The University of Birmingham
School of Computer Science

MSc in Advanced Computer Science

Human-Robot Interaction System Capturing Human's Interest via Eye Gaze Estimation

Vivek Chauhan

Supervisor: Dr. Hyung Jin Chang

September 2018

Abstract

Human-Robot interaction is an area of robot vision which addresses the understanding between humans and robots through communication. Understanding of human behavior is an essential part of operating Human-Robot interaction naturally.

In this work, we document a detailed analysis of a robotic system which is capable of predicting human attention using eye gaze estimation. This system aims to understand the human attention in an office environment. We have formed this system with two main components. We first detected a user's gaze direction and his/her surrounding objects and then by combining both parts, we predicted where a user is looking at. To evaluate the performance of the system we have conducted multiple experiments.

Keywords— Human-Robot Interaction, Gaze Estimation, Path Planning

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Autonomous robots are making their way into everyone's life by assisting users in their daily activities. In order to design these systems, robots have to understand a few communication skills and its surrounding environment. Appearance-based eye gaze estimation (2) is a crucial component in such applications to analyze human behavior and social interactions (3).

In our work, we have proposed an HRI system which can predict the user attention by understanding user surrounding objects and gaze. The experiment setup consists of two Astra Pro cameras and a mobile robot (Pioneer P3-DX). To validate our experiment's robustness we have tested the system multiple times with five users to report the prediction and evaluation rate as well as the path planning.

The structure of the thesis is organized as:

· Chapter 1 gives a brief introduction to the problem.

· Chapter 2 provides the information about the related work that has been done in the area of Human-Robot Interaction, gaze estimation, and object Detection.

· Chapter 3 presents the design of the system.

· Chapter 4 presents various experimental results

· Chapter 5 discuss and concludes the thesis.

Chapter 2

Related Work

The task aims to study the different aspects of Human-Robot Interaction. This chapter gives general information about the traditional methods of gaze estimation, object detection, and human image viewing behavior.

2.1 Object Detection

Convolutional Neural Network has commonly used for computer vision problems such as image classification where a trained network returns a label for an input image (4). In recent years, to solve the object detection problem three approaches have been proposed: R-CNN(5), Fast R-CNN (6), and Faster R-CNN (7).

R-CNN,(5) proposed a method where a segregation algorithm is used to extract multiple regions of an image and pass these regions through a CNN. After R-CNN, (6) proposed a new method to decrease proposal computation by using convolutional feature map based region proposals. Due to the usage of selective search, the performance of both R-CNN and Fast R-CNN is quite slow. Therefore, (7) proposed a Region Proposal Network (RPN) which allows the framework to learn a cost-free region proposal.

Another approach You Only Look Once (YOLO) for object detection is proposed by (1). in which an image is divided into multiple cells, and these grid cells predict the confidence score of a bounding box as shown in figure 2.1. Even though Faster R-CNN is more accurate, YOLO is three times faster. Due to that, we have used YOLO in our design.

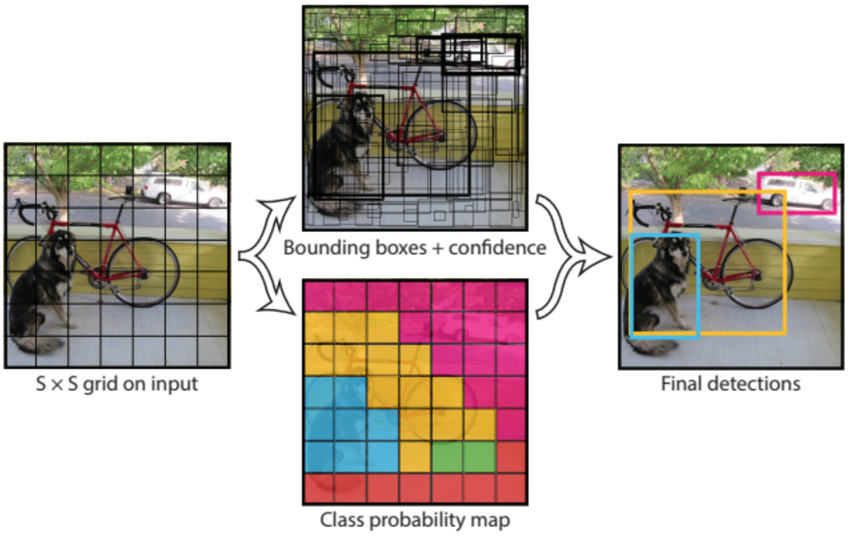

Figure 2.1: Multiple Object Detection in YOLO (1)

2.2 Gaze Estimation

For a long time, human gaze estimation has been addressed many times. It can be divided into two approaches (8): model-based and appearance based. The advantage of using appearance-based gaze estimation is that it can be used on low-resolution eye images (9; 10) On the other hand, model-based methods use a model of an eye which can require an additional light source (11; 12). In recent years, Convolutional Neural Networks (CNN)(13; 14; 15; 16), Random Forest(17; 18), and support vector machine (19) based approach has been proposed to use in appearance based gaze estimation. Out of these techniques, CNN based gaze estimation (16; 20) is proven to best.

