Follow The Leader: A Following Approach to Solving the Elevator Problem

Makoto Kinoshita
Colby College
Waterville, ME
Email: mkinoshi@colby.edu

Kayla Freeman
Colby College
Waterville, ME
Email: klfreema@colby.edu

Matt Martin
Colby College
Waterville, ME
Email: mjmartin@colby.edu

***Abstract*—We propose a method to solve the problem of a wheeled robot accessing different floors of a building. This problem is important in beginning to think about how robots can integrate smoothly into everyday human life, without having to change the environment to fit the robot's needs. Specifically, we see how robotic wheelchairs, delivery vehicles, and cleaning robots will need to be able to access the different floors of an existing building without having the ability to climb stairs. Our procedure involves six different states in a state machine, and ultimately provides an effective method for a robot to reach an elevator to access different floors.**

I. INTRODUCTION

As robots begin to be useful in more areas of everyday life, it is important that they are able to access all areas of a building. They may, for example, need to deliver mail to offices on all floors of an office building, clean the entire building, act as a wheelchair or like CMU's Snackbot [10] need access to all office workers to be able to offer them mid-afternoon snacks. One of the largest challenges in solving problems like these are how to give robot's access to all areas of a multi-floor building. Large humanoid and other legged robots have the ability to climb stairs to reach a destination on another floor. Flying robots have the ability to easily move from one floor to another. Wheeled robots and other small legged robots, on the other hand, do not have a way to traverse stairs. This means that when a path to their destination requires moving from one floor of a building to another, they need to rely on an elevator.

An elevator can be operated many different ways. In most recent research, there have been studies into how to press the buttons autonomously. This is yet another insurmountable obstacle for a robot without an effector and too small to reach the buttons. Thus, for small simple wheeled robots, they are left with the option of relying on the help of humans to move from one floor of a building to another.

This paper will investigate a method for robots to use human interaction to traverse a multifloor building. Human interaction requires extra care because the robot needs to act in a way that makes the people around it feel comfortable. There is a need to be aware of how close the robot can get to people before they become uncomfortable and how fast the robot can move towards people before they come to fear it.

With these goals in mind, this approach to solving the problem of getting a robot into an elevator to move to another floor uses a combination of vision and laser sensors to guide it towards the elevator. It makes use of a Haar face detector to initially detect faces in the environment, and then creates a histogram that represents the color of the person's shirt made from subsets of images of that person. It then uses this information to follow the person towards the elevator at a safe distance and speed. The following algorithm uses a proportional control law in which the robot moves faster when the person is further away.

Once the robot begins following, it needs to have a reliable method for determining when it is in the elevator so that it will stop following the person. Although this could also be achieved using vision, adding specific visual cues like flags around the elevator eliminates the possibility of being able to be used on more than one elevator without the clear marks used for recognition. Therefore, a Hough line transform algorithm was used for feature detection. This process uses an accumulator to identify lines in the environment and then picks out the three lines with the most support that are distinguishable from each other. It had to be ensured that the detection was general enough to deal with variability of entrance angle and velocity but not excessively general to ensure other situations could not be confused with the elevator.

This combination of vision and feature detection using laser sensors creates a robust system that enables the robot to traverse a multi-floor building.

II. RELATED WORKS

A. Robots and Elevators

The problem of traversing elevators has been investigated by many authors. Our work was motivated by robots such as Carnegie Mellon University's Snackbot, which is one example of a service robot that currently only has the ability to work on a single floor of the research building because it does not have legs to climb stairs or effectors that are able to push the elevator buttons [10].

One of many challenges that required a robot to be able to use an elevator took place at the AAAI-2002 Robot Challenge. Multiple teams competed, and they all used human interaction to help in the process. iRobot's machine CoWorker navigated to the elevator autonomously but then required a human to push the buttons for it [9]. Grace, the result of a collaboration of multiple teams, reached the elevator using verbal directions given by a human, but then had to wait for a human to open the door for it. This could lead to problems in performance because the person needs to be able to accurately judge distances in order to lead the robot to the elevators. The robot then has to wait for someone to come upon it to open the doors. While this requires less time for the humans involved in the interactions, it is not as trustworthy as our method that requires the human to take the robot directly to the elevator [13].

