Robust Door Detection in Static Images

Zhigang Zhu, Chikako Olsen, Ivan Miller

Abstract

Ability to identify doors provides important information about the environment that is crucial to a task of navigation in both indoor and outdoor settings, which is extremely important in our day-to-day life. Door detection could seem like a trivial task for a regular person, however, it may be extremely difficult for visually impaired people. Creating an algorithm capable of generalization could also be quite challenging due to the vast number of geometric shapes, designs, attributes, and, oftentimes, nondescript features of doors.

In this project we focus only on computer vision techniques and investigate different methods to extract meaningful elements of a static image such as lines and corners. We then introduce several sets of logical rules to analyze different combinations of those elements and identify image segments that could belong to a door frame. The experimental results show that the proposed approach successfully detects typical doors in indoor and outdoor environments and reaches 67.42% accuracy when tested on an input set of 132 images.

1. Introduction

Information about the position of a doorframe is extremely important in our day to day life and most of us probably take our ability to obtain it for granted. In indoor navigation doors help determine information about localization and route planning, which makes them one of the most important types of location markers of the indoor environment.

Navigating safely in an unknown environment (both indoor and outdoor), is one of the major challenges for blind and visually impaired people. The “World report on vision” [1] conducted by the World Health Organization in 2019 estimated that at least 2.2 billion people globally had vision impairment or blindness. The staggering amount of people affected by this issue makes it extremely relevant and important. That is why an algorithm that is capable of reliably detecting doors could become a crucial element of assistive technologies that give visually impaired people an opportunity to successfully navigate in indoor or outdoor environments. In 2022 Apple introduced door detection functionality to their iPhone and iPad devices through a built-in Magnifier app that supports blind and low vision users. The app takes the data from LIDAR and camera, and uses machine learning models to locate the door in real time. It is also capable of describing attributes of a door, such as whether it is open or closed, and “when it’s closed, whether it can be opened by pushing, turning a knob, or pulling a handle” [2]. The recognition of the importance of the task of door detection for assistive technologies gave additional reinforcement to our motivation to work on this project.

Additionally, reliable detection of doors in a scene has multiple applications in robotics, where many door detecting techniques take advantage of data beyond just images from a camera such as data from distance, range, localization and other sensors. Quite often a combination of computer vision and machine learning techniques is being used to achieve higher detection rates and improve the speed of an algorithm.

When approaching this project, we decided to specifically focus only on computer vision techniques leaving machine learning applications outside the scope of the project. One of the reasons behind that decision was the fact that such an approach could further improve the systems with limited hardware that only rely on visual information, for example, systems lacking 3D sensors or computational power necessary to run a machine learning model in the background. We also limited the scope of this project to the implementation of the proposed algorithm in MATLAB leaving for future work any possible integrations with other models that could use the outputs of our algorithm as input with a goal of building a more robust classifier and improving detection rates.

Despite the task being seemingly simple, detecting a door in an image with a reasonable degree of certainty could be quite challenging. Even though doors are very common in different environments, the diversity of shapes and sizes, changes in illumination and viewpoints, and, oftentimes, nondescript features of doors make generalization with pure computer vision techniques more difficult. Our method is based on a sequence of computer vision methods applied to a static input image such as Sobel filter for detection of edges, Hough transform for identification of lines that match certain criteria, and Harris-Stephens algorithm that helps detecting corners. Then we define several sets of logical rules to analyze different combinations of elements detected in an image and select only those that could potentially belong to a door frame. As a result, in an event when a presence of a door was detected in a given image, our proposed algorithm outputs a graphic overlay filled with a colored outline around the perimeter of a door. Additionally, once a location of a door is identified, it could provide insight about 3D information. Since most of the doors have consistent geometric properties, we could use sides (edges) of the door to find locations of vanishing points and then use them to obtain additional information about 3D, for example, infer information about structure of the room. The experimental results showed that the proposed approach successfully detects typical doors in indoor and outdoor environments and achieves an average/balanced accuracy rate of 67.42%.

