Merging human-object interaction behavior into a personal space model for social robot navigation

Bai, L.

Award date:
2018

Department of Industrial Engineering and Innovation Science
Human-Technology Interaction

Merging Human-Object Interaction Behavior into a Personal Space Model

For social robot navigation

Lang Bai

Supervisors:
Raymond Cuijpers
Gijs Dubbelman
Margot Neggers

Eindhoven, November 2018

Abstract

Creation of a personal space model and implementation in navigation algorithms constitute robot navigation studies in human-robot interaction filed. However, social awareness for assisting robot's navigation can be a challenge due to the lack of human activity detection and intention prediction. To solve this problem, the present work builds a personal space model based on the recognition of human-object interaction, objects and human, called activity-based personal space model. The present work first discusses the configuration of activity-based personal space, and then implements the detection algorithm separately, and a new integrated detection algorithm model has been proposed to provide the inputs for activity-based personal space model. Our activity-based PS model, which involving activity space, affordance space, and blind area has been visualized. And the simulation has been done for a technical evaluation of the motion trajectory and force equation. The present work lays the foundation of assisting robots which can detect human intention and updates its motion planning.

Contents

Contents **5**

List of Figures **7**

List of Tables **9**

1 Introduction **11**

2 Methods of building PS models **13**

2.1 Personal space 13

2.2 Personal space for groups: F-formation Model 14

3 Configuration of activity-based PS model **17**

3.1 Activity Space and Blind Area 17

3.2 Affordance Space 18

3.3 Towards a Dynamic Context Dependent Personal Space 19

4 Methods for detecting human attention **21**

4.1 Head orientation (gaze direction) 21

4.2 Body orientation 22

4.3 Pose estimation 22

4.4 Object Detection 22

4.5 Human-Object Interaction Detection 23

5 Implementation of detection **25**

5.1 A Proposal of Detection Algorithm Structure 25

5.2 Indoor-activity-scene Dataset 26

5.3 Head Orientation 28

5.3.1 Head Orientation Detection Result on Head Pose Image Database 28
5.4 Body Orientation Estimation 30
5.4.1 Body Orientation Detection Result on Indoor-activity-scene Dataset 30
5.5 Pose Estimation 32
5.5.1 Pose Estimation Result on Indoor-activity-scene Dataset 34
5.6 Object Detection and Result on Indoor-activity-scene Dataset 34
5.7 Human-Object Interaction Recognition and Result on Indoor-activity-scene Dataset 37

6 Results **39**
6.1 Individual-person PS model 39
6.2 Activity and Affordance Space 40
6.3 Blind Area 42
6.4 Activity-based PS Model Visualization 44
6.5 Application and Navigation 46
6.5.1 Simulation 46
6.5.2 Simulation with Noises 46

7 Discussion **51**
7.1 Stopping Distance 51
7.2 The Influence of Human-Object Distance 52
7.3 Robustness in Navigation Algorithm 53
7.4 Navigation Results on Using a Laptop 55

8 Future work **57**

9 Conclusions **58**

References **59**

List of Figures

1.1 Overview of the framework of activity-based social navigation. We have highlighted in red our contributions to advance the topics discussed later in the present work, red boxes means new model or new application in HRI. 12

2.1 Hall's Personal Space 14

2.2 Kendons F-Formation of the Face-to-Face type 15

3.1 Activity space and blind area of watching TV (top view) 18

3.2 Activity space and blind area of watching TV (side view) 18

3.3 the chair's affordance space equals the activity space when the person has intention to sit or pass by 19

3.4 the chair's affordance space equals the area to avoid collision when the person has no intention to sit 19

5.1 Overview of Our Proposed Detection Approach 26

5.2 The pictures from our test dataset 27

5.3 Head Orientation Estimation: red dots represent the points chosen for orientation estimation, the blue line represents the estimated head orientation. 29

5.4 Sample pictures of the head pose database 29

5.5 The inference results of body orientation detection 31

5.6 The spine-leg angel 33

5.7 The failure examples of pose estimation: the left is detected as sitting with a spine-leg angle of 162 degree; the right is detected as "unknown" with none of body parts is recognized. 34

5.8 The results of object detection 36

5.9 The results of human-object interaction detection 38

6.1 The probability of individual-person PS model: human is located at (20, 20) as a red circle, facing angle is 60 degrees; the TV is placed at the polar coordinates (60, 3) with respect to the users reference frame, with width of 90 cm and height of 125 cm. 40

6.2 The left image shows a Nao robot stops just next to the border and in the personal space; the right image translates the situation shown in the left image into a mathematical problem to derive the value of ψ 41

6.3 The Probability of Activity Space Model Changing Along with Angular Positions . 42

6.4 The Probability Distribution of the Blind Spot from Side-view 43

6.5 The Probability of the activity space and blind area when a human is watching a TV 43

6.6 the activity-based personal space model of watching TV: (1) the image shows the probabilistic model where the person wants the robot to stop; (2) the red line presents the TV, the person locates at [35, 70] 45

6.7 The probability cross section of old model and new model 45

6.8 How the strength of the weight expression change along with the distance between the robot and particle 47

6.9 The trajectory of approaching from the right point based on the new model 47

6.10 The top view of the world when the robot is located at [90,75] 48

6.11 The probability distribution of approaching from the right point based on the new model 48

6.12 The probability density function of normal distribution. From "Standard deviation diagram" by J.Muelaner, 2013, http://www.muelaner.com/metrology/. 49

6.13 Trajectory of the robot's approaching the human from three sides. In these simulations, human body orientation is randomly generated from a normal distributions with mean -35 or 0 or 35 degrees, and variance 22.5 degree during the frames update. 50

7.1 The force brought from the human-object distance 53

7.2 The trajectory of approaching from the middle point based on old model and new model 54

7.3 The target force and obstacle avoidance force changing during the navigation starting from the middle point based on old model 54

7.4 The trajectory and forces changing based on old model after improving the obstacle avoidance algorithm 55

7.5 The trajectory of approaching from the middle point based on old model and new model when a human is using a laptop 56

List of Tables

5.1 Head orientation detection evaluation result on head pose database, yaw success rate (YSR) and pitch success rate (PSR) are presented 29

5.2 Body orientation detection evaluation result on indoor-activity-scene dataset, yaw success rate (YSR) is presented based on human pose 31

5.3 Pose estimation evaluation result on different poses 34

6.1 Summary of the parameters of the approximating curves that relate distance (γ) and preference (η) to direction of approach for the two experiments used in individual-person PS model 39

Chapter 1

Introduction

With the growing interests in the assistive robots, social navigation has become the key feature for a robot to be used in the real-life application. When the robots and humans share the same physical space, a social robot should obey similar social rules as humans, otherwise, the robot might harm people or itself (Chung & Huang, 2010). In particular, when approaching people, a robot should respect human proxemics. The term, proxemics, first introduced by Hall (1966) to describe human space management, which is mainly constituted by the concept of personal territory. There are four kinds of spaces with different distances that people generally use in social interaction behavior: intimate space, personal space, social space, and public space. Many previous studies (Walters, 2008; Torta, Cuijpers, Juola & van der Pol, 2011; Obaid et al., 2016) found that the human prefer to interact with a robot which is located in personal space area. For a robot proxemics behavior in HRI, a probabilistic model named personal space model is built to provide the likelihood of people expecting the robot to stop at for each position. Previously, robot proxemics behavior was modelled based on an appropriate distance and direction of approach, when humans are standing or sitting in an open space Torta2013. However, in real life, people employ many different activities. For example, when people are watching TV, stopping between human and television may be regarded as impolite. When people are using a laptop, standing in the area right behind the laptop might not be noticed by the human. In other words, the robots should recognize the so-called activity space and affordance space, which are representing spatial regions related to the current action and the potential action, respectively (Rios-Martinez, Spalanzani & Laugier, 2014).

