Real-time Door Detection for Indoor Autonomous Vehicle

Zhihao He* and Ming Zhu

Department of Automation, University of Science and Technology of China, Hefei, China

ABSTRACT

Indoor Autonomous Vehicle(IAV) is used in many indoor scenes. Such as hotels and hospitals. Door detection is a key issue to guide the IAV into rooms. In this paper, we consider door detection in the use of indoor navigation of IAV. Since real-time properties are important for real-world IAV, the detection algorithm must be fast enough. Most monocular-camera based door detection model need a perfect detection of the four line segments of the door or the four corners. But in many situations, line segments could be extended or cut off. And there could be many false detected corners. And few of them can distinguish doors from door-like objects with door-like shape effectively. We proposed a 2-D vision model of the door that is made up of line segments. The number of parts detected is used to determine the possibility of a door. Our algorithm is tested on a database of doors.1 The robustness and real-time are verified. The precision is 89.4%. Average time consumed for processing a 640x320 figure is 44.73ms.

Keywords: LSD; Door Detection; Real-time; Indoor Autonomous Vehicle

1. INTRODUCTION

Indoor Autonomous Vehicle is quite useful to decrease the labor cost. They have been used in some places successfully, such as hospitals, hotels, airports, factories and so on. But their tasks are limited to some simple and specified situations. There is much more work need to be done to make IAV more useful and intelligent.

Indoor space is divided into several zones by the wall. Zones are connected with the door. If IAV wants to pass through a door, it needs a robust and real-time door detection system.

An indoor autonomous vehicle can be divided into two parts: location part and obstacle detection part. This paper will focus on problems in location part. Many works have been done on this part, like SLAM.[1] There are even toolkits for it.[2] But SLAM is a global location system, it can only tell IAV where door is nearby with a certain degree of accuracy. IAV needs to detect doors and track them when going through the door. Doors are also valuable landmarks in navigation. We can use them to calibrate the self-location system and SLAM. Because doors are usually not movable.

In some complex but common scenarios, many things are similar to doors. Such as the fire cabinet in Fig.1a. It is easy to classify the fire cabinet as a door. Although there have been many researches about door detection, it still needs more in-depth researches. The illumination variation caused by light tube is also a serious problem we should solve.

2. RELATED WORKS

The 3D laser scanner can provide depth information of a scene. That is useful for detection of a door. Especially when the door is open, or the door is recessed into the wall. 3D sensor is used in some papers to detect door. Goron et al.[3] suppose that the door is recessed into the wall. Point clouds from an LRF are used to extract the planes of closed door by applying RANSAC. But many doors are in the same plane with the wall. DaWei Dai et al.[4] used an RGB-D camera to obtain the depth image of a opened door. Besides detecting the door, the relative position between door and robot is calculated to navigate the robot through the door. B. Quintana et al.[5] use 3D laser scanner and a color camera to get geometric (i.e. XYZ coordinates) and color (i.e. RGB/HSV) information. And they developed a 6D-space framework to detect door.

*him@mail.ustc.edu.cn

Ninth International Conference on Digital Image Processing (ICDIP 2017), edited by Charles M. Falco, Xudong Jiang
Proc. of SPIE Vol. 10420, 104200V · © 2017 SPIE · CCC code: 0277-786X/17/$18 · doi: 10.1117/12.2281651

But 3D-methods is high-cost and complex. Other methods using monocular camera have succeeded in some scenarios. Xiaodong Yang[6] and Yingli Tian[7] proposed a four corner plus 4 edge line model of door. That model gets a high precision in some common doors. But it will predict the fire cabinet in figure 1a as a door. This model needs at least two corners of a door to be detected which is susceptible to the light interference and image blur. And the speed of the algorithm is not shown. Rafiq Sekkal et al.[8] built a door model in corridor scenery. Floor/wall boundaries and vanishing points are estimated first. The relation between vanishing point and two vertical lines are used to classify door. But boundaries and vanishing point could not be detected accurately in many situations. And Floor reflection will make the vertical lines of door extended. Alexander Andreopoulos et al.[9] used 1500 instances of the door handle to train a detector in HSV color space. Wei Chen et al.[10] trained a convolutional neural network to detect door. That model has a low error rate in their dataset. But since the images in the dataset are similar, it is not sure whether that model will still work in a different environment. F.Mahmood et al.[11] employs feature based classification and uses the Kohonen Self-Organizing Map (SOM) for the purpose of door detection. However, this method relies on the vertical gap between the door and the floor. So the camera should be fixed at a low position and it couldn't detect opened doors.