Different types of input have been examined to predict the gaze estimation which includes full-face or eye images with the landmark information (20; 14). Eye-based gaze estimation is less robust than face-based CNN (20). Due to this, our design uses information from both eye and face regions.

2.3 Human-Robot Interaction

There has been much research conducted on Human-Robot Interactions where the robot performs different tasks as a tool, assistant or guide. As tool robots are commonly used in dangerous tasks. In (21) a robot performed a task of placing the radioactive waste better than humans. Under normal conditions, a robot can perform dynamic tasks quite well, but hazardous tasks have not been done autonomously.

In recent years, robots are being used as a guide in museums where they assist and interact with users through audio or displays. In (22), an autonomous robot was used to enhance the interaction between the robot and museum visitors by displaying the mood or emotions. This experiment showed how interaction through verbal and non-verbal communication affects communication between humans and robots.

Even though many robots work in a limited environment, they can still assist humans in various tasks. Considering these outcomes, we have proposed our approach in this dissertation which can further be useful for Human-Robot interactions.

2.4 Viewing Behavior

Viewing behavior depends on the characteristics of an environment itself. The surrounding environment of a human can be represented as images. Human eye movements can be divided into two categories: saccades and fixations. Saccades mean when an eye moves swiftly between two interest points. On the other hand, fixation is a duration in which an eye steadily focus on a single interest point. Fixation depends on a few factors such as the salience of an area (23), information about the surrounding environment (24), and task (25).

In recent years, many research has been conducted on the influence of gaze with different objects (26) and how gaze is used for the prediction of semantic categories (27). In our experiments, we use gaze direction to measure the distribution of user attention with their surrounding objects.

Chapter 3

Design

The system was designed to answer our research question:
How can a robot predict the attention of a user in an office environment?
No previous research has been done to predict the attention of a user by gaze estimation. Due to that, two subquestions arise:

1. What kind of information we need from a user and his/her surrounding environment?

2. How can we predict the attention from the above information?

3.1 Experimental Setup

The experimental setup can be divided into two steps to answer our questions:
First question, *What kind of information do we need from a user and his/her surrounding environment?* To evaluate the attention of a user, first, we have to gather information about the gaze direction and surrounding objects. In order to obtain this information first, we place an Astra Pro (RGB-D) camera in the front of a user as shown in figure 3.2 which is helpful to determine where s/he is looking at (figure 3.3). After that, we detect the surrounding objects such as a book, cell phone, etc. of the user using another Astra Pro (RGB-D) camera which is mounted on the top the robot as shown in figure 3.1. Besides, we record the point of interaction of the gaze direction with respect to the object where the user is looking at for two minutes. The entire process is summarized in Algorithm 1

To answer our second question, first, we have to wait until the completion of the above process. After that, we create a Gaussian Mixture Model (using