In this work, we define the approaching a human and stopping task in the context of an assisting robot approach a human in a living room, where the human can be watching a TV or using a laptop in different poses (sitting or standing), while the robot does not know the human status at the start, and needs to somehow infer it, to be able to approach the human in a socially acceptable way. This is might be a simple task for an adult. From human tuition, when one approaches another person, he consistently perceives that persons status changing to adjust his approaching strategies, e.g., if this person switch his(her) sight from the TV downward to the phone, he might step closer and make some sound to draw the persons attention; if this person puts his(her) hands on the keyboard of a laptop and looking at the screen, he could stop behind the laptop and draw his(her) attention, no need to worry about being in the way of watching the TV.

However, it is a challenging problem for robot to tackle, which requests a semantic understanding of human activities and the prediction of human potential activity, moreover, the human orientation and pose should be detected real-time. However, with the development of computer vision recently, to point out the most related and recent literatures, e.g., getting human skeleton information from one RGB camera (Cao, Simon, Wei & Sheikh, 2016), human-object interaction detection from one

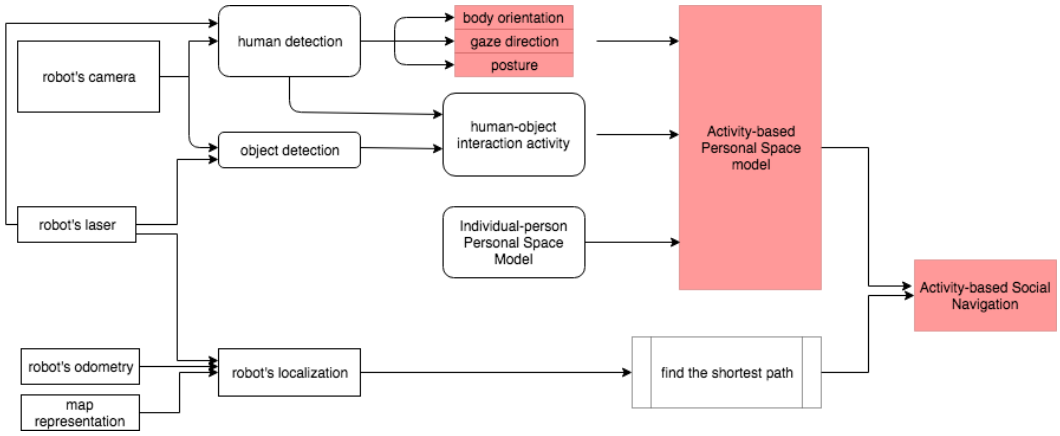


Figure 1.1: Overview of the framework of activity-based social navigation. We have highlighted in red our contributions to advance the topics discussed later in the present work, red boxes means new model or new application in HRI.

single camera (Gao, Zou & Huang, 2018), real-time depth prediction in dense monocular SLAM (Tateno, Tombari, Laina & Navab, 2017), to extract that semantic features from a single image become a less tough problem.

In the field of human-robot proxemics behavior, robot motion planning is modelled by creation of a personal space model and implementation in navigation algorithms. Since personal space model provides probabilistic distribution of the world in line with human preference, we could assume that a personal space model which concerns human-object activities provides the foundation of a socially acceptable navigation. Then what constitutes a activity-based personal space model? Figure 1.1 shows the components required for activity-based personal space model for activity-based social navigation and where the present study has made contributions.

In the present study, we have extended the previous individual-person PS model built by Torta, Cuijpers, and Juola (2013) by using the state-of-art computer vision approaches to extract semantic features. We developed a activity-based personal space model that takes the observed activity of a person and objects in the environment into account. We combined the human orientation estimation, pose estimation, object detection, and human-object interaction detection algorithm to propose the method of perceiving the human activities and the objects they are currently interacting with. In the line with the human proxemics behavior in human-human interaction, robots proxemics behavior model was modified considering real-time human activities and spatial information. To refine the overall navigation behavior based on our model personal space model, we discuss the influences of different finding the shortest path forces. The model was validated through simulations.

In the first part of our study, we implemented the computer vision algorithms relevant to human detection, object detection, and human-object interaction detection, propose the framework to integrate the algorithm. Then we visualized our activity-based personal space model, and validated the activity-based personal space model in contrast with the individual-person personal space model. Finally we discussed the superiority of activity-based PS model in the complex real-life environment.

Chapter 2

Methods of building PS models

2.1 Personal space

From the view of human-robot interaction, how people think and behave should be the key when the robot is approaching people. According to Hall (1966), the personal space around a person as an invisible concentric space. The space is classified into four sub-spaces with respect to social interaction behavior. Each social interaction behavior is along with a specific distance that the person feels comfortable (see Figure 2.1). As Hall claims, the intimate space is for lovers or close friends. People touch, hug each other, and have intimate conversations in this space. In personal space, people interact with well-known ones. People have a conversation in public and interact with non-friends in social space. Usually, people ignore others in public space since they treat them as public entities only. Based on Halls work, further researches were carried out, the shape of personal space (Hayduk, 1981), the spatiotemporal model of personal space (Park & Trivedi, 2007).

In addition to Hall's work, Torta, Cuijpers, and Juola (2013) propose the parametric model of personal space based on an appropriate distance and direction of approach, when the human is with standing or sitting posture in an open space. However, in real life, the activities of human should be taken into account. Most human activities can be regarded as human-object-interaction activities, namely, engaged with at least one object. The object could be touched by the human, like using a phone, reading a book; the object can also be quiet close, but retaining a certain distance, like using a laptop, talking with someone; the object can also be relatively far from the human, like watching TV, reading a board. However, The activity space does not have an explicit definition of shape, as claimed by Rios-Martinez, Spalanzani, and Laugier (2015), the shape depends on the specific action.

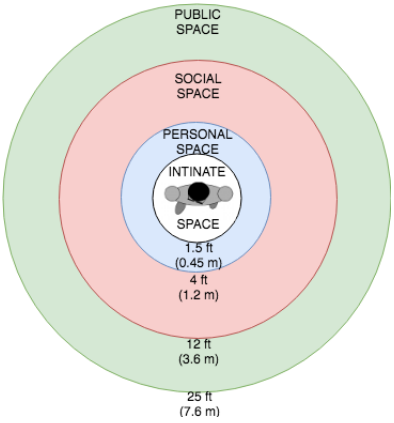


Figure 2.1: Hall's Personal Space

2.2 Personal space for groups: F-formation Model

Kendon (1990) investigated how people position themselves in groups for making conversation. He found people always establish and maintain a convex space in a socio-spatial formation, named F-formation. An F-formation consists of three regions: transactional region (o-space), agent-region (p-space), and buffer region (r-space) (see Figure 2.2). An F-formation is formed when the transactional segments of two or more people overlap, a joint transactional space formed an o-space to where the interaction actually takes place. The agent region is where the agents are located at. The buffer region is the boundaries of the human-human conversation area, separated from the environment, which is not directly used by the participants in the F-formation, but it is monitored by them whether anyone is approaching to join in the conversation. Though F-formation is designed for face-to-face interaction between two people or more, if we replace one human by a robot, does the F-formation system exist in human-robot interaction? The answer is affirmative: When people interact with robots, they follow the configuration of F-formation to arrange space (Hüttenrauch, Eklundh, Green & Topp, 2006). In the report of Kuzuoka, Suzuki, Yamashita, and Yamazaki (2010), they found out people use body orientation and gaze to control F-formation. The usage of F-formation system is for people to create and maintain this o-space (Kuzuoka, Suzuki, Yamashita & Yamazaki, 2010), for example, when a new agent is going to join the conversation, people can increase the distance between participants and step to the side to create the new o-space. From the F-formation's point of view, the scenario of a robot approaches a human who is engaging in a human-object interaction to start a conversation is to build a two-person o-space between human and the robot. Semi-separated from the human-object interaction, the human-robot interaction is supposed to be described as one-person personal space (Hall's model), as well as not arouse the discomfort of the person to switch interactive agent spatially (from the object to the robot).