More recent projects have path planning that allow the robot to localize where it is in its environment, plan a path to the elevator, and then use an end effector to push the buttons necessary to the the elevator [7] [8]. This is not a viable option since Magellan Pro robots do not have end effectors.

B. Human Interaction

Controlling and manipulating human interactions with robots is becoming a more and more relevant part of the robotics field. There is a need for researchers to consider how the humans in the situation will respond to the actions of a robot to make them feel comfortable and safe. We therefore considered the characteristics of robot speech and following that would make humans most comfortable when interacting with our robot.

Eyssel et al. studied the effects of a robot's voice by studying the receptivity and anthropomorphism of the robot when gender and synthesized-ness were changed [3]. They found that people responded most positively to humanoid robots whose voices matched their gender and who sounded most human. The researchers concluded that people have the most human-like interactions with robot's that they share some qualities with. In a robot that does not look at all humanoid, however, the uncanny valley would indicate that giving a robot a very human-like voice may seem curious coming from a machine. Therefore, we synthesized voice used to ensure that the human would feel comfortable approaching and being followed by the robot.

Walters et al. completed a study that investigated how close people will approach a robot depending on which type of voice the robot was using to talk: a realistic female voice, a realistic male voice, and a synthesized gender neutral voice [17]. It was unsurprising that the participants of the study did not approach the robot as close when it had a synthesized voice. A human-like voice makes the robot seem more human-like, and therefore more approachable. Interestingly, people stand closer to the robot at first interaction than they would to people they just met. The researchers argue that there are two explanations for this: that people view the robot as a member of their close family or friends, or that people view the robot as an inanimate object that they cannot interact with like a human. Since the robot used in our study uses a synthesized voice, a larger follow distance is needed to ensure that the humans feel comfortable in all aspects of the interaction.

Takayama and Pantofaru investigated traits of individuals that affect how comfortable they are in proximity to robots [14]. They found that individuals who have pets or have experience with mobile robots are likely to interact with robots in a closer proximity than those who do not. They investigated whether people approach robots closer than they let robots approach them, but there was not a significant difference, which is important to this project because humans are engaging in both activities. They further found that individuals who are very agreeable according to a personality test and have a positive outlook on robots in general will approach the robots closer than those who are less agreeable. While it is impossible to judge a person's personality using vision, it is important to keep in mind that individuals will deal with robots in a variety of ways based on their own personality and experience.

C. Line Detection

Straight line detection is a fundamental problem in robotics and has been challenging researchers since the inception of the field. Detecting straight lines in an environment is essential for robots to be able to accomplish several common tasks, ranging from navigation to localization and mapping. Many solutions to this problem have been presented, all of which have slight variations that depend on the robot in question and the specific use case. We focused our review on the split and merge algorithm, RANSAC line fitting and the Hough transform.

Split and merge was originally presented by Theodosios Pavlidis and Steven Horowitz in 1974 [12]. They proposed the general case for extracting features from data, primarily images or laser readings. Their algorithm has been revised several times, notably by Borges and Aldon who discuss its particular application to line fitting from two dimensional laser scans [1]. They integrate the traditional split and merge framework into a fuzzy clustering algorithm, which makes the algorithm more robust to noisy data.

Random Sampling and Consensus (RANSAC) is a model fitting algorithm proposed by Martin Fischler and Robert Bolles in 1981 [4]. It is a probabilistic method for identifying a specified feature or model in noisy data. Since it is effective at identifying models within noisy data, it is primarily used in image procession applications, or other situations where the sensor exhibits non-Gaussian irregularities or excessive outliers. In the general case, the RANSAC algorithm proposed by Fischer and Bolles works as follows: specify some number of points that are necessary to fit a model (for a line this would be two points), n. Then select n random points from all of the data data and try to fit the specified model to it. Calculate how many points from the whole dataset are "close enough" to the fitted model, and if that count is larger than some specified threshold, set that as the best model. This algorithm depends on several parameters, and must be tuned carefully for each usage. It has been used extensively since its introduction in 1981, and it has had success as a line fitting algorithm in robotics.