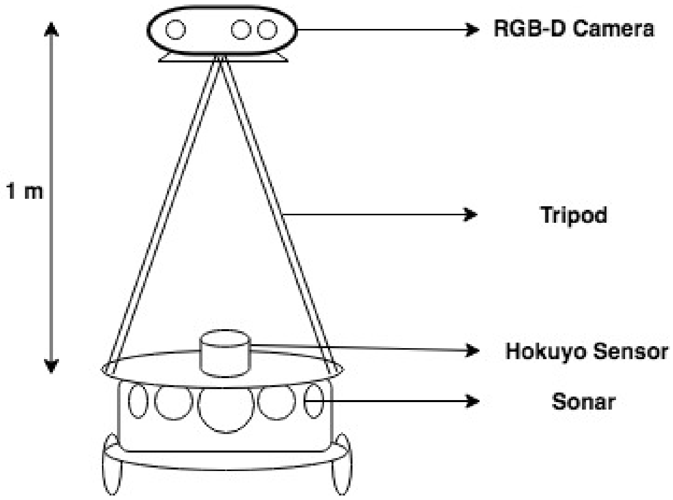

Figure 3.1: Robot Design

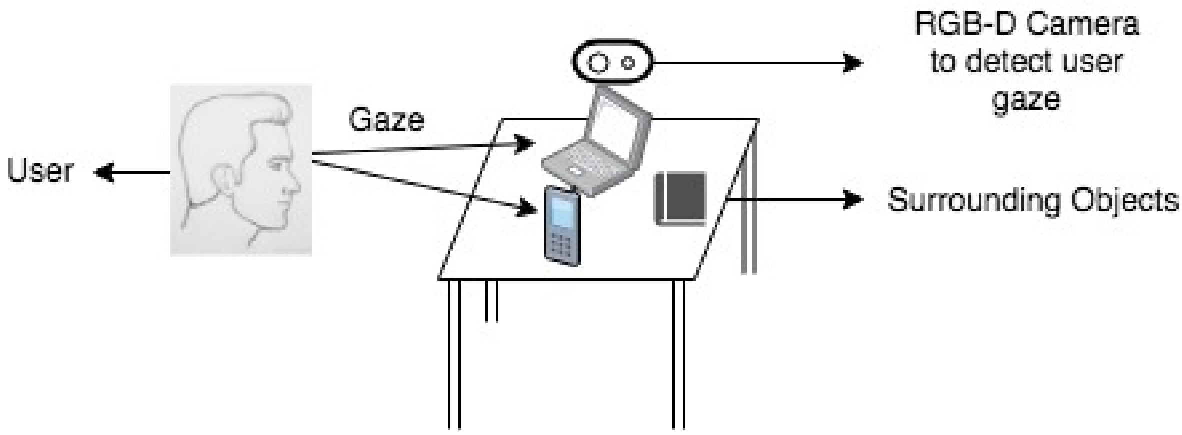

Figure 3.2: Perspective View of the Camera System

Algorithm 1 Procedure of gaze and object detection

```
Repeat
if Person = YES then
    if Record data ≤ 2 minutes then
        Record Gaze Direction POI w.r.t. Objects
    else Data has been recorded
else No person found!
```

Figure 3.3: Eye Gaze Estimation using RT-GENE

E-M algorithm) of POI with respect to detected objects. The entire process is summarized in Algorithm 2.

3.2 Structure of the system

A simplified diagram of the structure of our system is shown in figure 3.4. As you can see, our system has to perform multiple tasks in order to fulfill the goal:

- First and foremost, the robot has to be localized in order to perform any task because a robot has no idea where it is in the real world.
- After localizing the robot, it moves to the first location as shown in figure 3.5.
- Detects the user's gaze and surrounding objects where is s/he looking at
- Records the data of the point of intersection of gaze direction w.r.t. different objects for two minutes.
- The system creates a Gaussian Mixture Model to predict the attention towards an object.
- The robot checks next available cabin and move and repeat the same process until there are no more cabins available.

Algorithm 2 Gaussian Mixture Model using E-M Algorithm

1: **Repeat**

2: Randomly place Gaussians in space

$$P(x_i|O_i) = \frac{1}{\sqrt{2\pi\sigma_O^2}} e^{-(x_i-\mu_O)^2/2\sigma_O^2} \tag{3.1}$$

3: Computes the probability of a point which is generated by each model

$$P(O_i|x_i) = \frac{P(x_i|O_i)P(O_i)}{P(x_i|O)P(O_i) + P(x_i|O_k)P(O_k)} \tag{3.2}$$

4: Adjust the mean and variance of the model to fit points that are assigned.

$$\begin{aligned} \mu_O &= \sum_{k=1}^{n} \frac{O_k x_k + o_{k+1} x_{k+1} ... + O_n x_n}{O_k + O_{k+1} + ... + O_n} \\ \sigma_O^2 &= \sum_{k=1}^{n} \frac{O_k (x_k - \mu_k)^2 + ... + O_n (x_n - \mu_n)^2}{O_k + O_{k+1} + ... + O_n} \end{aligned} \tag{3.3}$$

