PhotoBot: An Autonomous Party Photographer

Fatma Akcay
Harvard College
fatmaakcay@college.harvard.edu

Emily Wang
Harvard College
emilyswang@college.harvard.edu

May 9, 2016

I. Abstract

In this project, we developed an autonomous photography robot, PhotoBot, which could interact with users in a party atmosphere to take photos. The culmination of this final project involves a live demonstration of the PhotoBot in a real party setting for 2.5 hours: at the 2016 SEAS Design Fair at the Science Fair Plaza.

II. Objectives and Conception

The original role of the robot was to act as a photographer in a party setting. In the final project module on Canvas, it was suggested that the photographer robot must capture the crowd's attention and smiles, show the scope of a party, and provide a means to share the photos to the public. Building a photographer robot instantly drew our interest, as a photographer robot could: i) provide much use at the party by capturing the fun moments from a turtle's eye view and ii) act as a fun source of entertainment as individuals love taking photos of themselves. In the brainstorming process, we defined metrics to determine if our project was successful and identified what our main areas of focus were for developing a robust PhotoBot. Our first priority was to design a natural and intuitive human-PhotoBot interaction. We wanted to achieve a smooth interaction through voice commands from the PhotoBot, inputs from the user, and an unique outer design of the PhotoBot that clarifies its role as a photographer. Second, we explored ways to improve the quality of the PhotoBot's pictures, and we defined good party photographs as ones with no blinking, happy faces and natural smiles, and a diverse range of party goers. Third, the PhotoBot had to navigate a room without colliding into walls or partyogers: it must smoothly explore the room and mingle with different people at the party. Finally, the PhotoBot had to increase the level of fun at the party: it must talk to and entertain guests to make them smile and brighten their days, and attract users to come over. To further elevate the level of fun, the photos must be uploaded to the Internet where party goers can immediately check out the photos they took to share with friends. For most of this project, during the engineering process, out of all of these objectives, we focused on optimizing the human-robot interaction, so that PhotoBot's photo-taking process made sense to humans.

III. Implementation

In our final implementation, we designed the PhotoBot to act like an automated photobooth. The PhotoBot's appearance is attention-grabbing (Figure 1). A platform with a long neck elevates the webcam to better take photos of faces, and on the platform, there are speakers, a bright, colorful balloon at its head, and a line of flags spelling out *PhotoBot* for the crowd to see. The webcam is framed by a foam camera shape and arrow (Figure 1), to indicate to users where to look. On the neck of the platform hangs a box containing photobooth props, such as heart-shaped sunglasses and DJ

Khaled memes. The PhotoBot's exterior design makes it very easy for partygoers to identify the PhotoBot's role as a photographer.

The PhotoBot moves at a slow velocity in the party venue, and uses depth sensing from the Kinect to avoid colliding into obstacles such as people or walls. If the PhotoBot senses an obstacle, it turns slightly to avoid the obstacle, then continues to move. As the PhotoBot moves, it scans the images received from the webcamera attached on its head for human faces. To detect faces, we used a pre-trained haar feature-based cascades classifier from OpenCV, where the classifier was previously trained on many face and non-face images to extract haar features, in order to more accurately detect a real face. Upon detecting a true face, the PhotoBot will introduce itself to the partygoer and ask if he or she would like a photo.

Figure 1: *PhotoBot at the SEAS Design Fair*

At this point, we use the Python speech recognition library SpeechRecognition 3.4.3 to listen from the laptop's microphone for the user's feedback and convert the user's voice to a string using Google Speech Recognition API, which requires Wi-Fi. If the user says any statement with a *yes*, *sure*, or *yay* in it, the PhotoBot will proceed with informing the user to take a prop from the prop tray, tell the user to smile, and countdown from 3, 2, 1. If the user says any statement with *no* or *nope*, the PhotoBot questions the user as to why it doesn't want a photo, is shocked the user does not want a photo, and then moves on to find more people. During the SEAS design fair, due to the excessive background noise, we used a numeric pad to more accurately confirm the user's interest for a photo rather than through speech recognition.