In 1972, Richard Duda and Peter Hart took the original Hough transform algorithm [6] and modified it into the form that it is widely used today [2]. The main adjustment that Duda and Hart made to the original algorithm was to use angle-radius pairs to define lines rather than slope-intercept pairs. They also expand the algorithm and explain how it can be used to fit curves and other shapes. They primarily apply the transform to image data, rather than laser data, which is what we use. In general, the Hough transform works by accumulating votes for all possible lines in the data. It defines lines by a specific angle and distance from the origin. This angle and distance represents the perpendicular line segment from the origin to that line. Then, the number of times that line appears in the data is counted, and the lines with the n highest counts are returned. the number of lines returned, n, is arbitrary which is important to our work.

D. Face Detection

Face detection and tracking has been a central topic in computer vision. Face detection technology is used in many services, and it is a critical part in human interaction in robotics. Even though it is a widely studied topic, many methods do not work properly under certain constraints created by the illumination, orientation of the face, and face variations. Moreover, face tracking methods have to detect a face in small time intervals to perform effectively with no noticable latency.

Yang and Waibel presented the stochastic face tracking model based on skin-color [18]. They found out that the histogram of the skin-color is clustered in a certain area in chromatic color space. They further found that skin-color of different people surely vary but their distributions are clustered in one place in chromatic space. The histogram depends not only on skin-color but also on the illumination. However, they discovered that the shape of distribution remains the same under different lighting conditions although there is a slight difference in the position of histograms. From these experimental results, they developed a method to adjust the skin-color histogram based on the lighting condition. Finally, they combined this skin color model with the motion detection system and the geometry detection system, which showed the robust real-time face tracking performance.

Foytik et al. proposed the system using a Kalman filter to track one or multiple faces [5]. Their system first detects faces using an Adaptively Weighted Modular Principal Component Analysis and follow each motion using a Kalman filter. In their previous research, they found out that the Kalman tracking system is capable of tracking one face correctly, but it is not able to track multiple faces. Therefore, they incorporated the Kalman filter with a low-level face matching algorithm. They found out that this system can track multiple faces even when there are overlapped faces.

Viola and Jones proposed a machine learning based approach to detect a human face [16]. They first introduced a new representation of images called integral image, which represents an image using the sum of surrounded pixels rather than using pixels directly. This method produces features of images quickly and speeds up the process of detecting face. As a learning algorithm, they chose the AdaBoost learning algorithm using features represented as a form of integral images. Furthermore, to improve the accuracy of face detection, they introduced a process called cascade, which is similar to a decision tree. By incorporating all of these features, they developed a complete system that has 38 stages with more than 600 features. Their system substantially reduced the computation time to detect a human face.

III. Physical Hardware

A. Robot

Fig. 1. iRobot Magellan Pro

The Magellan Pro is circular, differential drive robot. It is 40 cm in diameter and 25 cm tall with a castor wheel in the back to eliminate tipping while driving and an additional wheel in the front to prevent tipping forward in the event of a fast stop or collision. It has 16 sonar, 16 infrared (IR), and 16 bumper sensors built onto the main unit.

B. Laser

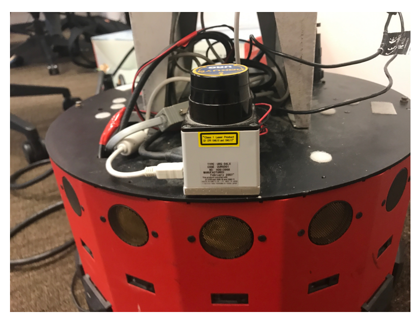

Fig. 2. Hokuyo URG-04LX

The main reactive sensor used is Hokuyo's URG-04LX Scanning Laser Rangefinder. It has a 240° degree scanning range with a 0.36° angular resolution. It has a range of 0.002m to 4m and a resolution of 1mm.