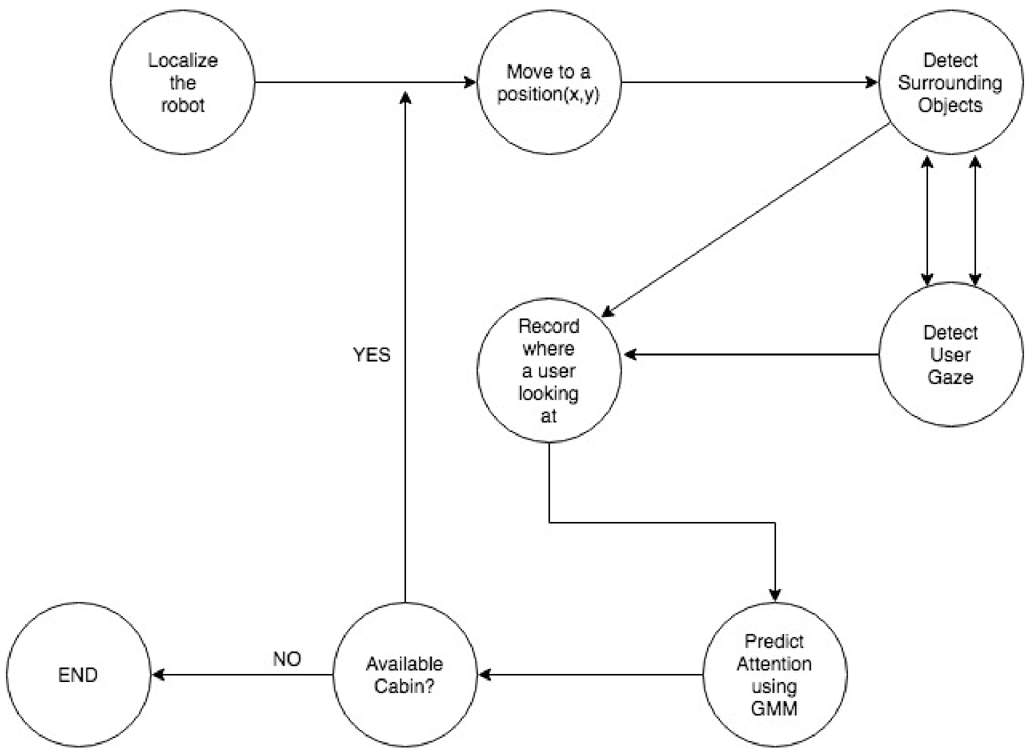

Figure 3.4: Experimental Setup

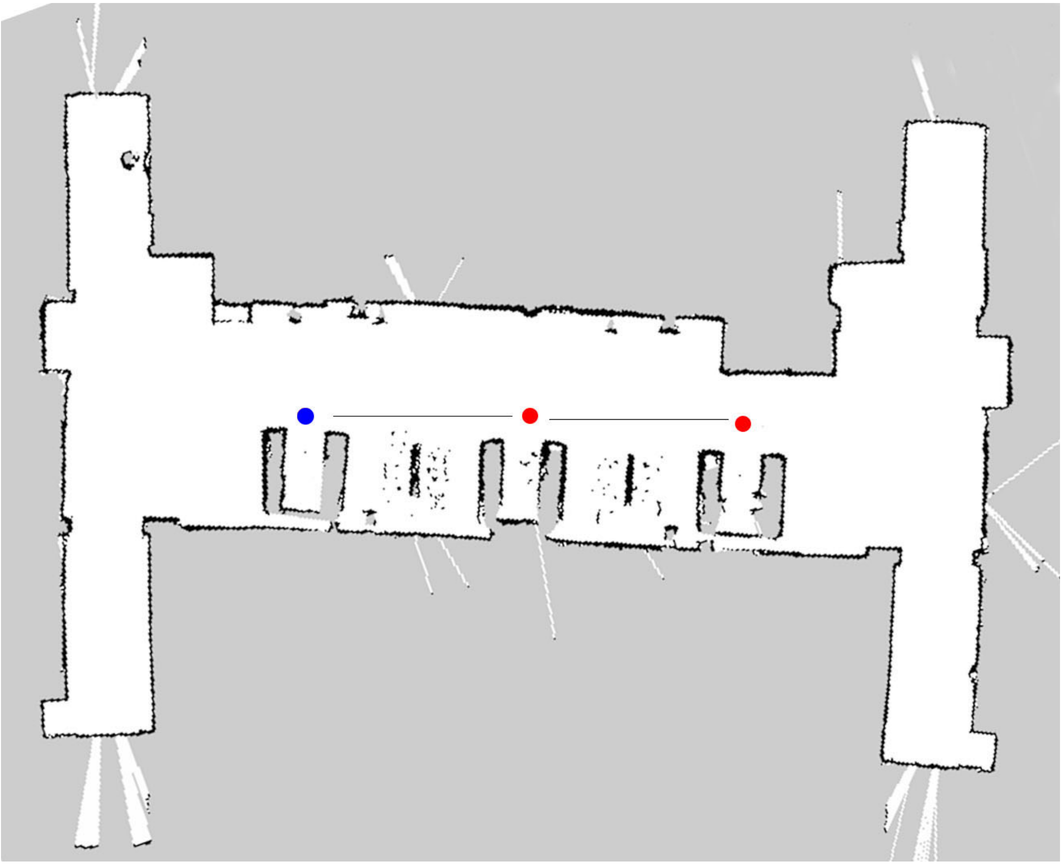

Figure 3.5: Lower Ground Floor Map

3.3 System Components

In order to perform any tasks, our system requires both hardware and software components. In this section, we talk about various aspects of our system.

