \otimes \dot{g}s(\mu,\epsilon) \quad (7)$$

$$\ddot{M}(\mu,\epsilon) = m(\mu) \otimes \ddot{g}s(\mu,\epsilon) \quad (8)$$

$$N(\mu,\epsilon) = n(\mu) \otimes \dot{g}s(\mu,\epsilon) \quad (9)$$

$$\ddot{N}(\mu,\epsilon) = n(\mu) \otimes \ddot{g}s(\mu,\epsilon) \quad (10)$$

Here, gs(μ,ϵ) is the Gaussian of width ϵ, with first and second derivatives being g ̇s(μ,ϵ) and(g) ̈s(μ,ϵ). m(μ) and n(μ) parameterize the curve by the arc length μ and ⊗ is convolution operator.

The resultant image after applying corner detector has corners marked on it as shown in figure 4.

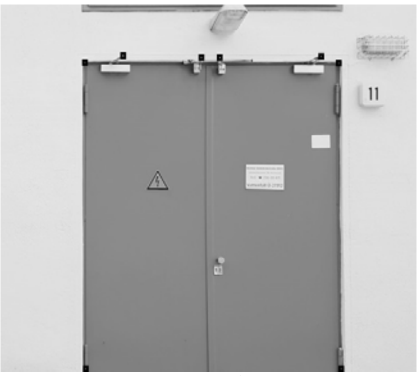

Fig. 4. Detected corners

Absolute value of curvature at every point of contour is calculated. Points that have local maximum curvature are selected as initial corner candidates. Adaptive curvature threshold and angles of corner candidates are calculated to truncate round and false corners due to noise and trivial details. Table 1 (see table 1 below) shows the total number of corners detected. z, m(z) and n(z) are the coordinates of each detected corner.

$$Gr = grouping(Corners) \quad (11)$$

Table 1: Coordinates of Detected Corners in Figure 4

Detected Corners (z)	m(z)	n(z)
1	108	101
2	12	100
3	13	41
4	109	42
5	111	37
6	2	105
7	119	33
8	2	122
9	119	118

The generated corners are grouped together in 4-corner groups. All the 4-corner groups are filtered by taking into account lengths and directions of each group for the removal of unwanted groups. There are 4 corners in the geometric model of a door. Therefore, the maximum number of corners to form a group is four. Figure 5 shows the geometric model of a door. Yang and Tian [2] did not check for the length difference between vertical lines made by each corner group which is very important, the ratio (see eq. 12) between both the lengths should be equal or below 0.1 for better accuracy.

$$Rd_{ab} = \frac{\sqrt{((m_a - m_b)^2 + (n_a - n_b)^2)}}{Diogn} \quad (12)$$

$$Dl_{ab} = arctan\ \frac{|m_a - m_b|}{|n_a - n_b|} \times \frac{180}{\pi} \quad (13)$$

Where, *Diogn* represents the length of the diagonal of an image. Rd$_{ab}$ should be in the range of (0 - 1) and Dl$_{ab}$ should not be less than 0 degrees and greater than 90 degrees. Each 4-corner group must pass through all the geometric requirements.

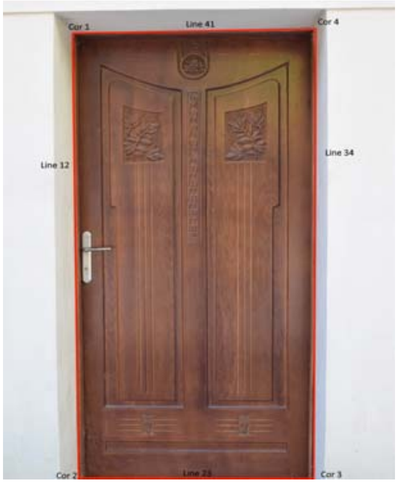

Fig. 5. Geometric model.