C. Camera

The camera used for vision tasks was the Logitech Webcam Pro 9000. It has a 75° diagonal field of view, and captures 340 by 240 pixel images.

Fig. 3. Logitech Webcam Pro 9000

IV. Proposed System

In this paper we propose a robot system that traverses a multi-floor building using human interaction. The decision-making process of this system is based on a state machine. In Fig. 4 we present the general diagram of this state machine.

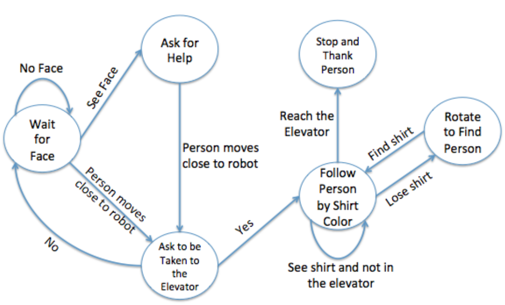

Fig. 4. State Machine Diagram

The initial state of this robot is wait-for-face state, and it runs face detection program using Haar Feature-based Cascade Classifiers, which is a machine learning based method. This program is implemented using OpenCV. Once it detects a face or person approaching to the robot, it changes its state to ask-for-help state or ask-to-be-taken to the elevator state. In these states, it runs the human interaction program using speech.

We determined that it would be most comfortable for the humans in the environment if the robot was stationary while it was in the wait-for-face state. While wandering might allow the robot to find a face faster, it is also harder for the system to identify faces while the robot is moving. It also might make people uncomfortable if the robot wandered too close to them. We determined that once the robot sees a face in the environment it should sit and wait for the person to approach it rather than having the robot approach the person. Based on related works, we considered how people would feel with robots in their personal space and decided this arrangement will give the people the power over the situation rather than the robot.

When the Haar face detector identifies a face in the environment, a message is pushed to the control system by the Social Vision Module (SVM), indicating where the face is. Once the control system has received this signal, it prompts the robot to speak to the human, saying "Hello, can you help me?".

Based on previous research outlined in the Related Works section, we determined that a synthesized voice is the best option for this system because the robot is too far from human-looking to be considered a human-like being. The uncanny valley indicates that trying to make a robot that does not look human but sounds human may make people uncomfortable interacting with it.

Due to people's natural inclination to be empathetic, the robot's pleading for help will push people to approach the robot like they would approach a person. Using a short range of laser sensors in front of the robot, the system determines when a person has gotten within some minimum distance to the robot. When it has, it asks, "Can you take me to the elevator?". The person then has the chance to respond yes or no by typing on the keyboard, triggering the robot to either move on the the following of the person or to go back to looking for a new face.

Based on past research into comfortable distance between people and robots, the fact that a synthesized voice was used implies that a larger minimum approach distance needs to be used. While a 0.2 meter approach or follow distance has been used for past projects, this project requires a 0.3 meter approach and follow distance to keep humans comfortable in their interactions with the robot.

We modified the SVM operator that is responsible for detecting faces so that it has a secondary purpose: once it has identified faces, it begins to collect images of where it thinks the person's shirt is most likely to be. It does this by finding a bounding box that the Haar face detector indicates encapsulates the entire face, then move it down below where the bottom of the face is. This accurately locates a large section of the front of the person's shirt. The operator loops through the bounded area and converts the pixels from an OpenCV image to a Pixel array that can be written out to a ppm image.

When the robot enters the following state, the control module takes the previously written images and creates a histogram of the colors represented in the images.

A new operator is pushed to the SVM module, so that it transfers its priority from detecting faces to detecting the color that is represented in the histogram. A bounding box is made that represents the outline of the location of the color in the image stream coming from the web cam. The SVM modules pushes back a message to the navigation module that contains the location of the box, which is used to determine the velocity commands that are sent to the robot. The navigation module uses the following proportional control algorithms to determine the forward and rotational velocities: the forward velocity is the minimum laser reading in the area in a small range in front of the robot minus the follow distance; the rotational velocity is determined by the column number of the center of color minus the center of the image.