3.3.1 Hardware

We have used various hardware components to evaluate our task. Our entire system includes two Astra Pro RGB-D cameras, Pioneer P3-DX robot, Hokuyo laser sensor, and A Xeon processor-based workstation with NVidia Titan X GPU which is used to run Robot Operating System (ROS) on Ubuntu 16.04.

As you can see in figure 3.1, in the center of the robot, there is a hokuyo laser sensor which allows us to gather information regarding the environment. In addition to that, we have an Astra Pro (RGB-D) camera mounted on the top of the robot which enables multiple objects detection in a scene as shown in figure 3.2. Besides, we have mounted another Astra Pro camera in front of the user to detect his gaze. The camera outputs a 720p RGB video with 30 frames per second(FPS) and also depth images. It has a horizontal and vertical field of view of 60 and 49.5 degrees respectively.

3.3.2 Software

Robot Operation System (ROS) is an open-source library consist of different services which allow low-level device control. ROS is available for multiple operating systems such as Windows, Linux, MacOs but it is recommended to use on Linux. ROS allows users to write simple task-based programs known as a node to control a robot instead of traditional complex programming. A node is used to transfer information between multiple nodes through topics with a specific structure.

3.4 Localization

The primary purpose of localization is to estimate the position of a robot in the real world by using laser scan readings. In our system, we have used probabilistic localization system Adaptive Monte Carlo Localization (AMCL) instead of tradition Monte Carlo Localization (MCL). The reason for using

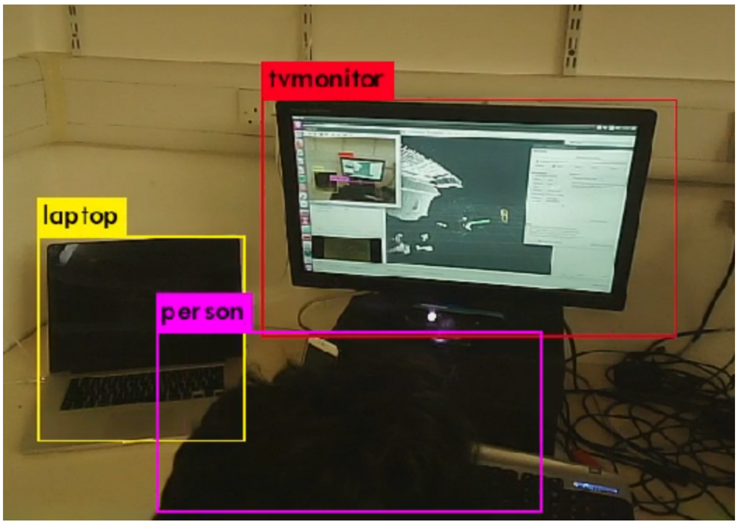

Figure 3.6: Object Detection using YOLO

AMCL is due to the higher estimation accuracy of a robot position and low tracking error.

3.5 Path Planning

Similar to AMCL, path planning is also based on a probabilistic technique. We have used ROS provided navigation module for path planning due to its robustness and higher performance for completing a task. The navigation module computes a path between an initial and a goal position.

3.6 Object Recognition using YOLO

We have used YOLO object detection as mentioned in section 2.1 to detect multiple objects in a scene as shown in figure 3.6 in order to complete the task of estimating the attention of a user via gaze direction. Reliable object detection is essential to understand the surrounding environment. A single Astra Pro (RGB-D) camera has been placed on the top of a robot to perform this task as shown in figure 3.1. Despite the availability of Kinect v2, in our case, we chose the Astra Pro camera due to its reliability.

YOLO models detection as a regression problem. It divides an input image into multiple grids. These grids predict a bounding box for each cell with their confidence score as shown in figure 3.1. Furthermore, YOLO can predict as much as 80 object classes while using common objects in the context (COCO) dataset (28). In our case, we chose seven objects which include a person, laptop, tv monitor, cup, cell phone, bottle, and book because of the office environment. Beside that, The network architecture consists of 24 layers of convolutions and two fully connected layers.