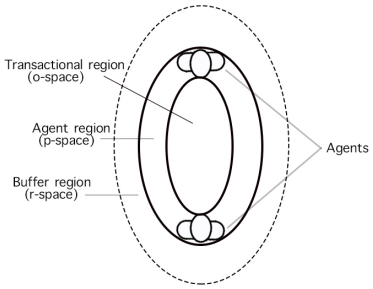


Figure 2.2: Kendons F-Formation of the Face-to-Face type

Chapter 3

Configuration of activity-based PS model

When a human is performing an activity (mostly human-object interaction activity), many spaces are related to the human and object. Rios-Martinez, Spalanzani, and Laugier defined two space related to activities: activity space and affordance space (Rios-Martinez et al., 2014). Activity space is defined as a social space consisting of the current human activity, like the space between a woman and the object of which she is taking a photo. Affordance space is defined as a space which is related to potential human activity, which is a potential activity space, like the space in front of a bus schedule where a human is supposed to stand. Therefore, socially acceptable robot navigation should build a personal space model concerning activity space and affordance space.

3.1 Activity Space and Blind Area

One type of activity space was introduced under the activity footprints (Ostermann & Timpf, 2007), which are used to estimate the space occupation in public parks. Activity footprints can well define the space consumed by some activities: ball games constitute the space needed for playing balls, jogging constitutes the space needed for jogging, etc. However, the space consumed by some activities can not be quantitatively measured by footprint: reading or chatting. On a qualitative level, an activity space model described by Kendon (1990) is widely accepted. Before introducing his concept of F-formation, he first claims that every individual in solitary activity has a space called a transactional segment. It's a sector of the environment in front of an individual, where most activities can be located, and where people have the greatest degree of control over their environment through eyes, head, or upper trunk. The size of this transactional segment space varies according to the activity in which people are engaged, e.g., watching TV versus writing. As Kendon described, "a man sitting over a book has a narrow, highly circumscribed transactional segment. A man sprawled on a sofa watching television has a wide transactional segment that extends at least as far as the television set." (Kendon, 1976). Thus, the concept of the transactional segment has already included human-object interaction activity. Taking the activity of watching TV for example, we visualized the activity space and the space behind objects (blind area), see Figure 3.1 and Figure 3.2.

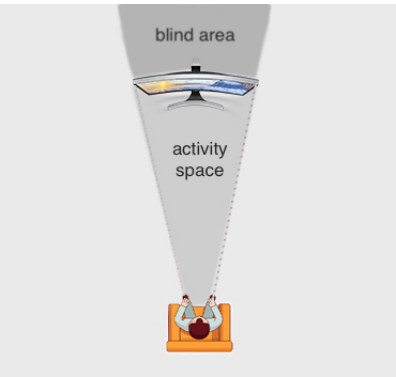


Figure 3.1: Activity space and blind area of watching TV (top view)

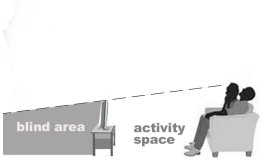


Figure 3.2: Activity space and blind area of watching TV (side view)

3.2 Affordance Space

It is worth noting here, activity space only exists when the activity is taking place. It is short-lived since human activity changes over time. As for an object, there is a space related to the potential activities for agents, called affordance space. Lindner and Eschenbach (2011) claimed it is crucial to distinguish the activity spaces from affordance spaces, since it is generally forbidden to cross an activity space while not that problematic to cross an affordance space. To support their statement, they used the space in front of a light switch as an affordance space example. In their another paper, they claimed affordances of an object can be regarded as traits, which can exist independently of activities (Lindner & Eschenbach, 2014). In most real-life indoor environments, there are multiple objects. Each object is coded with its affordances, books in the bookcase afford to view and to pick up for the human. However, the configuration of affordance space of one object is not eternal. To take an empty chair for example, when the human is sitting on the couch and watching at a TV, the chair beside may afford sitting on for this human. However, this potential action may not take place in a short time, then the robot should only take this chair into account as an obstacle when approaching (see Figure 3.4). Alternatively, when a standing person is walking towards or gazing at the empty chair, it is likely that (s)he intends to sit on this chair. Then the robot should take the affordance space of this chair into account and extend it to the transactional segment of this human (see Figure 3.3). We will refer to this space as 'activity space'.

We can apply our criteria to the light switch example of Lindner and Eschenbach (2011). If people are watching a TV in the living room, with no sign showing they are about to turn on the light. In this case the robot can safely ignore the affordance space and only regard it as an obstacle. The distinction is whether an object is under the state of interacting with the human i.e. part of a persons activity space. The challenge is for a robot to recognize when a person is likely to interact with such objects, so that the proper activity space is triggered. Supposing a huge bookcase, the

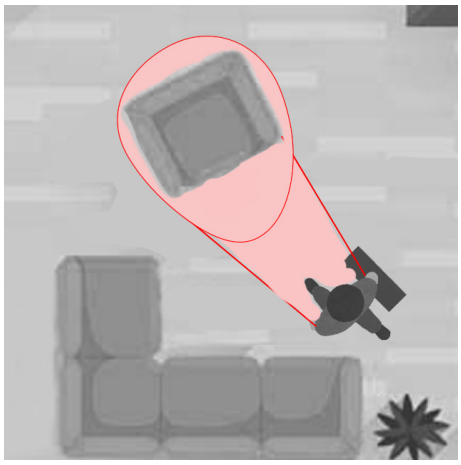


Figure 3.3: the chair's affordance space equals the activity space when the person has intention to sit or pass by

Figure 3.4: the chair's affordance space equals the area to avoid collision when the person has no intention to sit

affordance space of this bookcase is the whole area in front of it. If a human is searching for a book, the area between human and bookcase is the activity space. When this human is moving slowly, the space between human and bookcase always contains as an activity space. At the same time, a robot could approach this human crossing through the affordance space, but never the area between the human and the bookcase. In an idle state, the robot could stay in the affordance space for a while, once this space where the robot standing turns into an activity space, the robot should move away. Therefore, the status of the affordance space, passable or not passable, affect the robot behavior in this area, e.g., whether to stop in this area, whether to cross through this area, whether to stay long in this area. Since each object is associated with an affordance space, it is important to locate and determine the status of the affordance space in the creation of a personal space model.

3.3 Towards a Dynamic Context Dependent Personal Space

Most effort in mobile robotics has been spent on geometric navigation, which is navigation based on geometrical information of the entities in the environment. Given a start point and an endpoint, a purely geometric navigation could lead to an infinite number of solutions. Therefore, some constraints like the minimum length or the minimum energy consumption are imposed on determining the trajectory. However, Borkowski, Siemiatkowska, and Szklarski (2010) claimed the geometric navigation is insufficient because a social navigation raises higher demand for linking the semantic information to the entities than pure geometry (Borkowski, Siemiatkowska & Szklarski, 2010). For a social aware robot, it is important to perceive the intention of a human, in other words, to understand the current human activities and predict the potential activity. This ability gives robots more "intelligence". Reflected in the area of motion planning, apart from the geometrical information, detection methods should provide more semantic information link to the objects, human, and the scene. In other words, intelligent robots could recognize human, objects, and the current human activity then determine the appropriate activity space, and right affordance space. Thus, object detection, human detection, and human-object interaction detection can bring valuable information to navigate the robot.