2. Related Work

Door detection is not a new topic in the research community and a variety of methods have already been developed. The existing approaches could be divided by the type of sensors involved in data collection: visual information only or a combination of visual information with the data from other types of sensors such as sonars. Either of those two large groups could then be further divided into several smaller logical groups based on the methods and applications utilized by the developers.

One group could be represented by rule-based methods. For example, in [3] the authors developed a fuzzy logic-based method searching through various segments of the image and measuring the relationship between them to detect the presence of defined fuzzy logic concepts and establish a membership degree of a given segment. The method requires at least three segments of a door frame to be seen in an image for successful detection which reduces the performance of an algorithm when tested on large and diverse datasets. In [4] the authors used boolean logic rules and predefined threshold values across various metrics to identify doors. It is also worth noting that a lot of research in robotics is being done around SLAM (simultaneous localization and mapping) algorithms [5], which rely on stereo vision and wide range of additional sensors (laser rangefinders, LIDARs, sonars etc) and focus on 3D geometry of the environment with a goal of simultaneously building a map of the environment and localizing the position of the agent in that map for the purpose of navigation.

In a second logical group of methods the authors combine computer vision applications with machine learning algorithms. Relevant features are first being extracted from images using computer vision algorithms in order to then be used as input to a machine learning model that computes the final prediction about the presence and location of a door in an image. For example, in [6] the authors adopted a probabilistic approach and used a model-based Bayes inference to make the decision about location of a door in an image. In [7] the authors proposed a vision-based door detection algorithm that combines several feature-based classifiers using AdaBoost (Adaptive Boosting) to produce a final prediction.

The third large subset of methods could be grouped together because of the use of deep learning with a goal of detecting doors. In order to do that the authors of [8] used transfer learning to train a Mobile-Net-based model. In [9] the developers used a combined output of three different types of models to not only detect doors but also differentiate between open, closed and semi-open doors in real time: 3D (PointNet) and 2D object classification models (AlexNet, GoogleNet), semantic segmentation (FastFCN, FC-HarDNet, SegNet and BiSeNet), and an object detection algorithm

(a) (b)

Fig.1 Results of the segmentation performed by the Segment Anything model (b) and successful identification of all doors in the input image (a).

(DetectNet). The last model that we wanted to mention is called Segment Anything (SA) and it was released last month by Meta. Even though it wasn't specifically designed for door detection, SA was trained in zero-shot setting on the largest segmentation dataset to date, which resulted in a very promising performance. In Figure 1 you could see the results of a segmentation task produced by SA on one of the images from our dataset where the model successfully identified all doors seen in the image.

Our proposed approach combines some of the aspects of the above mentioned research in an attempt to improve performance and ability of the model to generalize well. The methods we used are most similar to [3] and [4]. Additionally, once a location of a door is identified, it could provide insight about 3D information. Since most of the doors have consistent geometric properties, we could use sides (edges) of the door to find locations of vanishing points and then use them to obtain additional information about 3D (e.g. infer information about structure of the room).

3. Proposed Approach

The proposed methodology involves several preprocessing steps of an input image followed by a Sobel operator. The resulting output is further improved through dilation and subjected to a Hough transformation, resulting in a map highlighting lines within the image. Subsequently, the Harris-Stephens algorithm is utilized to detect corners. Lastly, heuristics that combine lines and corners into groups that could represent a door frame are employed as shown in Figure 2:

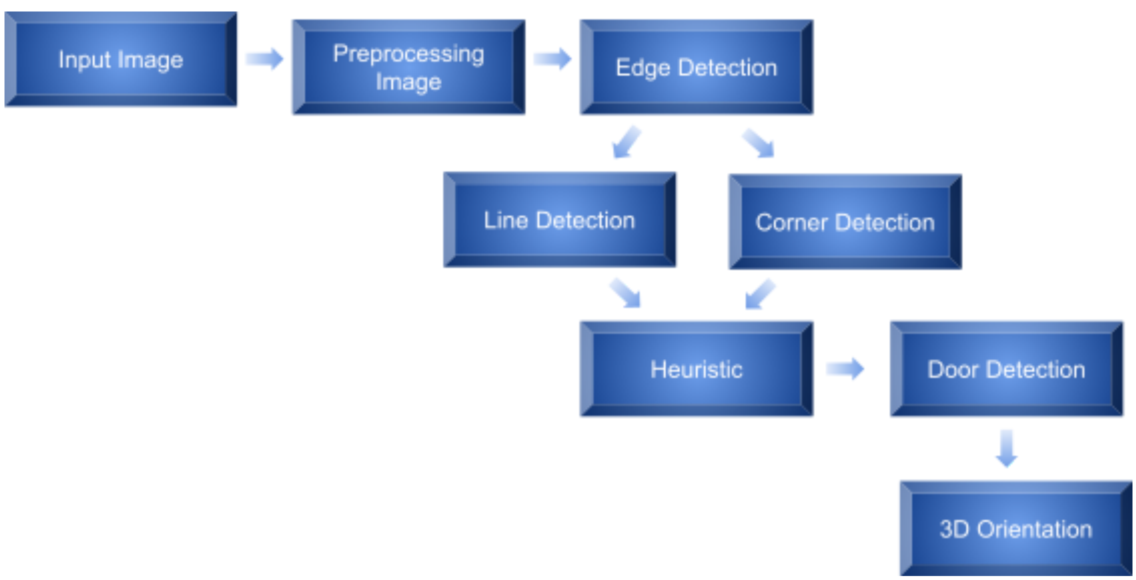


Fig.2: Sequential steps of the proposed door detection algorithm: 1) series of preprocessing filters; 2) Sobel filter for detection of edges; 3) Hough transform for identification of lines; 4) Harris-Stephens algorithm to detect corners. 5)Heuristics analyzing combinations of elements detected in an image and selecting only those that could potentially belong to a door frame. 6) Use sides (edges) of the door to find locations of vanishing points and then use them to obtain additional information about 3D orientation.

3.1 Image Preprocessing

Before transitioning to the edge and corner detection phase, preprocessing steps are applied to every input image. Initially, the image is converted to grayscale and subjected to a 2D median filter. This filter computes the median value within a 3-by-3 neighborhood around each pixel of the original image to generate a value for the output pixel. Following this, the intensity of the image is enhanced by remapping the data values to fill the entire available intensity range of [0,1]. Finally, a noise reduction process is conducted using a 5x5 Gaussian filter with a standard deviation of 2. These image transformations comprise a comprehensive pipeline that precedes the edge detection stage. Please refer to Figure 3 to see the sequential transformation of the input image as it travels through the preprocessing pipeline.

(a) (b) (c) (d) (e)

Fig.3: Image preprocessing pipeline - series of preprocessing steps which are employed to improve the results of the subsequent edge and corner detection from original image (a) through several filter transformations: (b) Grayscale; (c) 2D Median Filter; (d) Intensity increase (e) 5x5 Gaussian filter