Algorithms that use color, texture or door knob features are limited to environments. We improve that model of Xiaodong Yang[6] by adding the skirting line to it. And door candidacy is generated by line segments. We use LSD algorithm that is fast to detect line segments.[12] This improvement makes the system less susceptible to illumination changes, image noise, and floor reflection. And it is fast enough for a real-time work.

3. PROPOSED METHOD FOR DOOR DETECTION

In this paper, a model of the door is proposed in order to detect doors in different indoor environments. Door-like polygons are the key part of this model. And we proposed a scoring algorithm to assess the certainty of a door. Scores of doors are used to filter overlapping doors and false detected doors.

3.1 Geometric Door Model

The geometric model of the door in our algorithm consists of 6 line segments. As showed in figure 1b, there are three key line segments: the left vertical line(LV), the right vertical line(RV) and the head line(HL) showed in blue, green and red respectively in Fig. 1b. They are the parts most hardly changeable. We use the three key line segments to search door candidates. The three key line segments form a quadrilateral containing the door.

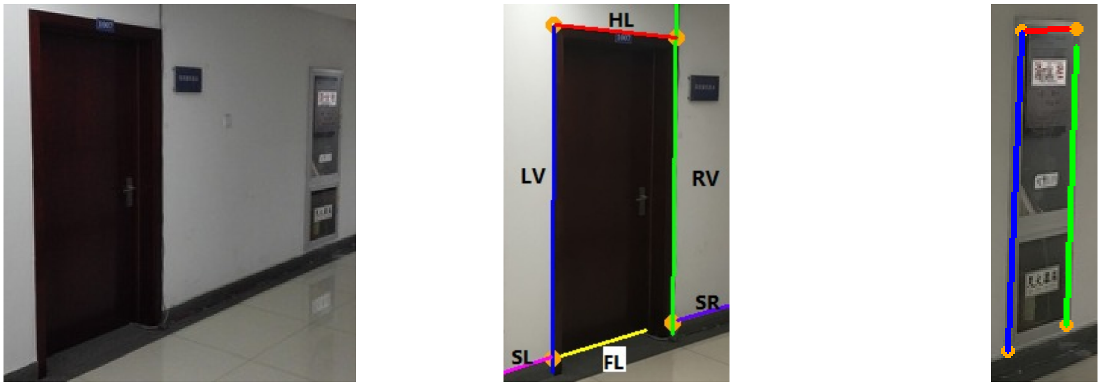
This quadrilateral model or four corner model is used in many methods[6,7].[13] But many other things attached to the wall will get same results under such model. As showed in figure 1c. The fire cabinet fixed in the wall has the same representation in such model. So we added the skirting line part to our model. The left skirting line(SL) and the right skirting(SR) are showed in figure 1b. Skirting line is split by a door. So there are two skirting line segments connected to a door under ideal condition. One is closed to the bottom of the left vertical line segment of the door. One is closed to the right. Something rectangle attached to door doesn't have such feature.

Foot line will not be detected for many opened door. But many false detections of door will contain a foot line segment part as showed in Fig. 1c. So foot line segment is not included in our model.

These are the assumptions for this model:

1. The vertical line of a door is nearly perpendicular to the horizontal axis of the image.
2. The head line segment of Door is visible in an image.
3. There are skirting lines connected to a door.
4. The sizes of the doors in the same image will not differ much.
5. The two vertical lines of the door are parallel.

It is easy to achieve these assumptions in indoor circumstance. Since it is a general model for an interior door, it wouldn't be over-fitted for a particular dataset. The method is robust to handle many situations. Such as the light changes caused by a light tube. The reflection of the floor. The door is opened or closed. Something rectangular on the wall, such as a poster, fire cabinet and paintings will not cause a false detection.



(a) Original Image (b) Standard Door Model (c) Things Similar to Doors

Figure 1. Door Model

3.2 Image Pre-processing

Out method is based on the detection of line segments. The detection result of the method will influence post steps much. Before the detection, we use a 5×5 Gaussian low-pass filters to smooth the image. This pre-process reduce the line segment fracture in LSD detection. Because the LSD is based on the local gradient of image to detect line segments. And the pre-process reduces the line segments from floor reflection and noise.