The 4-corner groups that fulfill the criteria as in equation 14 (see eq. 14) are selected as door-corner candidates, which will be further tested for door detection by making a 4 cornered frame of each group.

$$Matching = Fillratio(I_{ed}, 4 - CorneredFrame) \quad (14)$$

To determine if a door candidate matches to the real door frame, we combine the 4-cornered frame with the edge map. Edge map is used as a reference map for the matching process. Sometimes more than one final door-corner candidates match with the door frame. In this case, if the overlapped area of two detected doors is large enough, both are merged as one. A method known as fill-ratio (see eq. 15) is used to measure the deviation between the line made by 2 corners and the edge map as shown in figure 6 and figure 7 (see Fig. 6 and Fig. 7). Fill-ratio of 2 corners Cor(a) and Cor(b) can be defined as:

$$Fillrat(a, b) = \frac{Olap(a,b)}{Len(a,b)} \quad (15)$$

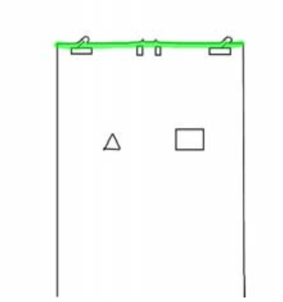

Fig. 6. Fill ratio horizontal line.

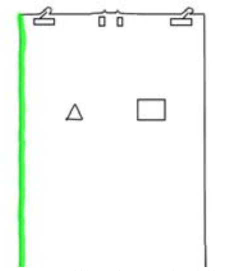

Fig. 7. Fill ratio vertical line

To overcome the issues related to the existing method such as accuracy, some additional steps are introduced just like, canny threshold setting that we used in the initial step to reduce the computational cost.

This technique fails to remove the false candidate in case of multiple final door candidates. Color averaging of the detected door candidates get rid of this problem. After the final door candidates are acquired, area segmentation of the detected door candidate on the real image takes place. The area covered by the door is divided into 16 segments. Average color of each segment is calculated. Every detected door is segmented into 16 parts. Each part is a matrix of 40x30 in a 160x120 image. Once we find out the average color of all the masks, standard deviation of all the masks is calculated to find out the level of overall variation in color for that particular candidate. In some cases, there are large windows in an image that can also be taken as a door. Therefore, to differentiate between doors and windows standard deviation is used. Doors usually have same color throughout its area; hence the value of standard deviation should be low. The candidate with the lowest value is selected as the final door candidate. For instance, let us consider a matrix A as shown in table 2.

Table 2: Matrix

40	50	35
50	45	30
35	56	70

Standard deviation can be calculated using the following equation.

$$S = \sqrt{\frac{\sum(x_i - \bar{x})}{n-1}} \quad (16)$$

Where, x_i represents the observational values, and n is the total number of observational values. $\bar{x}$ is the mean value which can be calculated as

$$\bar{x} = \frac{\sum(x_i)}{n} \quad (17)$$

Since, we are dealing with small images (160x120) and (320x240) because they result in detection of a smaller number of corners which consumes less time as compared to the images with large resolution. We can relate the door detection on small images to large images, which will eventually save us a lot of time. All of the standard resolutions increase with a certain factor e.g. the image with a resolution of 160x120 can be related to a 320x240 image with a factor of 2, likewise resolution of 160x120 can be linked with 320x240 with a factor of 3. Hence, the 4-corner candidates are simply multiplied with a certain factor to link it with a certain resolution.

3. Results

We have collected a database of 189 images from a wide variety of environments. These images were of doors with different colors, textures, partially occluded, partially opened/closed, elevators, different viewpoints, light changes and with obstructions. The accuracy achieved for 320x240 images is 93.5% and a percentage of 1.3% for false detection. Furthermore, images with a resolution of 160x120 were also tested and achieved an accuracy of 91.2% and 5.5% for false detection. Our database has been categorized in 5 categories and these 5 categories involves difference in background complexity, deformation, occlusion, partially opened/closed doors, obstruction and illumination and scale changes. First three categories i.e. simple, medium, and complex represent an increase in image background complexity, occlusions, perspective deformations etc. Fourth category is of partially opened or closed doors and the fifth category represents images with obstructions in front of the doors.