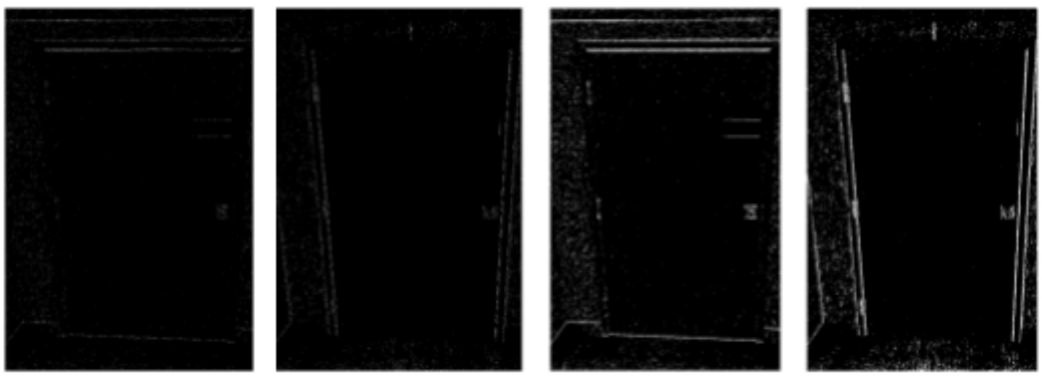
3.2 Edge Detection

After the application of the last image filter from preprocessing steps the resulting image is subjected to a pair of convolutional masks using 3x3 Sobel operators in both X and Y directions. These masks, represented by formulas (1) and (2), correspond to the Sobel filter for horizontal and vertical changes in pixel intensity, respectively. The purpose of these filters is to compute the gradient of the image intensity at each pixel, allowing to identify the direction and magnitude of the most significant intensity change.

$$\mathbf{G}_x = \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix} * \mathbf{A} \qquad (1)$$

$$\mathbf{G}_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * \mathbf{A} \qquad (2)$$

The resulting output gives insight into a level of abruptness or smoothness as it changes from one pixel to another throughout the image, indicating the likelihood of a given pixel representing an edge and providing information about its possible orientation. Figure 4 shows the results of convolving the image with the 3x3 Sobel kernels and illustrates horizontal (a) and vertical (b) edges detected by the filter. Following the detection of edges with the Sobel filter, a dilation operation is performed to enhance the prominence of these edges for the subsequent line detection algorithm. A disk-shaped morphological structuring element with a radius of 2 was utilized for the dilation operation.

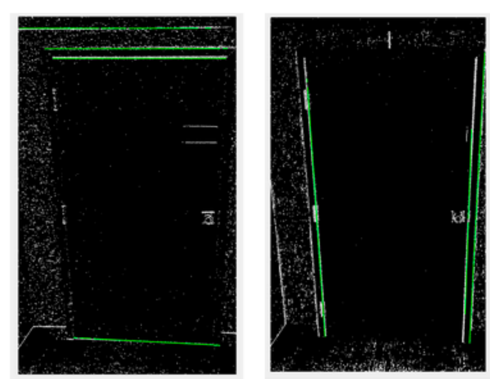


(a) (b) (c) (d)

Fig.4: Detection of the most prominent horizontal (a) and vertical (b) edges in the image using 3x3 Sobel operators by convolving them with the original image. Next, a dilation with a disk-shaped morphological structuring element is performed on the horizontal (c) and vertical (d) edges to make them more prominent for the line detection algorithm.

3.3 Hough Transform

The edges detected by the Sobel filters and enhanced by dilation form an edge map. Standard Hough transformation is then being performed over horizontal and vertical edge maps separately: every pixel of an edge identified by polar coordinates is added to an accumulator array that forms a grid. The algorithm goes through all cells in that grid and if a given cell exceeds a certain threshold a line is being detected. The transformation outputs a list of lines specified in polar coordinates, which in our implementation was limited to 100 lines (by specifying the max number of peaks in the Hough transform matrix) for each horizontal (See Fig.5a) and vertical edge map (Fig.5b), 200 lines in total.



(a) (b)

Fig.5: Horizontal (a) and vertical (b) lines detected with a Hough transformation

3.4 Corner Detection

Before proceeding to heuristics, the last step in extracting features and elements of the image that could potentially represent a door frame, utilizes Harris-Stephens algorithm to identify the most robust corners in the input image (See corner points indicated by green x marks in Fig.6). Through trial and error we determined that a somewhat conservative threshold of selecting up to 1,000 corner points sorted in the descending order of strength allowed for an optimal balance between having a feasible number of potential corner points to work with (and to eventually eliminate most them) and unnecessarily limiting the output of the algorithm and potentially missing points that could be crucial for the reconstruction of a door frame.