3.7 Real-Time Eye Gaze Estimation

As mentioned in section 2.2, we have used a real-time appearance-based gaze estimation (RT-GENE) (29) to detect a user's gaze direction. In order to successfully detect the gaze direction of a user, we have placed an Astra Pro camera in the front of the user(figure 3.2). The reason for using RT-GENE is that it uses information of face orientation with eye region which is robust as mentioned earlier in section 2.2.

How RT-GENE (29) predicts human gaze:

- Multi-Task Cascaded Convolutional Networks (MTCNN) (30) is used for face detection with multiple landmarks points such as nose, eyes, and mouth.
- The normalized face image is obtained through the accelerated iterative closest point algorithm (31).
- The normalized face images are used to extract eye patches.
- Performed feature extraction on the eye patches using VGG-16 Network (32).
- The VGG-16 network is then is followed by fully connected (512 layer size) then max-pooling, batch normalization and then ReLU activation.
- Followed by another fully connected (512 layer size)
- Append the head pose vector to the fully connected layer followed by 256 and two size fully connected layers.
- Eye gaze angles such as yaw and pitch are returned

3.8 Gaussian Mixture Model

The primary purpose of the Gaussian Mixture Model (GMM) is to fit the mixture of Gaussian models. Due to the unlabeled data, GMM does not know which point comes from which generative model. To solve this issue, we have used the Expectation-Maximisation (E-M) algorithm which uses an iterative process.

We have used GMM to fit the mixture of Gaussians of different object classes (discussed in section 3.2.3) that have been detected by YOLO and the point of interaction between gaze direction and the object bounding box. By using GMM, we can predict the interest point of a user in a scene.

Chapter 4

Experimentation

4.1 Accuracy of Attention Prediction

This experiment aimed to check the accuracy of user attention where s/he looking at in a known environment. To evaluate the system performance, we conducted 15 tests by five users. First, we collected the data for two minutes from a user's gaze with respect to different objects such as cell phone and book. After that, we found the interest point (on which object the user was focused the most), and by using GMM (figure 4.2), we predicted the attention of the user in a given scenario of a work environment.

On average, the system was able to predict the attention of a user 11 out 15 times and as we can see from table 4.1 the error between actual and predicted interest point where a user was looking at was 26.67%. In addition to that, the system failed 13.33% to perform the task.

4.2 Performance of Path Planning Algorithms

The aim of the experiment was to find the optimal algorithm for a global planner which can be used for path planning. In this experiment, we have checked the performance between two algorithms: Dijkstra and Navfn for optimal path planning. As shown in figure 3.5, we took readings at the tar-