Figure 8 shows the results of detected doors, Images from 1 to 4 represents the simple category, while from 5 to 8 represents the medium category, and images from 9 to 12 represents the complex category.

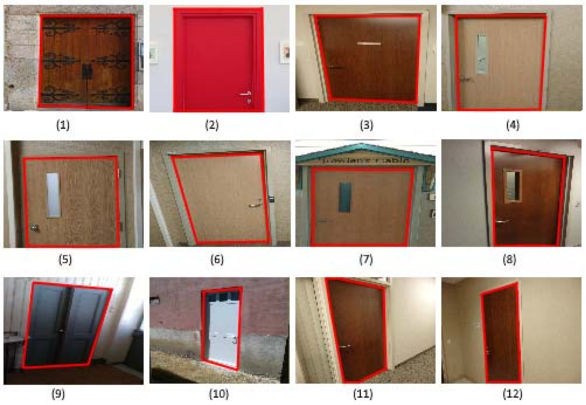

Fig. 8. Results of detected doors of Simple, Medium, and Complex categories.

Figure 9 represents doors detection with obstruction, and Figure 10 represents the category of partially opened/closed doors. It can be observed that the proposed technique is efficient enough to accurately detect the corners of the doors, even when it comes to the partially closed doors.

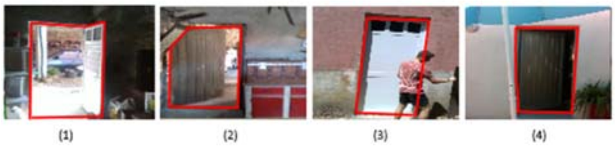

Fig. 9. Results of detected doors of obstruction category.

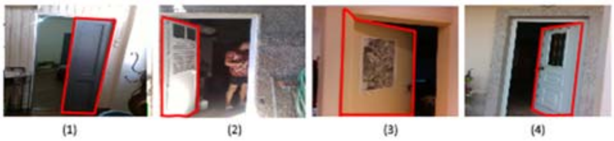

Fig. 10. Results of detected partially open/close doors.

Table 3 shows the results of images for resolution 320x240. Simple category has an accuracy percentage of 98.2% and a false/no-detection percentage of 1.8%. Medium category has an accuracy of 94.4% and a false/no detection percentage of 5.26%. Complex category has an accuracy of 88.0% and a false/no-detection percentage of 12%.

Table 3. Results For 320x240 Resolution Images.

Category	Total images	True %age	False/No detection %
Simple	55	98.2	1.8
Medium	95	94.4	5.26
Complex	50	88	12
Total	200	93.5	6.35

Table 4 follows the same pattern as it was observed in table 1 (corners). The existing method did not have any automatic threshold setting procedure for canny edge detector, and had a static threshold for all images, which resulted in more deterioration in results for resolution of 160x120. The proposed method assigns the threshold value according to the standard deviation of the image. Images with less variation in colors would require canny threshold to be low to extract a good edge map while images with more variation in color requires canny threshold to be high to extract a good edge map. This procedure helps to reduce the computational cost even more, table 4 shows the results for detection speed.

Table 4. Results For 160x120 Resolution Images

Category	Total images	True %age	False/No detection %
Simple	55	98.2	1.8
Medium	95	93.6	6.31
Complex	50	84	16
Total	200	91.2	8.7

Table 5 displays the results achieved in the presence of an obstruction or partially opened/closed doors. The existing method did not include partially opened/closed doors or doors with obstruction in front of them. In the proposed method operates on images with obstruction and partially opened/closed doors and produce good results because of automatic threshold setting of canny edge detector. Images with any kind of obstruction produce complex edge maps which result in greater number of detected corners which eventually increases the overall time consumption. Automatic threshold helps to produce less complex edge maps that helps in achieving good results in less time.