As described in the Chapter 3.2, human (upper) body orientation and gaze direction is associated with activity space and affordance space. According to Ciolek and Kendon (1980), the eyes total

range of scanning is up to +/-35 degrees, while this range can be increased by +/-80 degrees by rotating the head and neck. Besides, the rotation of the upper body can increase the range by +/-90 degrees on the left or right side, without moving the hips and legs. Since we chose the point with highest probability in personal space model as the target endpoint for the robot, human gaze direction is important for determining the target point in semantic(social) navigation. In the meanwhile, the target point highly affect the trajectory chosen in geometric navigation, e.g., we chose the shortest trajectory length. Besides, the last phase of approaching a human should be angular alignment with the human body orientation, namely, facing to the human. Thus, to recognize the body and head orientation in such a wide range is essential for robot to approach.

Apart from body orientations, personal space is also affected by the posture. Ciolek and Kendon (1980) claimed that the size of transactional segments varies with the posture. "When people stand with their faces close to a wall or lie down on their stomachs, their transactional segments can be said to be considerably reduced in size or even 'switched off' entirely."(Ciolek & Kendon, 1980). The investigation of Torta et al. also supported the statement of Ciolek and Kendon (1980), in the result of their study, the stopping distance varies when subjects are sitting and standing. Since the optimal stopping distance of the humans expectation also changes with the human postures, as a result, human postures have an effect on the endpoint and the probability distribution of the whole space. Therefore, to detect the human posture, and build a personal space model to cope with the dynamical human posture status could facilitate robot navigation.

To sum up, to estimate a dynamic personal space model which alters along with human behavior, we should consider human body orientation, gaze direction, and posture, as well as the current human behavior (human-object interaction behavior).

Chapter 4

Methods for detecting human attention

To detect what a person is attending to is in general very difficult as people can covertly attend to something while their overt behaviour is showing something else. Tracking overt attention is much simpler as people usually look at the object/person of interest. The tracking of attention then boils down to following the line of gaze and decide what is the focus of interest. Practically, this is still very hard as human gaze is changing rapidly, and the reconstruction of the 3D gaze direction from images is difficult at best. Here we present an overview of recent methods for head orientation, body orientation, object detection, and human-object interaction detection.

4.1 Head orientation (gaze direction)

Head orientation is intrinsically linked with gaze direction estimation, which infers the direction of current human attention. Observing a persons head orientation can also provide the information of the human activity and the target object he is interacting. If a person puts his hands on a laptop, when he shifts his head toward the person next to him, he is highly possible speaking to the one. If this persons head turns towards the computer screen, there is a high likelihood that this person is using a laptop. Gaze direction offers a vital cue to infer other persons intention (Wilson, Wilkinson, Lin & Castillo, 2000). The problem of extracting the head orientation on three axes from RGB camera, in other words, the angle of rolling, pitching, and yawing is described as a Perspective-n-Point problem in computer vision. The aim of this problem is to find the pose of an object from a 2D image, given the camera intrinsic parameter (focal length, optical center), the 3D model of n points and the corresponding 2D features in the image.

With leveraging large amounts of training data, deep learning method currently perform best for head orientation estimation (Bishop et al., 1995). Luckily, various public datasets are available for feature extraction. 300 W (Sagonas, Tzimiropoulos, Zafeiriou & Pantic, 2013), Helen (Le, Brandt, Lin, Bourdev & Huang, 2012) and LFPW (Belhumeur, Jacobs, Kriegman & Kumar, 2013), each of them contains large amounts of training and test images with accurate and detailed annotations of facial components. Though the head orientation is linked with gaze direction, it cant equate with gaze direction. The datasets specialized in gaze estimation field concern about eye movement, with detailed gaze directions in the different head pose. Columbia gaze dataset consists of images of various gaze directions and head poses of 56 subjects, with 5 head poses and 21 gaze directions per head pose (Smith, Yin, Feiner & Nayar, 2013), Sugano, Matsushita, and Sato (2014) built UT multi-view gaze dataset of 50 subjects with 8 views and 160 gaze directions

per subject(Sugano, Matsushita & Sato, 2014). However, the datasets above were collected under controlled laboratory conditions, MPIIGaze dataset covers a various range of facial appearance and illumination, containing 213,659 images obtained from 15 laptop users over several months (Zhang, Sugano, Fritz & Bulling, 2015). Apart from the appearance-based method we used, some studies (Wood et al., 2015) are based on the method of a computational model of eyes. However, this method is not suitable for our scenario due to the difficulty of detecting eyeball.

4.2 Body orientation

Many previous works based on multi-cameras or depth sensor. However, these approaches are not applicable in HRI, since this hardware require much more data bandwidth, power and computation resources. Performing detection on 2D camera would be an economical way. Some works (Choi, Lee & Zhang, 2016; Ahn, Park & Kweon, 2014) using the convolutional neural network and outperform the methods using HOG features and multi-class SVM classifiers (Weinrich, Vollmer & Gross, 2012). In the recent work (Choi et al., 2016), researchers cropped the human body with YOLO algorithm (Redmon, Divvala, Girshick & Farhadi, 2016), which is a real-time object detection API. A more efficient API specialized in human detection would provide more accurate information in body orientation problem. The image processed in OpenPose (Cao et al., 2016) is first analyzed by a convolutional network (initialized by the first 10 layers of VGG-19 and fine-tuned). The CNN feature maps are used as inputs to predict confidence maps for body part detection and part affinity fields for parts association. In our implementation, we apply the upper body keypoints generated by OpenPose in PnP problem solution, similar to the head orientation solution.

4.3 Pose estimation

Many existing pose estimation in 3 dimensions methods involve multiple cameras (Gavrila & Davis, 1996; Mikic, Trivedi, Hunter & Cosman, 2001). In place of multiple cameras, depth images captured by an RGB-D camera (Shotton et al., 2011) simplify the task with running in high speed in consumer hardware. Human pose estimation in monocular RGB images remains a challenging problem. Recent approaches focus on deep-learning based models. Tompson, Jain, LeCun, and Bregler (2014) propose a model combining deep Convolutional Network and a Markov Random Field (Tompson, Jain, LeCun & Bregler, 2014). Newell, Yang, and Deng (2016) demonstrate a novel convolutional network for the pose estimation task, named a “stacked hourglass” network (Newell, Yang & Deng, 2016). However, the model trained in OpenPose outperform the previous models and provide a convenient API. In our case, we need to distinguish the sitting and standing posture, and we found the distinction in the angle of upper body inclination.

4.4 Object Detection

Since the paper of Krizhevsky, Sutskever, Hinton (2012), Convolutional Neural Networks(CNN) have become a standard for image classification. However, the images used for classification in the challenge is trained with images that have only one object in the center. As a result performance is poor in cluttered environments. In real life, typical human environments are full of objects resulting in cluttered images. In fact, there is rarely a scene with only one object. To tackle this problem, Girshick, Donahue, Darrell, and Malik (2014) used bounding-boxes to identify the main objects. This region with CNN features method is called R-CNN. Firstly, R-CNN generates a bunch of boxes. Secondly, R-CNN runs a re-trained AlexNet and a Support Vector Machine (SVM)

layer to classify the regions and determine which regions are valid. Lastly, a linear regression layer is used to generate tighter bounding box coordinates of the objects in the valid regions. R-CNN works very well, while the training speed is very slow due to the three different models used in each step, in the meantime, the running speed is quite slow due to a pre-trained CNN model running on each region in the image. Girshick, the author of R-CNN, simplified and speeded up R-CNN in 2015. The Fast R-CNN trains the very deep VGG16 network 9x faster and tests 213x faster by using (Girshick, 2015). In the same year, Ren, He, Girshick, and Sun (2015) improved the fast R-CNN by sharing the CNN results for region proposal with region-based classification. Thus, there is only one CNN need to be trained in faster R-CNN. Based on faster R-CNN, He, Gkioxari, Dollr, and Girshick (2017) build high-quality segmentation for each object, by adding a branch for predicting an object mask in parallel with the existing branch for bounding box recognition in faster R-CNN.