Table 4.1: Error rates between actual and predicted interest point in 15 tests

Task	Error
Actual vs Predicted Interest Point	26.67%

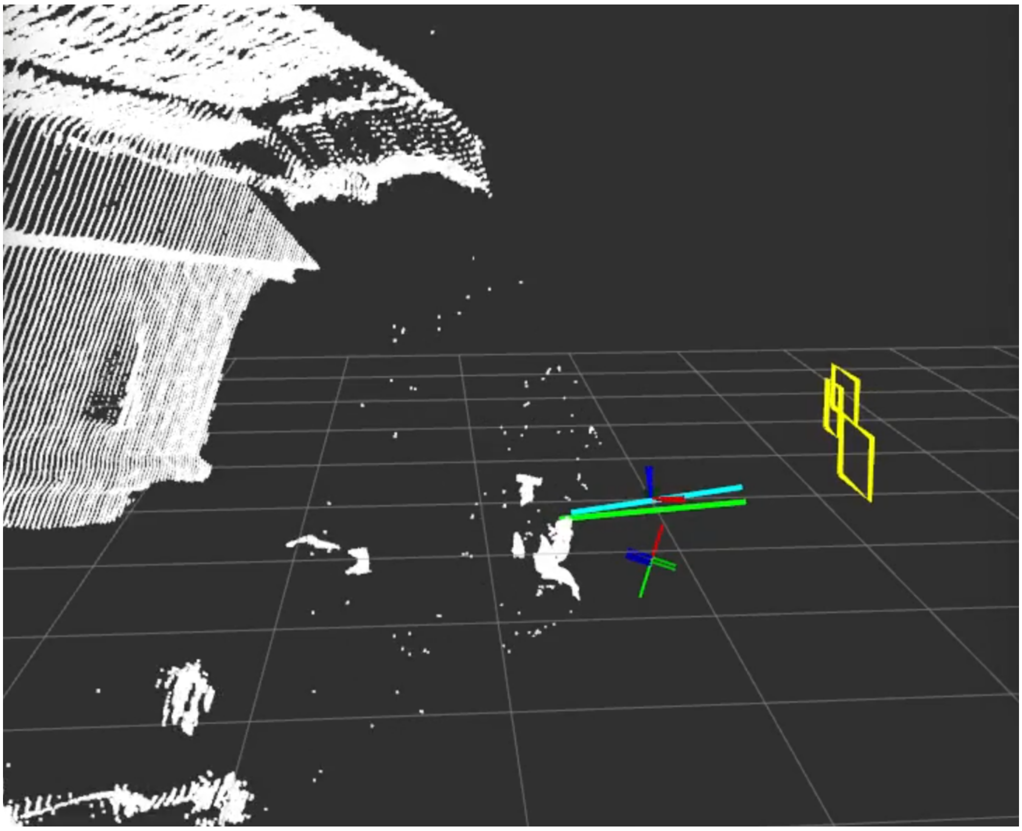

Figure 4.1: Data Collection from Gaze w.r.t. Multiple Objects

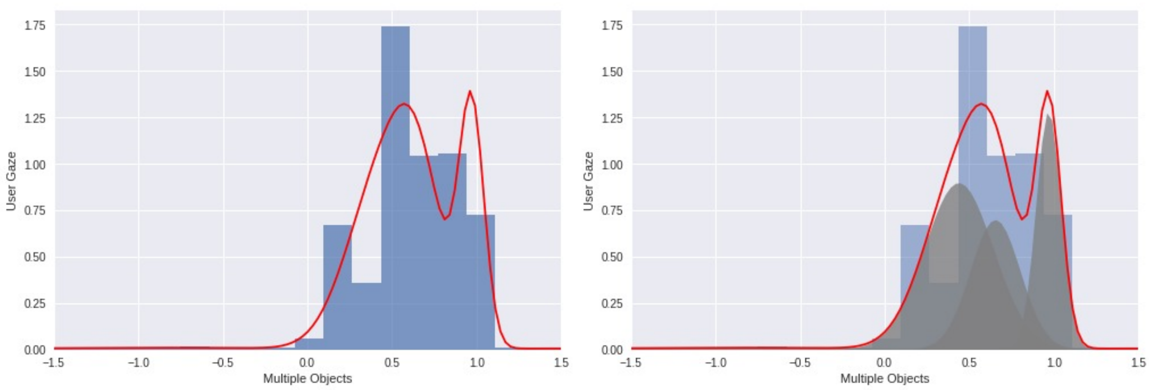

Figure 4.2: Gaussian Mixture Model of a user gaze w.r.t. multiple objects

Table 4.2: System failure rates of different global planners

Global Planner	Rate
Dijkstra	0.06%
Navfn	20.00%

get position on a predefined path. The starting and ending position of the robot is marked with the blue and red circle respectively. After that, the robot was instructed to reach the target location. This task was performed for 15 times for each algorithm.

As we can see from table 4.2, the system failed to reach the target location 20% and 0.06% while using Navfn and Dijsktra algorithm respectively.

Chapter 5

Discussion & Conclusion

After running multiple tests, we concluded that the system could recognize a user's attention. During the development, we have faced a range of difficulties. Occasionally, the laser sensor data was inaccurate due to that we have added additional noise to the position of the robot. Also, multiple times we faced the camera shut down because of our workstation data transfer (through USB) limit.

5.1 Improvement

Although the proposed system is robustly built and performed each task well, the efficiency and functionality can be improved in different ways. Due to the computational demand of various components such as YOLO, and RT-GENE, the overall performance of the system was quite slow. In the future, we would like to optimize our system for different hardware by using CUDA programming.

Aside from the hardware, In order to utilize the full potential of the system, instead of gathering information about user perception for a few minutes, we would like to predict the attention in real-time. Also, we can provide gesture-based controls like pointing or waving to the system which can enable more robot interactions.

5.2 Conclusion

To recapitulate, In this work, a robotic system is proposed which can predict user attention. An appearance-based eye gaze estimation and object detection method have been used to achieve the desired goal. The robot has shown an effective way of combining both gaze estimation and object detection to understand where a human is looking at. Furthermore, the robustness of the system has been checked through multiple sets of user experiments which allowed us to improve the entire system.

Bibliography

[1] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[2] Carlos H. Morimoto and Marcio R.M. Mimica. Eye gaze tracking techniques for interactive applications. *Computer Vision and Image Understanding*, 98(1):4–24, 2005.