In the event that the robot loses the color from its vision, its translational velocity is immediately set to 0, and the rotational velocity remains constant until it next sees the color in its sight. This serves two purposes: first, it allows the robot time to catch up to the person if they move too far to the left or right too quickly for the robot to follow; and second, it allows the system time to find the color at a slightly different angle if the lighting has masks the presence of the color the system is looking for based on the histogram.

The navigation module has a reactive control elements that prohibits the system from allowing the robot to move too quickly both to protect the motors from overexerting and people and obstacles in the environment from being hit by a robot moving too quickly. It also uses a combination of the sonar, IR and laser sensors in all directions around the robot to create an emergency stop function in the event that an obstacle becomes too close to the robot.

As the robot is following the person, it runs a Hough line transform algorithm each time iteration of the main navigation loop. The purpose of this algorithm is to detect when the robot has entered the elevator. We observed that the elevator is realistically defined as a closed box with three perpendicular walls that completely surround the robot. Thus, we felt that if we took the three best lines from a Hough transform, checked that they were perpendicular and at an appropriate distance from the robot, we would be able to detect precisely when the robot enters the elevator. We chose the Hough transformation instead of the other line detection algorithms examined in the literature review for a couple reasons. First, RANSAC was not a logical option for this application since the biggest advantage of RANSAC is that it is able to identify lines with a lot of outliers. This should not be a problem in our case since most of the laser readings in the elevator will be a part of a line. We chose Hough over split and merge because we thought that Hough would be more effective at finding the n best lines, rather than the best lines. Since we wanted the three best lines, we decided the Hough transform was the best algorithm for this application.

V. Results and Experiments

A. Experiments

We designed our experimentation first to test each distinct aspect of our software architecture independently, before testing them in conjunction.

1) Human Interaction: First, we tested our instructions for handling the initial state, when the robot is waiting to find a face. To do this, we simply set the robot in an empty space and let it loop, looking for faces. We walked in and out of the camera's view in order to see how effective it was at recognizing people. We found this to be fairly effective, as the built in face detection in SVM is pretty robust. Then, we wanted to test how persistent the robot should be in asking people for help. Once it sees a face, it asks that person for help politely about every three seconds. We varied this frequency during our testing to try to find the optimal value. We also

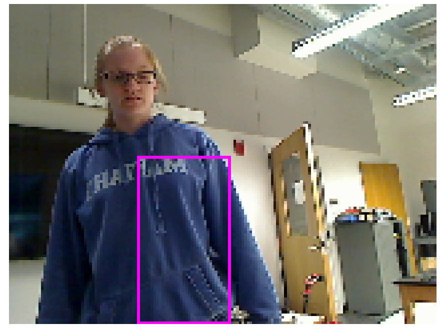

Fig. 5. Location of shirt bounding box

wanted to test how proficient the robot was at recognizing that the person was interested in helping them. We adjusted the laser threshold with several different values to test this.

2) Following: Next we needed to test how good the robot was at following a person. this involved testing the effectiveness of the histograms that we collected during the human interaction phase, as well as testing our proportional controller for the actual following. Here, we really focused on tuning our proportional constant to allow it to reach steady state quickly and accurately.

3) Elevator Detection: Testing our elevator detection algorithm was straightforward. We started by simply placing the robot inside the elevator and adjusting the parameters of our Hough transform until it recognized that it was inside the elevator right away. Then, we continued running the transform but drove the robot around in the hallway and elsewhere, in order to test for false positives. Finally, we tested it as the robot drove into the elevator, since we wanted it to stop only when it was fully inside, not just in the elevator doors.

4) Overall Experiment: Once we got each of the individual components of the system working, we tested the entire process from start to finish. To do this, we set the robot in the middle of the lobby of Davis, and approached it. We followed its instructions, and tested that it transitioned smoothly from state to state, and correctly recognized when it ended up in the elevator.