Tensorflow object detection API (Huang et al., 2017) is a modern and convenient tool for us to build customized use cases for object detection. Faster R-CNN and mask R-CNN models trained on different datasets are also integrated in model zoo.

4.5 Human-Object Interaction Detection

Based on the rapid development in object detection, action recognition, and segmentation, a step forward of detecting individual instance is to recognize the visual relationship between individual instances. Yao and Fei-Fei proposed a random field model to encode the objects and human poses. They then took the model learning task as a structure learning problem and used a max-margin algorithm to estimate the parameter of this structure search approach (Yao & Fei-Fei, 2010). Gupta and Malik annotated a dataset of 16K people instances in 10K images with the semantic relationship (Gupta & Malik, 2015), they believed to establish the complete understanding of a scene by associating objects in the scene to the different semantic roles of the action. Based on the previous works, in order to detect ¡human, verb, object¿ triplets, the models of recent works applied multi-stream architecture consisting a human stream, an object stream, and a pairwise stream (Gkioxari, Girshick, Dollár & He, 2017; Chao, Liu, Liu, Zeng & Deng, 2018; Gao et al., 2018). The interesting of the pairwise stream is that it encodes spatial layouts between the detected human and object.

Chapter 5

Implementation of detection

Complex tasks of Human-Robot interaction are with a high demand for an in-depth understanding of human body composition. RGB-D camera, such as the Microsoft Kinect, provide depth information which makes human body detection more efficiently. However, RGB-D cameras, with much larger size, higher price, and higher power consumption, have lower resolution, compared with color cameras. In assistive robots, the color cameras are more widely used. To meet the demand for Human-Robot interaction, the challenging problem is to get real-time skeletal information through a single RGB camera. Openpose, a real-time open sourced platform, can detection multi-person keypoints estimation including 25-keypoint body/foot, 2x21-keypoints hand, and 70-key points face. Despite the multifunction in human detection, It runs robustly with partly cover on a human body part, which facilitates the application in real-life Human-Robot interaction. In this chapter, head orientation, body orientation, and pose estimation will be implemented based on OpenPose.

5.1 A Proposal of Detection Algorithm Structure

For a social robot navigating, the personal space model is built on recognizing human-object interaction, human head pose, human body orientation, and human pose (see elaboration in chapter 3.3). The detection results of these approaches should be taken as the inputs of robot navigating behavior. We propose an approach of how to develop social awareness to personal space model. Figure 5.1 shows how detection results can be combined into a probabilistic model of personal space.

The proposed approach takes the results of detection as inputs, generates a probabilistic model of personal space and updates it. First, the pose, body orientation, and head direction would help find the target object and reinforce the correct human-object interaction behavior. Second, the pose, body orientation, and head direction update the parameter in the single person personal space model. The affordance space of multiple detected objects is re-assigned based on human-object interaction behavior, in the meanwhile, the human-object interaction behavior is associated with its activity space. The new probabilistic model of personal space can be presented as:

$$Individual\ personal\ space\ model \times (1 - Pr(activity\ space))$$

It's important to note here that the affordance spaces only take into account the objects which are near human and the target object in human-object interaction. The other objects can be regarded

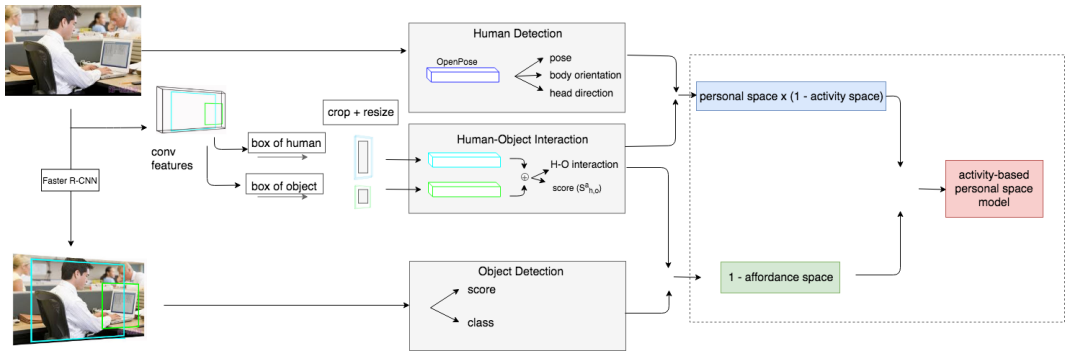


Figure 5.1: Overview of Our Proposed Detection Approach

as an obstacle. It's not for robots to avoid affordance space during navigating but not stay too long in this area.

In the following, we implemented the algorithms separately. Since it's not the goal of present work to implement the complete structure on robot, we skipped the model fusion. It's supposed to be finished in the future work.

5.2 Indoor-activity-scene Dataset

Before we proceed to the detail of implementing the computer vision methods, we created an indoor-activity-scene dataset to evaluate the accuracy of each method we applied (except the head orientation estimation). For head orientation estimation, it is tested on the head pose database (Gourier, Hall & Crowley, 2004). The elaboration of this dataset is in chapter 5.3.1. Our indoor-activity-scene dataset varies from pose (sitting and standing), activity (watch a TV, read a book, and using a laptop), and shooting angle (back, front, left or right, left or right back, left or right front), light level (dark and bright). It contains 144 pictures taken from a 'living room', and Figure 5.2 shows some examples in our dataset. To notice here, the dataset is insufficient to draw a precise scientific result of the algorithm performance, but it can provide a pilot result on our topic-tailored scenarios. Thus, after each implementation section, we tested the performance of our approach on this test dataset.

(a) (b)

(c) (d)

Figure 5.2: The pictures from our test dataset: (a) is a picture shot from the front side when the human stands in bright light condition and (b) is a picture shot from the left-front side when the human stands in dark light condition; (c) is a picture shot from the front side when the human is reading a book in bright light condition and (d) is a picture shot from the left-back side when the human is using a laptop in bright light condition

5.3 Head Orientation

As we presented in the previous chapter, we regarded head orientation problem as a PnP problem. For each PnP problem, there are three coordinate systems. World coordinates (U, V, W), camera coordinates (X, Y, Z), and 2D image coordinates (x, y). The 3D points in world coordinate can be transformed into 3D points in camera coordinates through rotation R (a 3x3 matrix) and translation t (a 3x1 vector). 3D points in camera coordinates can be projected on 2D image coordinates through the known camera intrinsic parameters.

The location (X, Y, Z) of the point P in the camera coordinate system can be yielded from the following equation:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} \mathbf{R} & | & \mathbf{t} \end{bmatrix} \begin{bmatrix} U \\ V \\ W \\ 1 \end{bmatrix} \tag{5.1}$$

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} r_{00} & r_{01} & r_{02} & t_x \\ r_{10} & r_{11} & r_{12} & t_y \\ r_{20} & r_{21} & r_{22} & t_z \end{bmatrix} \begin{bmatrix} U \\ V \\ W \\ 1 \end{bmatrix} \tag{5.2}$$

For the point P in camera coordinates (X, Y, Z), the 2D image coordinates (x, y) can be conducted from the equation involving camera matrix (M), where

$$M = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

where, (f_x,f_y)is the focal lengths and (c_x, c_y) is the optical center. In most cases, the lens distortion is considered negligible . And the equation is given by:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = s \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{5.3}$$

After combining the equation 5.3 and equation 5.3, the position in 2D image coordinates can be predicted from given 3D points in the world (a facial keypoint or a body part keypoint), if the right pose ($\mathbf{R}$ and $\mathbf{t}$) is given.