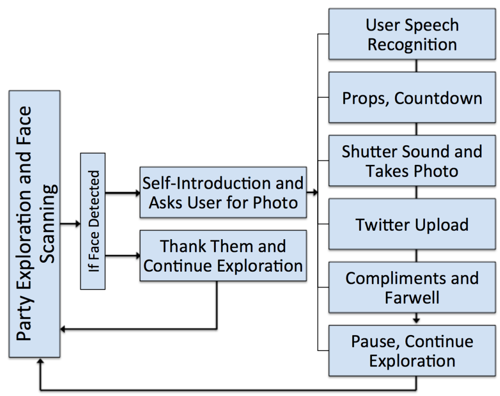

Figure 2: *Workflow of the PhotoBot*

Both methods reduce the change of false positives: they ensure that a PhotoBot only takes photos when a user specifies it to, and also confirms that the PhotoBot actually takes a photo of a real person, rather than, for instance, a window in Pierce which is detected by the classifier as a face (the window cannot speak or press a numeric pad, so PhotoBot will not take its picture). Before taking the photo, a shutter sound is produced in order to mimic the behavior of a real camera and indicate to the user when the photo is taken. The PhotoBot also randomly chooses whether it wants to take one photo or a series of three photos. If it chooses to take three photos, it will prompt the user to keep posing by declaring the famous DJ Khaled quote, "Another one." In fact, the PhotoBot's personality is based off of DJ Khaled, a rapper renowned for his humorous sayings and Snapchats.

With a sense of humor similar to DJ Khaled, as well as props of DJ Khaled memes, the PhotoBot adds another layer of light-hearted fun to the party. Upon taking the photos, the PhotoBot places a Harvard SEAS logo on top, and then uploads the photos to Twitter via its account *@seas_photobot*, so the partygoers can see the photos immediately. There is a Harvard SEAS logo on the bottom right corner of the image that is posted on Twitter, and a random Tweet is posted along with each photo, using the humorous language of DJ Khaled. The PhotoBot thanks the partygoer and then explains it must find new people; it pauses the face detection for a short duration and traverses in a new direction in the room to achieve better exploration. The cycle then repeats, where the PhotoBot searches for a new face in the crowd to take photos.

IV. Engineering Process and Results

Our engineering process was broken down into three main parts: professionalism of photography, the robot-human interaction, and movement. In this section, we discuss our process as well as the major difficulties we faced with each part, as well as the results achieved in the end.

1. **Photography Professionalism** A major challenge during this project was taking the photo at the right time to capture the right expression of the user. The webcam subscriber constantly passes in a new image into the "takePhoto" function. When a face is detected, the process of introducing the robot to the user begins. We found that if this entire process is started within the function that also conducts the face detection, the photo stream freezes, waiting for the robot to introduce and save the image. As such, the image that is saved is the one when the face was first detected, and not the one the user posed for. Normally, the face is detected through the non-posed image is one with the user blinking or staring with confusion at the robot, not the one with a happy, smiling user holding a prop. In order to solve this problem, we removed our face detection and photo-taking process from the "takePhoto" function. In doing so, every frame that is passed in from the subscriber is checked for a face, and the moment a face is detected, a global boolean, "faceDetected" is set to True, and the process for taking an image begins in a separate function. Thus, the image stream is not paused because this is done outside of the subscriber function. This also allows us to take a series of three photos within the same function iteration, because the image is constantly updated by the webcam subscriber.

2. **Robot-Human Interactions** In addition to taking well-timed photos, the robot-human interaction had to be natural. We found that the more vocal commands the PhotoBot gave, the better the user interaction. The PhotoBot begins by introducing itself, asking for user input (originally a key press via the numeric pad, which became voice commands through speech recognition in further iterations). If indication of user agreement is received, the PhotoBot tells the user to take a prop, giving an reasonable amount of time to do so. We experimented with different timings in between statements to determine the optimal amounts of the time for the user to follow through with each command. Moreover, in our initial tests, users found it confusing as to when the photo was actually captured, and they would pause and smile in a forced manner for a long duration until their mouths were tired, which often led to photos with blinking and a stiff appearance. To combat this, we implemented a count-

down and a shutter sound, which clarified when the PhotoBot took a photo, and significantly improved the smile and eye contact in the resulting image. The PhotoBot will also play random voice clips of DJ Khaled as a way of complimenting the user or prompting them to take another photo if it is taking a series of three photos. Despite its robotic voice, the PhotoBot's comments in the style of DJ Khaled create a fun atmosphere that puts users at ease. By allowing the PhotoBot to take on this unexpected personality, users are intrigued and excited to respond to the PhotoBot.

Figure 3: *Example photo taken by PhotoBot and uploaded to Twitter at the SEAS Fair*

3. **Movement** Initially, we tested the functionality of our PhotoBot with linear velocity set to 0. The PhotoBot would simply rotate in place until a face was detected and then carry out its tasks. During the rotation face detection process, we noticed that if the PhotoBot spun too quickly, no faces would be detected. Since Haar cascades can have many false positives depending on the environment background, we wanted to increase the probability of detecting a real face by reducing the number of false negatives caused by moving too quickly past a face. As such, we set the velocity of the robot to be very low in order to ensure a stable image feed and prevent the PhotoBot from speeding by potential users. Moreover, for movement, we tested a timed strategy, where the PhotoBot would move to a new location after a given amount of time, and slowly spin in place until it found a face, similar to a how a photographer finds a location and then gazes around to see which people to take photos of. Through more testing, however, we found that the best strategy for ensuring a diverse range of people was discovered was to simply move forward unless there is an obstacle or a face detected, and after taking a picture, stop face detection for a given time and move to a new location, to prevent taking photos of the same user the entire time. Moving forward to detect faces was more effective than turning to scan for faces. By basing the movement off of whether or not a photo was taken, rather than based on allotted time, we achieved a better flow of the PhotoBot throughout the room.