B. Limitations

1) Light in the Environment: A limitation of this proposed system is that it is very susceptible to light in the environment. The Haar face detection system is not able to locate faces if they are distorted by light coming from behind the head. This results in a failure to detect faces in the environment surrounding the robot, which halts the progression of the program. Furthermore, the histogram of color that allows the robot to follow a person based on the color of their shirt is taken in a very specific environment: whatever lighting is available as the person approaches the robot. This results in poor performance when the lighting dramatically changes to very bright or very dark (which was not a problem in our testing environment). It does well trying to transition to moderately darker environment, but can get distracted by the presence of white light. This materializes in the robot focusing on an overhead light or a window and approaching it rather than the person it should be following. To avoid problems with this limitation, experiments were run at night to avoid the effects of bright natural light coming in the large windows in the environment distracting the robot and affecting its following abilities.

2) Shirt Color: Finally, there is the possibility that the color of a person's shirt matches the color of obstacles in the environment. The system is not able to detect color that is attached to a person in comparison to color that is in the environment, so the robot may become distracted in these situations and fail to follow the person it should. In this case, the robot will approach the object and sit the inputted distance from it, which although is incorrect, is safe for all in the environment. A second limitation is that since the system uses images of the front of the person's shirt to make the histogram that is responsible for correct following, if the person's shirt is a different color on the front than it is on the back, the robot will not be able to follow if the person is walking forward. Based on basic surveying of people in the area where they system will be used, this seemed to be a reasonable assumption to make. To avoid these limitation in this system, a yellow vest was used for the robot to follow and the person walked in a way that ensured that the yellow was facing the robot's camera at all times.

C. Results

For the human interaction portion, we found that asking a person for help once every three seconds was appropriate. We found this to be a comfortable level of interaction, without being too pushy or allowing the person of interest to sneak away. Our testing of the following algorithm allowed us to tune our proportional controller to be somewhat aggressive, since it was often following a person in a straight line. However, we did find that when the robot was distracted by lights or other things in the environment that looked like the shirt color of interest, it wandered off. The fact that the robot stopped moving and started rotating when it completely lost the color turned out to be an important feature, since it allowed the user to track back to the robot and help it find the color again. Finally, we found that our elevator detection algorithm worked very well after tuning and debugging, never giving a false positive and always realizing it was in the elevator within a couple time steps.

VI. Conclusion

A. Discussion

Overall, we were satisfied with how our system performed. It managed to accomplish the goal of getting to the second floor of a building without using the stairs. We believe that the biggest weakness of our algorithm is the person following stage. It is fairly easily distracted by lights and other variables in the environment. Also, it was disappointing that we could not get it working with using a person's regular shirt color and that the system mandated the person use yellow felt. However, we were happy with how accurately the robot could detect when it was in an elevator. We also thought that we found the right balance of persistence and patience in the human interaction phase.

B. Future Work

Future work should be executed to address the limitations of this system: namely to eliminate the effects of the bright environmental light on the system and to allow the system to be able to use a wider range of shirt designs and colors to follow the person.

1) Light in the Environment: Based on experiments completed for this study, the SVM module operators are compromised by bright light of any kind: sunlight, overhead lights, and reflections of either of these lights in glass and metal. There was discussion of completing some type of post-processing on the histogram completed for color following to add variety in the shade of the color detected. While this might increase the effectiveness of the system, it will not solve all of the problems because the Haar face detection operator also does not function if faces have bright back light. Thus, investigations are necessary to determine how to filter out bright light in an effort to increase the functionality of this system. It is possible that the input images could be filtered before they are used in the SVM operators to eliminate or darker pixels that are excessively bright.

2) Shirt Colors: There should also be future work to investigate techniques into how to use the color of the back of a person's shirt rather than the front to eliminate the assumption that the front of a shirt and the back of a shirt is the same. The current system will not work if there is a large logo on the front of the shirt, if a person is wearing an open sweater, or many other clothing scenarios. Thus, a possible solution would be to write an operator to determine the likely location of the back of the person's head once they have turned around based on the previous location of the person's face then solidify the location using the probable shape of a head. If the back of a head can be found, the color of the back of the shirt can be found in the same way as the front of a shirt is found in this system.