For the equations above, Direct Linear Transform (Sutherland, 1974)(Sutherland, 1974) is used for getting the pose $\mathbf{R}$ and $\mathbf{t}$. However, the DLT algorithm does not minimize the reprojection error. Reprojection error represents the sum of squared distances between the projected 3D face points and 2D image points. Since the 2D points of face and body can be detected from OpenPose, an intuitive way to converge the error curve is to adjust the $\mathbf{R}$ and $\mathbf{t}$ to align the projected 3D points with detected 2D image points. Levenberg-Marquardt optimization is widely used to minimize reprojection error. In our implementation, CV_ITERATIVE in OpenCV's function solvePnP was chosen, which is based on Levenberg-Marquardt optimization. Once we get the rotation matrix, it can be converted into Euler angles (roll, pitch and yaw). Figure 5.3 shows the result of head orientation estimation:

5.3.1 Head Orientation Detection Result on Head Pose Image Database

The head pose database (Gourier et al., 2004) contains 2790 monocular head images of 15 persons, in steps of 15 degrees of pan and tilt angles from -90 to +90 degrees. The sample images of this dataset are shown in Figure 5.4. The pan angle and tilt angle represent the yaw angle and pitch

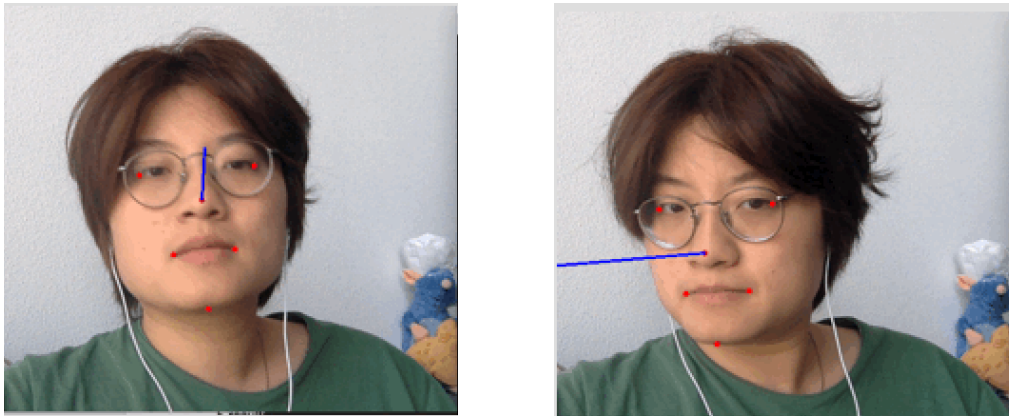


Figure 5.3: Head Orientation Estimation: red dots represent the points chosen for orientation estimation, the blue line represents the estimated head orientation.

Figure 5.4: Sample pictures of the head pose database

angle in our head pose estimation result respectively. However, we consider the most common cases in our task scenario. Since people seldom keep their heads at +/-90 or +/- 75 degrees of pan angle, +/- 60 degrees of tilt angle for a long time, pictures with heads of the above angles are excluded. For evaluation, we compute the yaw success rate (YSR) and pitch success rate (PSR), YSR is the percentage of face instances where of the detected faces, the estimated pan angle is within +/- 10 degrees of the annotation, PSR is percentage of face instances where the estimated tilt angle is within +/- 15 degrees of the annotation. During the evaluation, we found this detection approach can hardly work on the pictures of +/- 60 degree pan angle, we excluded these pictures and tested on the remaining dataset. The result is reported in Table 5.1.

Table 5.1 result shows a better detection result in pan/yaw angle. For daily activities, people's pitch angle is relatively stable (from -15 degrees to +15 degrees), however, people have a larger range in yaw angle and move more frequently. Like most of the other detection techniques,

head pose database (excluded +/- 60 degrees pan angle)		+/- 60 degrees pan angle	
YSR	PSR	YSR	PSR
54.54%	43.93%	0%	1%

Table 5.1: Head orientation detection evaluation result on head pose database, yaw success rate (YSR) and pitch success rate (PSR) are presented

our approach was evaluated on static images or camera-still videos, in the application of those techniques on robot motion, the dynamic images changing in the process of a robot approaching a person should be considered. At the start phase, the distance between the robot and the person may be around 4-5 meters, and its challenging for the robot to generate an accurate result on the facial landmark. Besides, the illumination in the living room is usually in high contrast at daytime (brightest near window) and dim at night, and unstable illumination conditions also set up the barriers. In contrast, body orientation estimation is an easier issue at the early phase. Once the robot approaches close enough to the human, in another word, the human face contains enough pixels in the image, and the head orientation should be put on more weights. In this case, a correct body orientation estimation should provide more insights into the robot approaching behavior. Next, elaboration will be present on how we implement body orientation estimation.

5.4 Body Orientation Estimation

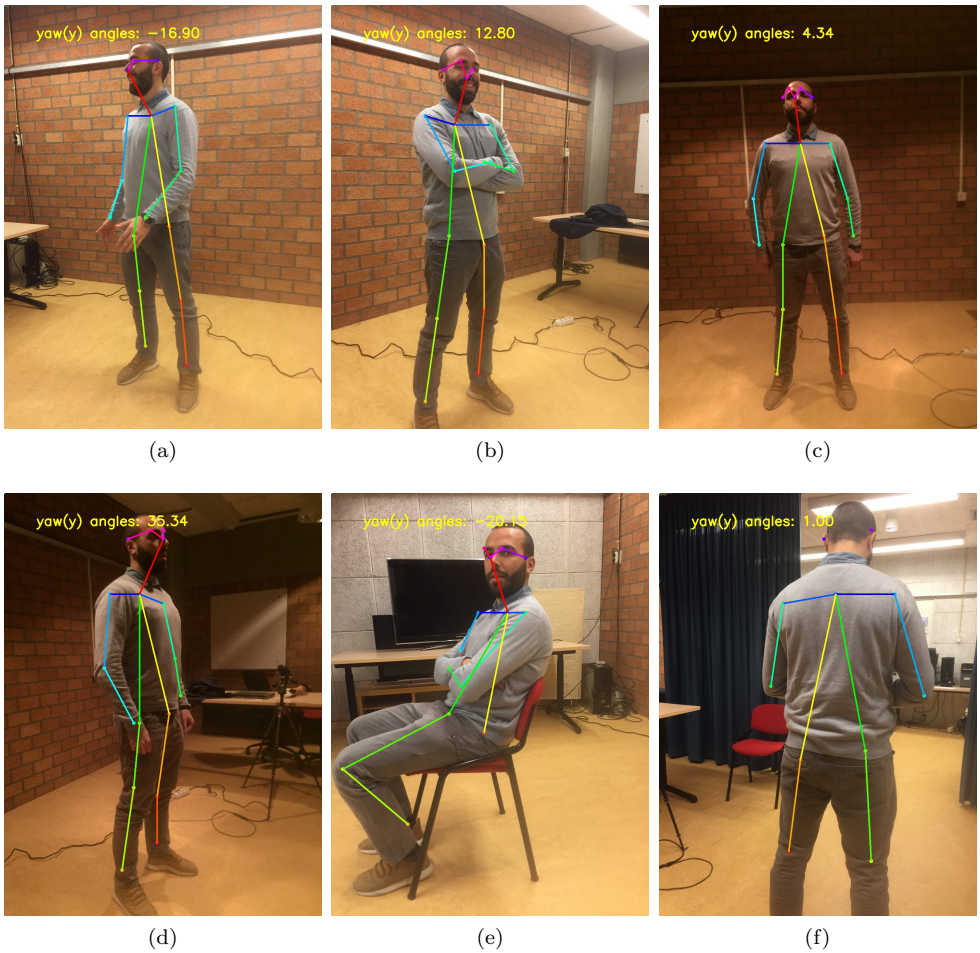
Same with head pose estimation, body orientation estimation is a Perspective-n-Point problem, n represents the minimum number of given points in the 3D model and their corresponding 2D features in the image coordinates. In our implementation of head orientation estimation, 6 points (left corner of left eye, right corner of right eye, nose tip, left mouse corner, and right mouse corner) in 3D face model are chosen. Instead, in body orientation estimation, we chose the points of the left shoulder, right shoulder, neck, left hip, and right hip. By replacing the points on the face with the points on the upper body, the problem of body orientation estimation was solved. We used the 3D body model coordinates provided by BodyParts3D (Mitsuhashi et al., 2008).