V. Discussion and Future Work

One of the PhotoBot's strengths is its humor when communicating with people, which stems from its creative language inspired by DJ Khaled. While we were testing the TurtleBot in the classroom, many of our classmates would suddenly burst into laughter in the middle of coding because of the PhotoBot's compliments and questions. At the design fair, people were also intrigued by the DJ Khaled personality and chuckled to his statements about "the key to success." An exciting avenue to explore would be to place the PhotoBot in various locations around Harvard, such as in Maxwell Dworkin, in the Science Center, in CGIS, and monitor how the PhotoBot interacts with people on a daily basis, or a non-party setting. Perhaps, with this humor combined with its desire to capture photos in people's lives, the PhotoBot can help reduce stress around campus and act as a source of positivity. Many people love taking photos, hence the rise of selfies, so the PhotoBot's job would be one appreciated by most individuals. The PhotoBot could keep a daily record of what is going on in each person's life. We are curious as to whether the

PhotoBot can serve as a stress reliever to bring joy to people's lives.

At the SEAS design fair, we found that another major strength of the PhotoBot was its unique decorative appearance, which captivated passerbys who would ask questions and were intrigued to try communicating with the PhotoBot. The props were also appreciated; people at the SEAS fair told us it made the PhotoBot's interaction more lively, and the phototaking process more exciting. Other feedback we received from the SEAS fair was that the Harvard SEAS geofilter added an air of legitimacy to the photos, and that the instantaneous Twitter uploads were a nice surprise. Furthermore, the PhotoBot takes advantage of and capitalizes on our addictions to self-promotion and desire to publicize to the world more photos of ourselves. The PhotoBot's weaknesses are that speech recognition is only functional when Wi-Fi is present; while humorous, the PhotoBot is not intelligent enough to have a real conversation (the responses are hard-coded). To resolve these problems, we can experiment with other speech recognition libraries that do not use Google, as well as voice libraries, and use machine learning approaches to generate more natural and flexible conversations through social learning. Furthermore, the robotic voice is often difficult to discern for newcomers, which creates a communication barrier, especially in loud environments such as the SEAS fair; at the SEAS fair, many times, people could not understand when the PhotoBot indicated for the user to press the keypad and would ask us (we were standing nearby) when the PhotoBot was actually taking a photo. Another weakness of the PhotoBot that we observed at the SEAS fair is that in large crowds, it simply turns in place and waits for the crowd to disappear, as it senses obstacles. In heavily packed parties, there may always be huge crowds present, so we could experiment with ways to notify the partygoers to clear a path to let the PhotoBot escape the havoc, as a real person would do, rather than stalling the PhotoBot from taking on its duties by waiting for obstacles to disappear. It is essential that we experiment with ways to increase the PhotoBot's robustness at communication and navigation in at a loud and crowded party. Other interesting extensions include improving the PhotoBot keep track of the faces of individuals it's taken photos of, so it can greet them by name and have more personal conversations, adding on stickers based on the locations, adding personalized hashtags to Twitter based on each conversation by keeping track of what the person says in more depth (ie: if a partygoer discusses reinforcement learning with the PhotoBot, the PhotoBot will post "Best machine learning conversation! $\#rl\ \#fun$". In conclusion, through our final project, we developed an autonomous and fun-loving photography robot, which can capture exciting moments at a party.

To see the PhotoBot in action, please view this video where the PhotoBot communicates with a user at a mock party (held at Pierce) where the user accepts a photo in the first round, then rejects the photo in the second round. In this video, footage from the PhotoBot at the SEAS design fair can also be found! Photos the PhotoBot took at the SEAS Design Fair are located on its Twitter feed. Finally, the PhotoBot's code can be viewed on the code.seas repository.

VI. Acknowledgments

We would like to thank Radhika Nagpal for her unwavering enthusiasm during lectures and demonstrations, clarity when explaining ideas, and dedication towards helping us explore the field of robotics in a hands-on process. We would also like to thank Mike Kester for his sarcastic humor which helps to reduce stress, his commitment in guiding us when stuck on hard problems, and patience towards answering all sorts of questions. Moreover, we would like to thank Serena Booth for her active and creative feedback during the brainstorming process, constant support throughout the semester, and effort in helping us figure out logic errors in the code. Finally, we would like to thank our peers for common and infectious excitement for robotics.