3) Navigating Out of the Elevator: This proposed system focuses solely on the task of moving the robot from the lobby of a building to inside the elevator. For the system to have maximum utility, it must also have the ability to get out of the elevator once it has reached its desired floor. This could require the robot to turn around and either find the person so that it is able to follow it out of the elevator or turn around and identify where the door is either based on landmarks placed on either side of the door or by the shape of the doorway.

4) Reduce Human Reliance: In many realistic applications of our work, it would be a distinct advantage, if not a requirement, for the robot to be able to navigate to the elevator completely autonomously and with no human guidance. However, once we take away that aspect from our work it becomes a different problem, one of localization and navigating on a map. This would be an important step to take, but requires moving the focus away from computer vision problems and toward mapping and localization problems.

The combination of these advancements would increase the breadth of situations in which this system can be used.

ACKNOWLEDGMENT

The authors would like to thank Professor Bruce Maxwell for his encouragement and guidance through this project.

REFERENCES

[1] G.A. Borges and M.J. Aldon, "A Split-and-Merge Segmentation Algorithm for Line Extraction in 2-D Range Images", *Pattern Recognition*, 2000, pp. 441-444.

[2] R. Duda and P. Hart, "Use of the Hough transformation to detect lines and curves in pictures", in *Communications of the ACM*, vol. 15, 1972, pp. 11-15.

[3] F. Eyssel et al., "If You Sound Like Me You Must Be More Human: On the Interplay of Robot and User Features on Human-Robot Acceptance and Anthropomorphism", *Proc. 7th ACMIEEE Int. Conf. Hum.-Robot Interact. HRII2*, 2012, pp. 125-126.

[4] M. Fischler and R. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography", in *Communications of the ACM*, vol. 24, 1981, pp. 381-395.

[5] J. Foytik et al., "Tracking and Recognizing Multiple Faces Using Kalman Filter and ModularPCA", 2011, pp. 256-261.

[6] P. Hough, "Method and means for recognizing complex patterns", U.S. Patent No. 3,069,654. 18 Dec. 1962.

[7] J. Juang et al., "Real-Time Image Recognition and Path Tracking of a Wheeled Mobile Robot for Taking an Elevator," *Acta Polytechnica Hungarica*, vol. 10, 2013, pp. 5-23.

[8] E. Klingbeil et al., Autonomous operation of novel elevators for robot navigation, in *Proceedings of ICRA*, 2010, pp. 751758.

[9] B. Kuipers and A. Stroupe, "The AAAI-2002 Robot Challenge," *AI Magazine*, vol. 24, 2003, pp. 65-71.

[10] M.K. Lee et al., "Ripple Effects of an Embedded Social Agent: A Field Study of a Social Robot in the Workplace," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2012, pp. 695-704.

[11] C. Nass et al., "Computers are Social Actors," in *Human Factors in Computing Systems*, 1994, pp. 72-78.

[12] T. Pavlidis and S.L. Horowitz, "Segmentation of Plane Curves", *IEEE Transactions on Computers*, vol. C-23, 1974, pp. 860-870.

[13] R. Simmons et al., "Grace: An Autonomous Robot for the AAAI Robot Challenge," *AI Magazine*, vol. 24, 2003, pp. 51-72.

[14] L. Takayama and C. Pantofaru, "Influences on Proxemic Behaviours in Human-Robot Interaction", in *Proc. IROS*, 2009, pp. 5495-5502.

[15] D. Troniak et al., Charlie Rides the Elevator Integrating Vision, Navigation and Manipulation Towards Multifloor Robot Locomotion, in *Proceedings of the 10th International Conference on Computer and Robot Vision*, 2013, pp. 18.

[16] P. Viola and M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features," in *Proceedings of the Conference of Computer Vision and Pattern Recognition 2001*, 2001, pp. 1-9

[17] M. Walters et al., "Human Approach Distances to a Mechanical-looking Robot with Different Robot Voice Styles," *Proc. IEEE Int. Symp. Robot and Human Interactive Communication*, 2008, pp. 707-712.

[18] J. Yang and A. Waibel, "A Real-Time Face Tracker", in *Proceedings of the IEEE Workshop on Applications of Computer Vision*, 1996, pp. 142-147.