5.4.1 Body Orientation Detection Result on Indoor-activity-scene Dataset

Our indoor-activity-scene dataset only concerns body images shot from back, front, left or right, left or right back, left or right front, namely, in steps of 45 degree of yaw angle. Thus, YSR in body orientation evaluation is the percentage of body images where the estimated pan angle is within +/- 22.5 degrees of the annotation. Different from head orientation detection, yaw success rate (YSR) is the only criterion, due to people rarely pitch or roll their bodies. The evaluation result is shown in Table 5.2, we also present the inference result on images (see Figure 5.5). From the result, we can draw a conclusion that our body orientation detection approach works well when the robot approaches from -45 to +45 degrees with respect to the standing human; it also works well on both human poses when the approaching angle is from -22.5 to +22.5, namely, from the front of the human. These failure examples shows our body orientation detection can not inference correctly from the side. However, the detection result doesn't go too far way when the human is standing, while a sitting position from the side may raise OpenPose's confusion that the chair would be regarded as parts of a human body. A drawback in our detection method is that, when the robot approaches from the back, the yaw angle result is similar with the that of the picture taken from front. The color of body parts shows OpenPose can distinguish the front and the back, the error caused by the process of converting the rotation matrix into Euler angle. Thus, we consider to add 180 angle to the Euler angle when the OpenPose inference a back picture.

Shooting angle	YSR (human standing)	YSR (human sitting)
left front	75%	0%
right front	67%	0%
front	100%	100%

Table 5.2: Body orientation detection evaluation result on indoor-activity-scene dataset, yaw success rate (YSR) is presented based on human pose



(a) (b) (c)

(d) (e) (f)

Figure 5.5: The inference results of body orientation detection: this approach can successfully work on (a), (b) and (c). The pictures in the bottom line are the failure detection examples. The human in (d) is standing almost 90 degree to the camera, the human in (e) is sitting -90 degree to the camera, and (f) is shot from the back, the ground truth is 180 or -180 degrees.

5.5 Pose Estimation

For the pose detection part, since a 2D camera is used in our study, we compile the OpenPose method to detect and collect the key points in the human body: the points of neck, left hip, right hip, left knee and right knee. We can compute the vector of 'spine' and 'leg' from left hip to neck and from right hip to the neck, from left hip to left knee and from right hip to right knee, respectively. The left spine-leg angel presents the angle between the vectors of left spine and 'left leg', and the right spine-leg angle presents the angle between the vectors of 'right spine' and 'right leg' (shown in Figure 5.6).

The left or the right spine-leg angle θ can be derived with below formula, where V_{spine} and V_{leg} represents the vector of spine and leg, P_{neck}, P_{hip}, and P_{knee} represents the keypoints of neck, hip, and knee respectively :

$$V_{spine} = P_{neck} - P_{hip}$$
$$V_{leg} = P_{knee} - P_{hip}$$
$$\theta_{spine-leg} = \arccos(V_{spine} \cdot V_{leg} / \|V_{spine}\| / \|V_{leg}\|)$$

In the real-life setting, the human is not always in front of the camera. If only one spine-leg angle is detected, the final spine-leg angle is defined by this angle. If both of the angles are detected, the final spine-leg angle θ is defined as the mean of the two angles:

Use the previous spine-leg angle
if left spine-leg angle and right spine-leg angle both exist **then**
    $spine - leg\ angle \leftarrow the\ mean\ of\ (left\ spine - leg\ angle\ and\ right\ spine - leg\ angle)$
else
    if $left\ spine - leg\ angle\ >\ right\ spine - leg\ angle$ **then**
        Use the measurements of left spine-leg angle
    else
        Use the measurements of right spine-leg angle
    end if
end if

This procedure contains two phases: Phase 1. Spine-leg angle model learning 1. Collect spine-leg angel samples of the standing pose from the training images/videos. 2. Estimate your model parameters, that is, the mean and the (co)variance, from the sample data. 3. Save the parameters for later use in Phase 2.

Phase 2. Pose detection 1. Set a threshold probability. 2. Take the video from 2d camera as input, get the input spine-leg angle. Adopt Gaussian function and then compare the result with the threshold probability.

Hence we propose the following Threshold Probability algorithm for detecting body pose. First of all, we calculate $P(standing \mid \theta_{spine-leg} = \alpha)$ by applying the Gaussian equation:

if $P(standing \mid \theta_{spine-leg} = \alpha) > P(threshold)$ **then**
    $pose = Standing$
else
    $pose = Sitting$
end if

A simple hypothesis test has completely specified models under both the null and alternative hypotheses, which are written as:

$H0 : Pose = standing$

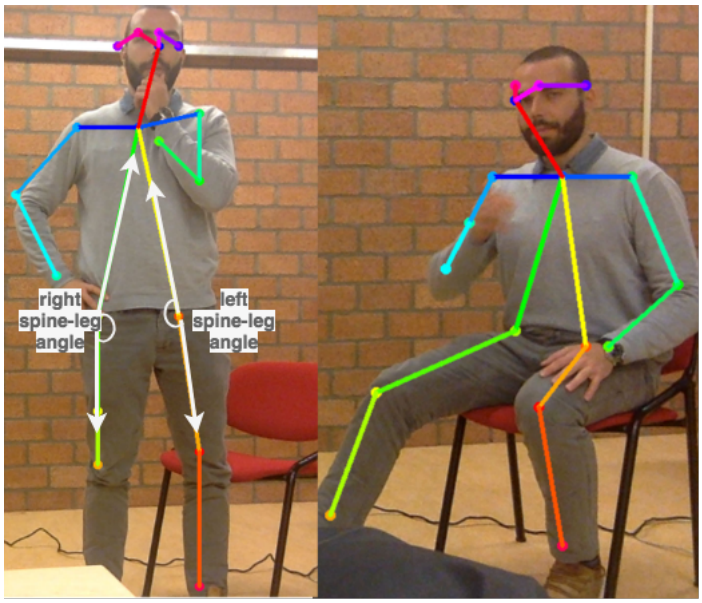


Figure 5.6: The spine-leg angel

$H1 : Pose = sitting$

The reason for choosing standing as the $H0$ is that, when people are standing, they have less space to lean the upper body, while they can lean forward and backward in a large range of angles when sitting. Our data collected in Phase.1 also prove this quantitatively. When people are sitting in front of the camera, the spine-leg angles obey the distribution of $G(152.15, 5.12^2)$,when people are sitting with side facing the camera, spine-leg angles obey the distribution of $G(103.84, 7.99^2)$, in the meanwhile, the probability density function (PDF) of standing pose is $G(168.03, 2.18^2)$.Much smaller variance found in the PDF of standing pose. Beside, when people are sitting, the mean angles are very different in different body orientations.