Table 5. Results For 160x120 Resolution Images

Category	Total images	True %age	False/No detection %
Partially opened/closed	12	67	33
Images with obstruction	12	75	25
Total	24	70.05	29

Table 6 demonstrate the results for the same image but with different resolutions. Detection time increases as the resolution of image increases because of a greater number of detected corners. It is obvious from the table 6 that as the resolution keeps on increasing, the number of corners would also be increasing, and the time consumed for detection will also increase. To resolve this issue, we can relate a detected door in a small image to a large image by multiplying the coordinates of the detected door with a certain factor, e.g. a door detected in an image with a resolution of 160x120 can be related to the same image with a resolution of 640x480 by a factor of 4. By simply multiplying the coordinates of the detected door with 4 and the result for the resolution of 640x480 can be obtained with the help of a smaller image. This would help in reducing the detection time drastically. The overall accuracy achieved by images of 160x120 resolution is 91.2% which is sufficient to be used in practical applications.

Table 6. Results For 160x120 Resolution Images

Resolution	Number of corners	Time (seconds)
160 x 120	23	1.6
320 x 240	48	8
640 x 480	64	45

This paper presents a novel approach for automatic door detection, the major differences of the proposed scheme from the existing methods are as follows:

- The existing scheme proposed some assumptions such as both the vertical lines of the door frame should be visible. However, the assumption can be ruled out because even if one of the 2 vertical lines of the door frame is occluded, the proposed method can still detect a door.
- Edge detector threshold values are automatically assigned to produce an edge map. It would eventually result in less detection of corners, which would reduce the overall time consumption.
- As a geometric door model has 2 vertical lines and 2 horizontal lines. The existing scheme did not check for the length difference between vertical lines of the detected door. This check would reduce the false detection.
- The proposed scheme has been tested successfully on partially opened/closed doors and results in good detection.
- The proposed scheme has also been tested successfully on images with different types of obstructions in front of the doors.
- Selection of the best candidate as a detected door results from finding out the average color of the area occupied and the overall standard deviation. This process improves the true rate.
- Detection on small images can be related to larger images.

4. Conclusion

In this paper, we have improved the performance of an existing scheme of door detection. First of all a geometric model of a door is established i.e. four corners and four lines connecting those corners then the standard deviation of the whole image is calculated which would automatically assign the threshold values for canny edge detector, to produce a good edge map. To reduce the computational cost, the original image is down sampled. The down sampled images are then smoothed with a Gaussian smoothing filter. Once the image is smoothed, canny edge detector is applied to extract an edge map. After the extraction of an edge map, a corner detector is applied on the edge map to find the corners. Corners are then divided into different groups of 4 corners. These corners groups are matched with the edge map to get final door candidate/candidates. Finally, to remove more than one final candidates, standard deviation for each final candidate is calculated and the one with the lowest value is selected as the detected door in an image. This is because doors normally have the same colors throughout. Large windows in an image sometimes are also detected as doors along with the real doors, hence by making selection based on standard deviation improves the accuracy. We have evaluated our method on a large database collected from a wide variety of environments, the proposed algorithm achieves 91.2% for 160x120 resolution which was 87.3% in the existing algorithm and for 320x240% resolution, the accuracy is 93.5% which was 91.7% in the existing scheme.

The future work will focus on improving the accuracy and detection time for larger images, distinguishing between doors, elevators, windows and marking them accordingly. More generic features for door detection like doorknobs can be incorporated for achieving further accuracy.

References

[1] World Health Organization, "*Global data on visual impairment.*" [Online]. Available: https://www.who.int/blindness/publications/globaldata/en/. [Accessed: 20-Mar-2021].

[2] X. Yang, and Y. Tian, "*Robust door detection in unfamiliar environments by combining edge and corner features*," IEEE Computer Vision and Pattern Recognition Workshops, pp. 57-64, 2010.