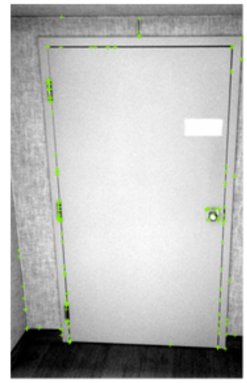


Fig.6: Output of the Harris-Stephens algorithm detecting up to 1,000 strongest corners in the input image indicated by green x marks

3.5 Algorithm for Door Detection

Once all relevant features and elements of the input image have been successfully extracted we begin the process of elimination of those segments that for any reason could not represent a door frame. Fig. 7 shows a schematic representation of logical steps for the proposed door detection algorithm:

Steps for the proposed door detection algorithm

1. We begin by identifying the corners associated with detected line segments.
2. We then proceed to cluster those corners together.
3. By drawing lines between the clusters of corners, we identify potential candidates representing the vertical lines of potential doors.
4. We then calculate the length of diagonals and height-to-width ratio for pairs of lines.
5. In an event when the remaining image segments match all available conditions, a presence of a door is being detected in the input image and the algorithm outputs a polygon filled with a colored outline around the perimeter of a door displayed as an overlay over the input image.

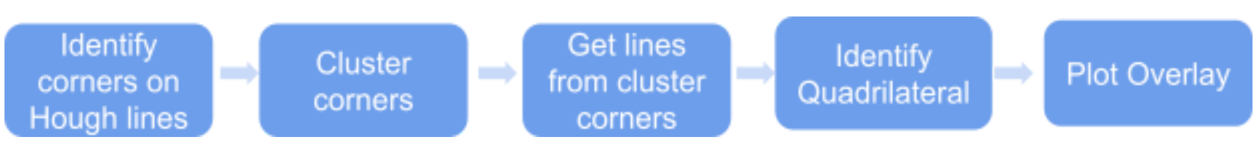


Fig.7: Schema representing the logical steps for the proposed door detection algorithm.

3.5.1 Corners on Line segmentation

As a first step we identify corners that belong to horizontal (see Fig.8a) and vertical (Fig.8b) lines obtained from Hough transformation. To achieve this, we conducted a series of experiments to establish a threshold of 20px, which allows us to consider corners within a defined distance that is small enough from a given line to be potentially located on the line itself and treated as such. The distance between a point (with coordinates (x0, y0)) and a line (defined by its start and end points (x1, y1) and (x2, y2)) is computed using formula (3):

$$\frac{\left|(x_2 - x_1)(y_1 - y_0) - (x_1 - x_0)(y_2 - y_1)\right|}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}} \tag{3}$$

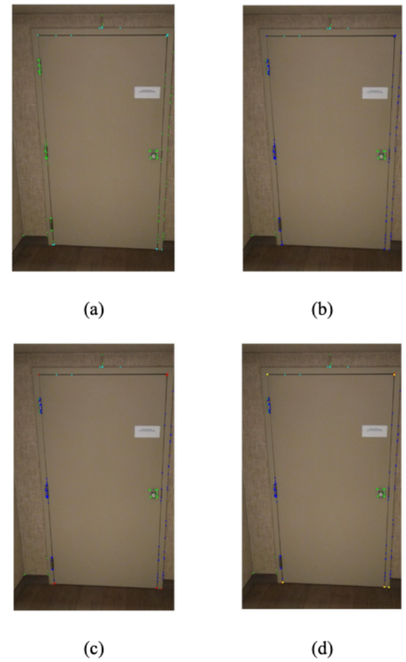


Fig.8: Corner grouping and elimination: (a) Corners that belong to horizontal lines colored in cyan; (b) corners on vertical lines - blue; (c) corners that belong to both horizontal and vertical lines - red; (d) Clusters of corners that belong to both horizontal and vertical lines and located with a threshold distance to each other (25px)

All identified corners associated with the Hough lines from the Hough transform are presented using distinct colors in this representation. For instance, the corners highlighted in cyan in Fig.8a correspond to corners located on or near horizontal lines. Similarly, the blue stars in Fig.8c represent corners belonging to vertical lines, while the red stars represent corners that are present on both horizontal and vertical lines.