3.3 Line Segments Detection and Merger

Line segments detection is a key step in our method. In order to make this method work in real-time, we need to detect line segments fast. We use the LSD algorithm[12] to detect line segments. The line segments are divided into the vertical and the non-vertical. Since it is based on local gradient orientations, the algorithm is fast. The main issue is the cut of one line when the gradient orientation is changed. So we merge the collinear line segments whose endpoints are close to each other. In order to reduce the time-consuming of the merger process, only the vertical line segments are merged.

3.4 Door Candidates Grouping

The three key line segments of a door are showed in figure 1b– LV, RV and HL. If top points of two vertical line segments are connected by a line segment, it will be a door candidacy. The framework of the algorithm is shown in algorithm 1.

Algorithm 1 Door Candidates Search

Input: Vertical Line Segemnts: L_v, Non-vertical Line Segments:L_c
Output: Door Candidates: S_{door}

sort L_v by x coordinate of top endpoint
for l_v^i in L_v **do**
  set l_v^i as the left vertical line of door candidacy.
  get vertical lines near $l_v^i \rightarrow L_{neib}^i$
  search right vertical line segments in $L_{neib}^i \rightarrow L_{candi}^i$
  for $l_{candi}^{ij} \in L_{candi}^i$ **do**
    if there is a head line connect l_v^i and l_{candi}^{ij} **then**
      add this door candidacy to S_{door}
    end if
  end for
end for

There are two main steps in algorithm 1:

- search the vertical line segments of a door in line 3∼ 5.
- search the head line segment of a door in line 7

Only if a pair of vertical line segments and a head line segments connects them are founded, a door candidacy could be added. The two main steps are described in detail in following sections.

3.4.1 search two vertical line segments

Define D_{ij} as the distance between the left vertical line segment and the right. In order to decrease the computation complexity, D_{ij} is limited to $[0.05W_{im}, 0.8W_{im}]$, in which W_{im} denotes the width of the image.

In the step of finding right vertical line segment on line 5 of algorithm 1. There are 5 requirements. Only if the pair of vertical line segments meet all the requirements, it will be processed further. Let $\vec{lv}$ denotes the direction vector of the left vertical line segment. Let $\vec{rv}$ denotes the direction vector of the right vertical line segment. θ_{lr} denotes the angle between $\vec{lv}$ and $\vec{rv}$. Let len_l and len_r denote lengths of the left and right line segment.

- The vertical line segments should be parallel:

$$\theta_{lr} < ParallelThres$$

- The length of both lines should be long enough:

$$len_l > LengthThres$$

$$len_r > LengthThres$$

- The top endpoints of vertical line segments should be in the up half part of the image.
- The bottom endpoints of vertical line segments should be in the bottom half part of image.
- The length of both vertical line segments shouldn't change much.

3.4.2 search head line segment

A perfect head line is showed in figure 2a. Searching such head line segments of a door is easy. Firstly, find the line segments whose left endpoint is closed to the top endpoint of the left vertical line segment. And then, check if the right endpoint is closed to the top endpoint of the right vertical line segment of door.

But some head line segments could be cut off, divided into parts or interfered. The head line segment showed in Fig.2b doesn't connect the top endpoints of right vertical line segments. Because the right vertical line segment extends too much. The head line segment showed in Fig.2c is cut off by doorplate. The head line segment showed in Fig.2d is interfered by a mechanical component mounted on the door.

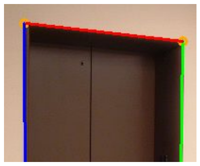
To test if there is a head line for a door candidacy, we use the fill-ratio[6] to evaluate each head line candidacy. We adapt the algorithm to make it more robust. The main steps is showed in algorithm 2.

Firstly, a rectangle search range is generated according to the top points of two vertical line segments. As showed in Fig. 3a. Secondly, we search line segments in this range. Thirdly, the line segments are extended to intersect with the two vertical lines of the door. This extended line segment is the correspond possible head line. The intersection result is showed in Fig. 3b. The likelihood of this extended line segment being a head line depends on the fill-ratio of this line segments.