[3] X. Yang, Y. Tian,, C. Yi, and A. Arditi, "*Context-based indoor object detection as an aid to blind persons accessing unfamiliar environments,*" International conference on Multimedia, pp. 1087-1090, 2010

[4] R. Sekkal, F. Pasteau, M. Babel, B. Brun, and I. Leplumey, "*Simple monocular door detection and tracking*," IEEE International Conference on Image Processing, pp. 3929-3933, 2013.

[5] F. Mahmood, and F. Kunwar, "*A Self-Organizing Neural Scheme for Door Detection in Different Environments*," International Journal of Computer Applications, vol. 60, no. 9, pp. 13-18, 2013.

[6] J. Jelínek, "*Kohonen map modification for classification tasks*," Proceedings of the 11th International Conference on Agents and Artificial Intelligence, 2019.

[7] J. Hensler, M. Blaich, and O. Bittel, "*Real-time Door Detection Based on AdaBoost Learning Algorithm*," International Conference on Research and Education in Robotics, pp. 61-73, 2009.

[8] J. Sturm, K. Konolige, C. Stachniss, and W. Burgard, "*Vision-based detection for learning articulation models of cabinet doors and drawers in household environments*," IEEE International Conference on Robotics and Automation, pp. 362-368, 2010.

[9] H. Zhang, L. Dou, H. Fang, and J. Chen, "*Autonomous indoor exploration of mobile robots based on door-guidance and improved dynamic window approach,*" IEEE International Conference on Robotics and Biomimetics, pp. 408-413, 2009.

[10] C. Chen, and Y. Tian, "*Door detection via signage context-based hierarchical compositional model*," IEEE Computer Vision and Pattern Recognition Workshops, pp. 1-6, 2010.

[11] A. Hinkenjann, T. Roth, J. Millberg, H. Yun, and Y. Li, "*Real-time simulation of camera errors and their effect on some basic robotic vision algorithms,*" IEEE International Conference on Computer and Robot Vision, pp. 218-225, 2013.

[12] S. Borgsen, Z.M, M. Schopfer, L. Ziegler, and S. Wachsmuth, "*Automated Door Detection with a 3D-Sensor*," IEEE Canadian Conference on Computer and Robot Vision, pp. 276-282, 2014.

[13] D. Dai, G. Jiang, J. Xin, X. Gao, L. Cui, Y. Ou, and G. Fu, "*Detecting, locating and crossing a door for a wide indoor surveillance robot*," IEEE International Conference on Robotics and Biomimetics, pp. 1740-1746, 2013.

[14] M. Vlaminck, L. Jovanov, P. V. Hese, B. Goossens, W. Philips, and A. Pizurica, "*Obstacle detection for pedestrians with a visual impairment based on 3D imaging*," IEEE International Conference on 3D Imaging, pp. 1-7, 2013.

[15] M. Afif, R. ayachi, Y. Said, E. Pissaloux, and M. Atri, "*Recognizing signs and doors for Indoor Wayfinding for Blind and Visually Impaired Persons*," 2020 5th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP), 2020.

[16] R. Trabelsi, I. Jabri, F. Melgani, F. Smach, N. Conci, and A. Bouallegue, "*Indoor object recognition in RGBD images with complex-valued neural networks for visually-impaired people,*" Neurocomputing, vol. 330, pp. 94–103, 2019.

[17] J. Canny, "*A computational approach to edge detection*," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-8, no.6, pp. 679-698, 1986.

[18] X. C. He, and N. H. Yung, "*Corner detector based on global and local curvature properties*," Optical Engineering, vol. 47, no.5, pp. 057008, 2008

Mohammed Habib received the B.E. and M.E. degrees, from Suez Canal Univ. in 2000 and 2004, respectively. He received the Dr. Eng. degree from Suze Canal Univ. in 2009. After working as a research assistant (from 2000), an assistant professor (from 2009) in the Dept. of Electrical Engineering, Suze Canal Univ., His research interest includes Artificial Intelligence, Machine Learning, Deep Learning, Image Processing and Computer vision.