3.5.2 Cluster corners

Quite often multiple corners are being detected on both vertical and horizontal lines and may also be close to one another. We employ a minimal distance threshold of 25px to group such corners together. The distance between corners is calculated using formula (4), and in Fig.8d the resulting corners meeting this criterion are denoted as bold yellow dots. These corners are regarded as the most robust candidates for the algorithm to proceed with.

$$D = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{4}$$

3.5.3 Best lines from corners

In some cases, the Hough lines obtained from the Hough transform may need to be longer to represent the entire edge line of the door accurately. Consequently, it becomes necessary to redraw vertical and horizontal lines by connecting corners from different clusters. We assume that the angle of the vertical line falls within the range of 85 to 95 degrees while the angle of the horizontal line is less than 45 degrees. The angles are calculated using formula (5) and (6). Additionally, if a line is shorter than 25 pixels, it is disregarded. At this stage, we consider the candidate lines as the potential sides of the quadrilateral, as illustrated in Fig.9.

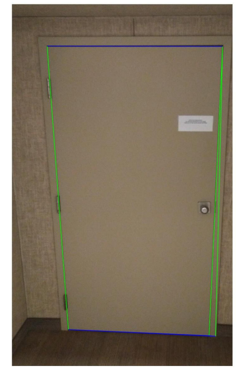


Fig.9: Vertical and Horizontal lines based on clusters of corners

$$Slope = \frac{(y_2 - y_1)}{(x_2 - x_1)} \qquad (5)$$

$$Angle = \left|tan^{-1}(slope)\right| \qquad (6)$$

3.5.4 Quadrilateral Check

We evaluate and identify the pair of vertical lines that form the most optimal quadrilateral shape among the candidate lines. To accomplish this, we establish specific criteria:

- The height should be more than 1.5x the width.
- The diagonal lines should be approximately equal, with a difference of 25px.
- The two diagonal lines from a vertical line pair should have equal angles but opposite signs.

By applying these conditions, we can eliminate the unsuitable vertical line pairs, thereby obtaining the four corners that define the quadrilateral shape. Based on the outcome of four corners, we visualize the results by plotting overlay using the poly shape methods.

3.6 Vanishing Points

As additional steps of this project, we incorporate the construction of a 3D orientation based on vanishing points. Once doors are detected in the image, we extend their edge lines to obtain vanishing points. We acquire two vanishing points

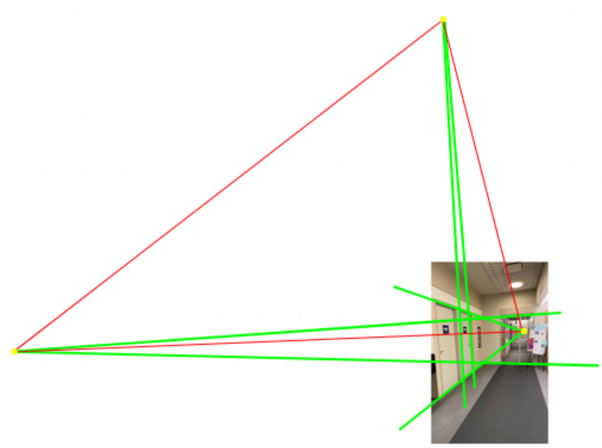


Fig.10: Example of three Vanishing Points

if the algorithm identifies only one door in the picture. In such cases, we utilize the Hough lines to determine the third vanishing point required for applying the Orthocenter Theorem. Following the estimation of the image center, we proceed to calibrate the intrinsic and extrinsic parameters of the 3D coordinates. This calibration is achieved by employing the Homogeneous system and Singular Value Decomposition (SVD) methods.