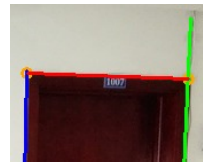
Fill-ratio is defined as follow:

$$FR_{ij} = \frac{ed_{ij}}{len_{ij}}$$

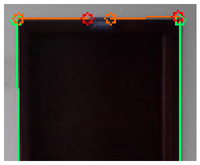
In which, ed_{ij} is the number of edge points in this line segment. len_{ij} is the length of the line segment. But the extended line segment could only cover part of a the head line edge. As showed in Fig. 4. So in line 9 of algorithm 2, if the points above or below pxl_i is an edge pixel, the edge point count is increased. In our algorithm, the threshold of FR is 0.6.



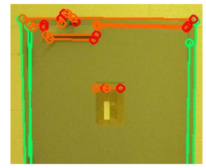
(a) Perfect Head Line Segment of Door



(b) vertical line is extended

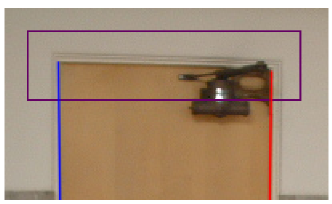


(c) head line is cut off



(d) head line is interfered

Figure 2. Different Head Line



(a) Door Head Line Search Area

(b) extended line segment

Figure 3. Search Head Line Segment of a Possible Door

Algorithm 2 Calculate the fill ratio of a Suspect Line Segment

Input: A line segment:ls_i, Vertical Line Segments of A possible Door: LV_i, RV_i, edge image img_e
Output: Fill Ratio of this line segment

calculate the Line where the ls_i is located $\rightarrow l_i$
calculate the intersection point of l_i and $LV_i \rightarrow p_l$
calculate the intersection point of l_i and $RV_i \rightarrow p_r$
get the points of line segment$(p_l - p_r) \rightarrow PT$
init edge point counts $\rightarrow count = 0$
for point pt_i in PT **do**
  get the pixel in pt_i of $img_e \rightarrow pxl_i$
  get the pixels near pt_i in y coordinate $\rightarrow pxl^i_{neib}$.
  if pxl_i is edge point or pixels in pxl^i_{neib} is edge point **then**
    count ++
  end if
  get the fill ratio=$count$ / length of line segment $(p_l - p_r)$
end for

Figure 4. Possible Bias of a Head line Segment

3.5 Find Skirting Line of Door

We search the skirting line segments which are near bottom endpoints of two vertical line segments. Firstly, a rectangle search range is generated according to the bottom endpoints of vertical line segments. Secondly, if the right endpoint of a non-vertical line segment is in this range, it is added to S^l_{sk}. if the left endpoint of a non-vertical line segment is in this range, it is added to S^r_{sk}. Finally, the line segment in S^l_{sk} that is closest to the bottom end point of LV is selected as left skirting line. The one closest to the bottom endpoint of RV in S^r_{sk} is selected as right skirting line.

3.6 Scoring and Filter door

In some complex situations, there could be many "door" detected around a real door. As showed in Fig. 5a. So we build a scoring system to evaluate the certainty of a "door". The scoring rules are listed as below.

- The basic score is s_b. It represents the certainty of the basic part (left vertical line , right vertical line and head line). It is defined as follow.

$$s_b = S_b \cdot \Delta \cdot \lambda$$
$$\Delta = \frac{\max(d_l, d_r)}{w}$$
$$\lambda = \cos(\angle(\vec{lv}, \vec{rv}))$$

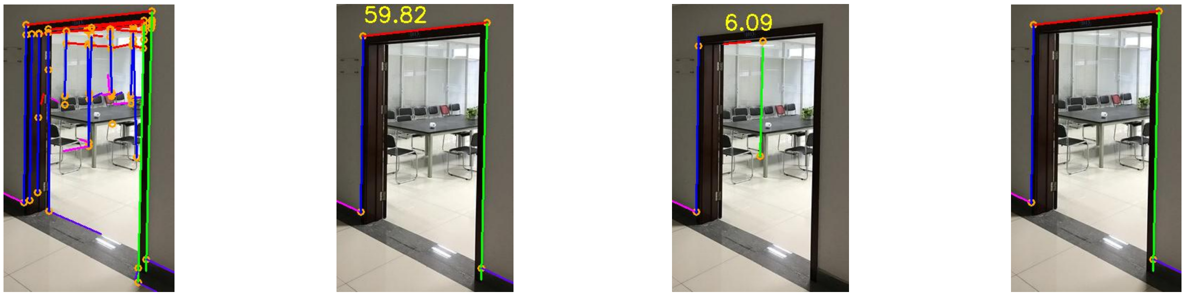
  in which, S_b is the score for a perfect basic part. d_l is the distance between left endpoint of head line segment and the left-top point of the door. The left-top point of door is the intersection between head line and left vertical line of door. d_r is the distance between right end point of head line segment and the right-top point of door. The right-top point of the door is the intersection between head line and right vertical line of the door. w is the distance of two vertical lines. $\vec{lv}$ is the direction vector of the left line segment of the door. $\vec{rv}$ is the direction vector of the right line segment of the door. λ represents the parallelism between the left and right vertical line segment of door. Δ represents the certainty of up part of "door".