[3] Hyun Soo Park, Eakta Jain, and Yaser Sheikh. Predicting primary gaze behavior using social saliency fields. *2013 IEEE International Conference on Computer Vision*, 2013.

[4] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[5] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[6] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[8] Dan Witzner Hansen and Qiang Ji. In the eye of the beholder: A survey of models for eyes and gaze. *IEEE transactions on pattern analysis and machine intelligence*, 32(3):478–500, 2010.

[9] Jinsoo Choi, Byungtae Ahn, Jaesik Parl, and In So Kweon. Appearance-based gaze estimation using kinect. *2013 10th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, 2013.

[10] Feng Lu, Takahiro Okabe, Yusuke Sugano, and Yoichi Sato. Learning gaze biases with head motion for head pose-free gaze estimation. *Image and Vision Computing*, 32(3):169–179, 2014.

[11] C.H. Morimoto, A. Amir, and M. Flickner. Detecting eye position and gaze from a single camera and 2 light sources. *Object recognition supported by user interaction for service robots*.

[12] S.-W. Shih and J. Liu. A novel approach to 3-d gaze tracking using stereo cameras. *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, 34(1):234–245, 2004.

[13] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. Appearance-based gaze estimation in the wild. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[14] Erroll Wood, Tadas Baltruaitis, Xucong Zhang, Yusuke Sugano, Peter Robinson, and Andreas Bulling. Rendering of eyes for eye-shape registration and gaze estimation. *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015.

[15] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Joshua Susskind, Wenda Wang, and Russell Webb. Learning from simulated and unsupervised images through adversarial training. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[16] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. It?s written all over your face: Full-face appearance-based gaze estimation, 2018.

[17] Qiong Huang, Ashok Veeraraghavan, and Ashutosh Sabharwal. Tabletgaze: dataset and analysis for unconstrained appearance-based gaze estimation in mobile tablets. *Machine Vision and Applications*, 28(5-6):445–461, 2017.

[18] Yusuke Sugano, Yasuyuki Matsushita, and Yoichi Sato. Learning-by-synthesis for appearance-based 3d gaze estimation. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[19] Meng-Che Chuang, Raja Bala, Edgar A. Bernal, Peter Paul, and Aaron Burry. Estimating gaze direction of vehicle drivers using a smartphone camera. *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2014.

[20] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. Mpiigaze: Real-world dataset and deep appearance-based gaze estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[21] Lefteri H Tsoukalas and Dimitrios T Bargiotas. Modeling instructible robots for waste disposal applications. In *Intelligence and Systems, 1996., IEEE International Joint Symposia on*, pages 202–207. IEEE, 1996.

[22] Roland Siegwart, Illah Reza Nourbakhsh, Davide Scaramuzza, and Ronald C Arkin. *Introduction to autonomous mobile robots*. MIT press, 2011.

[23] Benjamin W Tatler, Roland J Baddeley, and Benjamin T Vincent. The long and the short of it: Spatial statistics at fixation vary with saccade amplitude and task. *Vision research*, 46(12):1857–1862, 2006.

[24] Laura Walker Renninger, Preeti Verghese, and James Coughlan. Where to look next? eye movements reduce local uncertainty. *Journal of Vision*, 7(3):6–6, 2007.

[25] Benjamin W Tatler, Nicholas J Wade, Hoi Kwan, John M Findlay, and Boris M Velichkovsky. Yarbus, eye movements, and vision. *i-Perception*, 1(1):7–27, 2010.

[26] Wolfgang Einhäuser, Merrielle Spain, and Pietro Perona. Objects predict fixations better than early saliency. *Journal of Vision*, 8(14):18–18, 2008.

[27] Kiwon Yun, Yifan Peng, Dimitris Samaras, Gregory J Zelinsky, and Tamara L Berg. Exploring the role of gaze behavior and object detection in scene understanding. *Frontiers in psychology*, 4:917, 2013.

[28] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[29] Tobias Fischer, Hyung Jin Chang, and Yiannis Demiris. RT-GENE: Real-Time Eye Gaze Estimation in Natural Environments. In *European Conference on Computer Vision*, September 2018.

[30] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, Oct 2016.

[31] Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Sensor Fusion IV: Control Paradigms and Data Structures*, volume 1611, pages 586–607. International Society for Optics and Photonics, 1992.

[32] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Appendix A

System & Software Requirements

- Workstation with powerful GPU (NVidia 1080Ti Recommended)
- Two RGB Cameras
- Operating System: Ubuntu
- Robot Operation System (ROS)

Appendix B

How to run the code