4. Results and Analysis

In order to test the algorithm's ability to generalize, it was tested on various types of doors as they appear in static images taken from different camera angles with varying degrees of perspective distortion. The system was implemented in a MATLAB environment running on MacBook Air M1, 16Gb RAM.

4.1 Simple Rectangular Door

By employing our algorithms, we successfully detected rectangular-shaped doors nearly parallel to the image frame, given that the four cluster corners of the doors are detected in step 3.5.2. An example result is illustrated in Fig.11. This detection process was generally more effective when the door was in a closed position.

(a) (b) (c) (d)

Fig.11: Results of the application of the door detection algorithm to an image of a simple rectangular door with minimal perspective distortions: original images (a,c); Original image; polygon filled with a colored outline around the perimeter of a detected door displayed as an overlay over the input image(b, d).

4.2 Angled door

The proposed door detection algorithm has successfully detected angled doors, even though they also identified additional shapes, as evident in Fig.12. The most challenging aspect of detecting angled doors is identifying corners on vertical and horizontal Hough lines. However, as long as we can detect four cluster corners of the door, the detection remains effective regardless of its angle.

(a) (b) (c) (d)

Fig.12: Angled door results: original images (a, c); polygon filled with a colored outline around the perimeter of a detected door displayed as an overlay over the input image(b, d).

4.3 Multiple doors

Observing Fig.12d, it is evident that the proposed algorithm has successfully detected multiple doors. Once again, detecting corners on both Hough lines remains crucial, enabling the identification of all doors within the image.

4.4 Opened doors

Detecting open doors presented a challenge due to detecting line segments, specifically horizontal lines, from the Hough transform. While there have been instances of the proposed algorithm being capable of identifying opened doors, in most cases the successful detection of the door was likely happening due to the similarity in color between the background behind the opened and the door itself (see Fig.12b where both the color of the door of the closet in its insides are the same). However, the algorithm still struggled to detect the folded part of the closet door in the same image.

4.5 Occluded doors

For our algorithm to function effectively, it relies on detecting all four corners of the door. Consequently, occluded doors or doors where one or more corners are not visible will not yield satisfactory results when processed through our algorithm.

4.6 Other type of doors

Our algorithm is not designed to handle doors with shapes other than rectangles, such as those with rounded tops, as it relies on detecting four corners to identify the door. Additionally, when the door is made of glass, like a French door that allows the background to be seen through, our algorithm faces limitations, as demonstrated by the glass door in the hallway shown in Fig.12d. Moreover, the door's texture plays a significant role, as doors with intricate or rough surfaces may detect numerous corners unrelated to the door frame.

5. Conclusions

In this paper we presented an algorithm for door detection in static images. Our approach to the problem is based on the analysis of segments of the image such as lines and corners extracted using computer vision methods (Sobel filter, Hough transform, Harris corner etc.). We then created a number of logical rules to analyze different combinations of image segments and sequentially eliminate those of them that are unlikely to be a part of a door frame. As a result, in an event when the remaining image segments match a certain criteria, a presence of a door is being detected in a given image and the algorithm outputs a graphic overlay filled with a color outline around the perimeter of a door.The experimental results show that the proposed approach successfully detects typical doors in indoor and outdoor environments. In order to be detected by the algorithm, the doors could be opened or closed and a picture could be taken at different viewing angles that introduce various degrees of perspective distortions. Additionally, the algorithm is capable of detecting multiple doors in a single image. We achieved the accuracy rate of 67.42% as tested on the dataset that consisted of 132 images with an average inference time of 6.74 seconds.

6. Discussions and Future Work

Further improvements to the existing version of the algorithm could be made by focusing on finding optimal threshold values over a more diverse set of training images in order to improve generalization. Currently the main limitation of the algorithm (and a major focus of improvement work) is the fact that it requires all four corners of the door to be visible in the image for successful detection. That means that the algorithm is struggling with identifying occluded doors in its current version. This limitation puts the algorithm at disadvantage which makes improvement of generalization over images with occlusions a number one priority for future work on this project. Additionally, further improvements in model's ability to generalize could be made, as the current performance of the model is likely limited due to the following factors:

1. Relatively small size of the dataset (132 images) substantially limited our ability to create a truly robust set of logical rules that would fit more complex images and successfully capture less common types of door images and scenes.
2. Analysis of images that contain bookshelves, large windows, or complex patterns painted on walls often results in false positive detections due to indistinguishable similarities in geometric features of a typical door and some of the mentioned objects.