- Extra score: s_e. It is depended on the skirting line of a door. Left and right skirting line is represented by l^l_{sk} and l^r_{sk} respectively.

$$s_e = \begin{cases} S_e & \text{if} \quad l^l_{sk} \textbf{ and } l^r_{sk} \textbf{ is collinear} \\ 0 & \text{else} \end{cases} \tag{1}$$

The score of a real door is showed in Fig. 5b. The fake door overlaps with it gets a low score as showed in Fig. 5c. After scoring every door candidate, we can filter the doors by score. If two doors overlap, the door with lower score will be eliminated. The filtered result is showed in Fig. 5d

According to the 4-th assumption of the proposed door model in Section 3.1. The size of door candidate can be used to filter fake door. Firstly, the door with the highest score is selected. Noted as DR_m. Secondly, for other doors, if one's height is less than $r\times$ height of DR_m, it is eliminated. r is the threshold ratio. In our experiment, it is set to 0.6.



(a) Doors Covered (b) high score for a real-door (c) low score of a fake "door" (d) After filtering

Figure 5. Filter Overlapping Door

4. EXPERIMENTAL RESULTS

To evaluate this algorithm, we collected 142 figures of indoor environments. Each one contains at least one door. Including some samples from Xiaodong Yang's paper[6] . Since images in their dataset are captured via a miniature camera mounted on the head via a cap or sunglasses. Only part of them are the same to a view of an indoor autonomous car. We used C++ to implement this algorithm using basic APIs from OpenCV.[14] The experiments are performed on an Intel Core i3-3220 3.30GHz CPU with 10 GB RAM. The precision is 89.4%. The average time consumed for processing a 640x320 figure is 44.73ms. Fig. 6 shows examples of detected doors.