Based on the review of recent research and the impressive results in classification being achieved by machine learning models, and especially deep learning methods, further improvements could be made in the proposed approach. By augmenting our algorithm with a convolutional neural network (CNN) that could be trained on a combination of original images and the extracted elements of the picture (lines and corners) as features, an increase in detection accuracy could potentially be quite significant and is probably worth exploring. Although neural networks could sometimes be seen as black-boxes that do not necessarily provide much intuition in the model behavior, ease of implementation with transfer learning of pre-trained networks certainly outweighs that concern. However, implementation of deep learning methods in our algorithm would also require a significantly higher amount of training data to be collected.

Additionally, once a location of a door is identified with an acceptable degree of certainty, it could provide insight about 3D information. Since most of the doors have consistent geometric properties, we could use sides (edges) of the door to find locations of vanishing points and then use them to obtain additional information about 3D (e.g. infer information about structure of the room), so as soon as the team is satisfied with the detection rates on more complex images, 3D estimation would be a logical next step for the development of this project.

7. References and Materials

[1] “World report on vision” World Health Organization (2019)
https://www.who.int/publications/i/item/9789241516570

[2] “Apple previews innovative accessibility features combining the power of hardware, software, and machine learning”. Press release by Apple Inc., 5.17.2022
https://www.apple.com/newsroom/2022/05/apple-previews-innovative-accessibility-features/

[3] Muñoz-Salinas, Rafael & Aguirre, Eugenio & Gar, Miguel & Gon, Antonio. (2004). Door-detection using computer vision and fuzzy logic. WSEAS Transactions on Systems.
https://www.researchgate.net/publication/229018885_Door-detection_using_computer_vision_and_fuzzy_logic

[4] Et.al, Md. Akber Hossain. “Door Detection Based on Geometrical Features and Harris Corner.” (2021).
https://www.semanticscholar.org/paper/Door-Detection-Based-on-Geometrical-Features-and-Et.al/70b2c43068a2e175d155918dfe0c88cfe1d57257

[5] N. Karlsson, E. di Bernardo, J. Ostrowski, L. Goncalves, P. Pirjanian and M. E. Munich, "The vSLAM Algorithm for Robust Localization and Mapping," Proceedings of the 2005 IEEE International Conference on Robotics and Automation, Barcelona, Spain, 2005, pp. 24-29, https://doi.org/10.1109/ROBOT.2005.1570091

[6] Murillo, Ana & Košecká, Jana & Guerrero, Josechu & Sagues, C.. (2007). Door detection in images integrating appearance and shape cues. 2nd From Sensors to Human Spatial Concepts, held together with IROS. 7. 41-48. https://doi.org/10.1016/j.robot.2008.03.003

[7] Zhichao Chen and S. T. Birchfield, "Visual detection of lintel-occluded doors from a single image," 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Anchorage, AK, USA, 2008, pp. 1-8, https://doi.org/10.1109/CVPRW.2008.4563142

[8] Deep learning model for doors detection: A contribution for context-awareness recognition of patients with Parkinson’s disease, Expert Systems with Applications, Volume 212, 2023, 118712, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2022.118712

[9] Ramôa, J.G., Lopes, V., Alexandre, L.A. et al. Real-time 2D–3D door detection and state classification on a low-power device. SN Appl. Sci. 3, 590 (2021).
https://doi.org/10.1007/s42452-021-04588-3

[10] Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y. and Dollár, P., 2023. Segment anything.
https://arxiv.org/abs/2304.02643