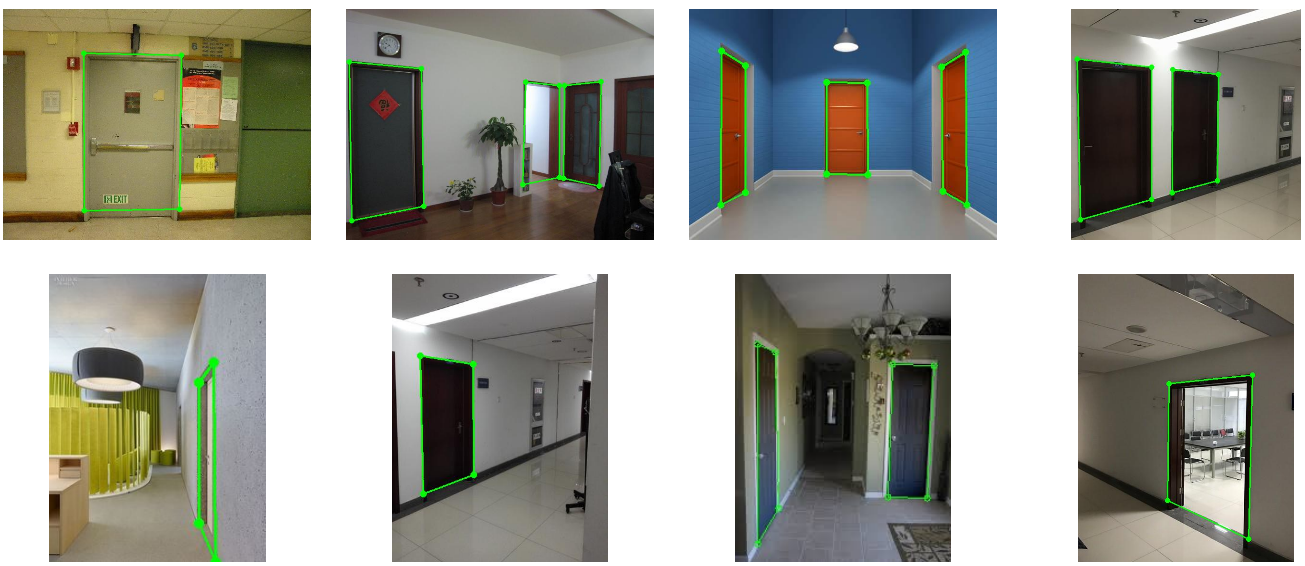


Figure 6. Examples of successfully detected doors in different environments.

5. CONCLUSION AND FUTURE WORK

In this paper, we have presented a door detection algorithm of computer vision. The geometric door model consisting of 3 key line segments and 2 extra line segments. With the help of LSD, a fast line segment detection algorithm, the search space decreased sharply. And the detection algorithm is fast enough for an indoor autonomous vehicle. The relationships between doors are considered to eliminate some false detection. Unlike existing models of the door needs a perfect detection of line segments or corners of the door, our method can overcome problems of line is cut off and corner missing. We have tested it in a dataset containing different indoor scenar. The precision is 89.4%.

The future work will focus on getting the depth information of environment using dual-camera. With more generic features, we will build a deep-learning system to understand the indoor environment. Not only door will be detected, bookshelves, but also cabinets and furnitures can be detected.

ACKNOWLEDGMENTS

This work is supported by Chinese Academy of Sciences(CAS) under Grant Nos. 2017ZX03001019-004

REFERENCES

[1] Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., Reid, I., and Leonard, J. J., "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Transactions on Robotics* **32**(6), 1309–1332 (2016).

[2] "Mobile robot programming toolkit."

[3] Goron, L. C., Tamas, L., and Lazea, G., "Classification within indoor environments using 3d perception," in [*Automation Quality and Testing Robotics (AQTR), 2012 IEEE International Conference on*], 400–405, IEEE (2012).

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

[5] Quintana, B., Prieto, S., Adán, A., and Bosché, F., "Door detection in 3d colored laser scans for autonomous indoor navigation," in [*Indoor Positioning and Indoor Navigation (IPIN), 2016 International Conference on*], 1–8, IEEE (2016).

[6] Yang, X. and Tian, Y., "Robust door detection in unfamiliar environments by combining edge and corner features," 57–64 (2010).

[7] Tian, Y., Yang, X., and Arditi, A., "Computer vision-based door detection for accessibility of unfamiliar environments to blind persons," in [*International Conference on Computers for Handicapped Persons*], 263–270, Springer (2010).

[8] Sekkal, R., Babel, M., Brun, B., Leplumey, I., et al., "Simple monocular door detection and tracking," in [*2013 IEEE International Conference on Image Processing*], 3929–3933, IEEE (2013).

[9] Andreopoulos, J. A. and Tsotsos, J. K., "A framework for door localization and door opening using a robotic wheelchair for people living with mobility impairments," in [*Robotics: Science and Systems, Workshop: Robot Manipulation: Sensing and Adapting to the Real World, Atlanta*], (2007).

[10] Chen, W., Qu, T., Zhou, Y., Weng, K., Wang, G., and Fu, G., "Door recognition and deep learning algorithm for visual based robot navigation," in [*Robotics and Biomimetics (ROBIO), 2014 IEEE International Conference on*], 1793–1798, IEEE (2014).

[11] Mahmood, F. and Kunwar, F., "A self-organizing neural scheme for door detection in different environments," *arXiv preprint arXiv:1301.0432* (2013).

[12] von Gioi, R. G., Jakubowicz, J., Morel, J.-M., and Randall, G., "Lsd: a line segment detector," *Image Processing On Line* **2**, 35–55 (2012).

[13] Shalaby, M. M., Mohammed, A., Salem, M., Khamis, A., and Melgani, F., "Geometric model for vision-based door detection," in [*Computer Engineering & Systems (ICCES), 2014 9th International Conference on*], 41–46, IEEE (2014).

[14] Bradski, G. *Dr. Dobb's Journal of Software Tools* .