The likelihood ratio test is based on the likelihood ratio, which is often denoted by Λ. In this case, the likelihood ratio is defined as:

$$\Lambda = \frac{P(\theta_{spine-leg} = \alpha \mid standing)}{P(\theta_{spine-leg} = \alpha \mid not\ standing)}$$

The likelihood ratio test provides the decision rule as follows:

If $\Lambda > C$, do not reject $H0$; if $\Lambda < C$, reject $H0$; if $\Lambda = C$, reject with probability q, The value if C is selected to meet a desired significance level α , C and q are derived from the relation:

$$q \cdot P(\Lambda = C \mid H0) + P(\Lambda < C \mid H0) = \alpha$$

We set the significance level $\alpha = 0.05$, then:

$$\Lambda = \exp\left[-\frac{n}{4}(x - 173.03)^2\right] \geq C$$

The equation can be transformed as follow:

$$\frac{|x - 168.03|}{2.18/\sqrt{n}} \leq Z_{0.025} = 1.96$$

Thus, we can detect the pose is standing or not by examining whether the spine-leg angle is within the range of $(163.76, 172.3)$. Since we easily know the angle of standing is larger than sitting, the

Figure 5.7: The failure examples of pose estimation: the left is detected as sitting with a spine-leg angle of 162 degree; the right is detected as "unknown" with none of body parts is recognized.

pose	accuracy
standing	88%
sitting	95.56%

Table 5.3: Pose estimation evaluation result on different poses

angle range can be extended to $(163.76, 180)$. Therefore, we set the range of spine-leg angle, $(163.76, 180)$ for standing.

5.5.1 Pose Estimation Result on Indoor-activity-scene Dataset

We evaluated our pose estimation method on standing and sitting pictures separately, the accuracy is reported in Table 5.3. As this table shows, the accuracy is quite satisfactory. Then, we examined the failure examples (see Figure 5.7), our method failed to work in some cases when the human is occluded seriously or when the human is at almost 90 degrees to the camera in a dim light condition. Occlusion happens frequently when the robot is approaching. Since sitting is a much regular pose when human is doing any activities, our algorithm set the incomplete-bodyparts-detected pose as sitting. The failure example in sitting cases is caused by none of human parts is detected, since we evaluated our method on pictures, in the application, the robot can regard this unknown pose same with the previous frame.

5.6 Object Detection and Result on Indoor-activity-scene Dataset

We use Tensorflow object detection API in our implementation. We evaluated object detection on our indoor-activity-scene dataset. The purpose is to examine the performance of object detection from a different angle. The result is shown in Figure 5.8. Faster R-CNN with inception model can detect all the important objects (book, laptop, TV, and person) from a large range of angles (see 5.8a to 5.8f). SSD with Mobilenet model can detect a large object, like the person and TV. While it can't detect the small objects, like a book (see Figure 5.8g and 5.8h) or part of a laptop (see Figure 5.8j). However, the inference time on SSD with Mobilenet model is much faster than

Faster R-CNN with inception model. Running detection on my 3 GHz Intel Core i7 and 8 GB 1600 MHz DDR3, the inference time of SSD with Mobilenet model is 3.87 seconds for a $300X400$ image; the computation time of faster R-CNN with inception model is 6.85 seconds for a $300X400$ image. To recognise the human-object interaction, it's vital to detect the small objects. We should consider using Faster R-CNN for a precision purpose. However, the current inference time is too long for robot navigation. One of the reasons comes from the posted models are trained on a huge dataset containing too many categories of objects, thus, to train model on a smaller and specified dataset could reduce a lot runtime. For each object in Figure 5.8, the class K_o and score S_o are saved for later use in human-object interaction. It's important to point out the confidence level of different objects in object detection result (Faster R-CNN) to understand the human-object detection results in Chapter 5.7: person, 97% to 99%; TV, 60% in Figure 5.8f and 98% in Figure 5.8e; book, 53% in Figure 5.8b and 76% in Figure 5.8a; laptop, 97% in Figure 5.8c and 66% in Figure 5.8d. Considering the trade-off for inference time and precision, the system equipped with Faster R-CNN method and trained on a larger indoor-activity-scene dataset can achieve the goal of multiple object detection in a relatively short time. Another reason of building an own training dataset is: in our problem setting, the robot perceives the human and surrounding world from its camera, the shooting angle of the images are rare among most popular datasets, which is from a relatively low angle. To build an own training dataset can make up for the defect from the pre-trained model and boost the model performance.

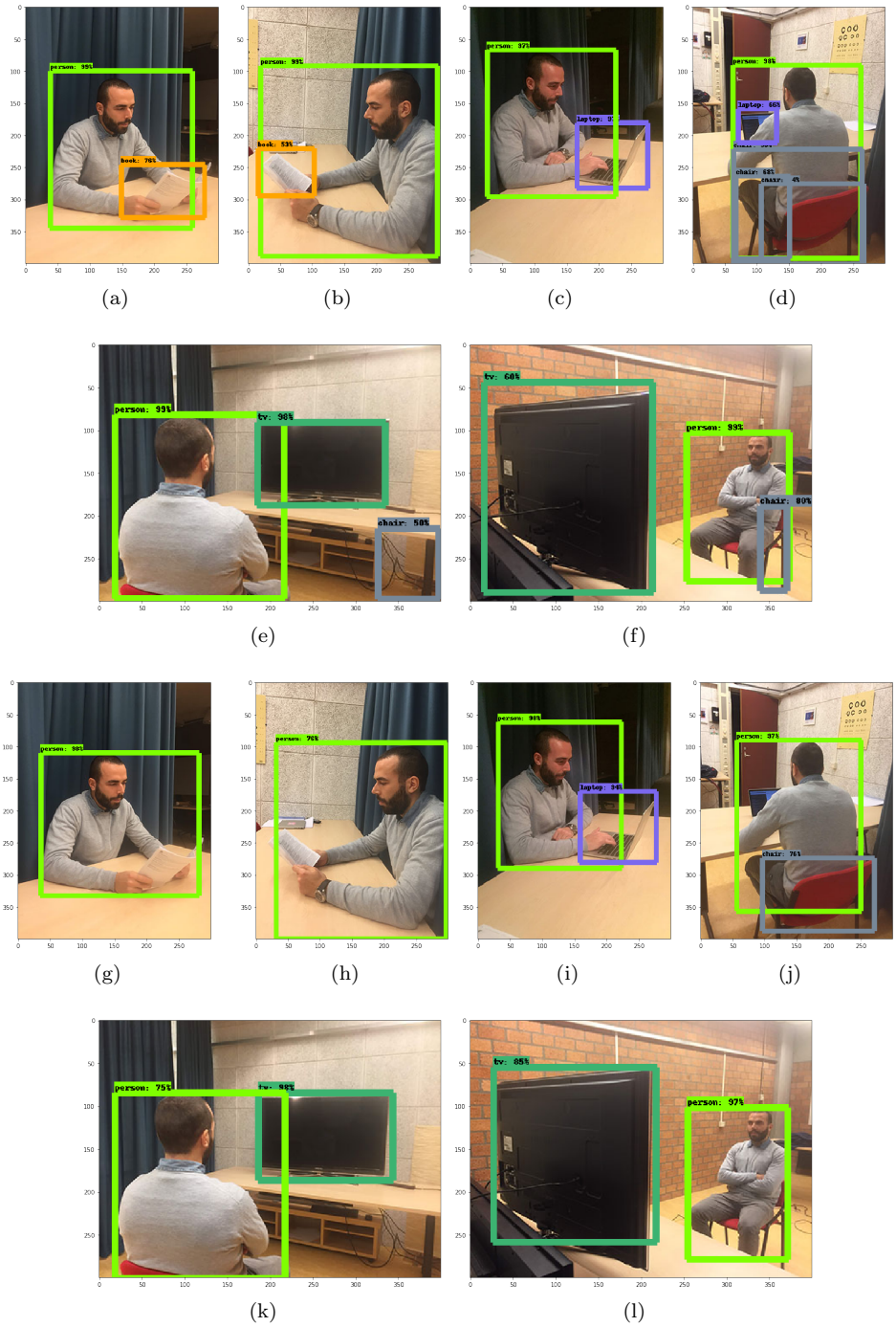


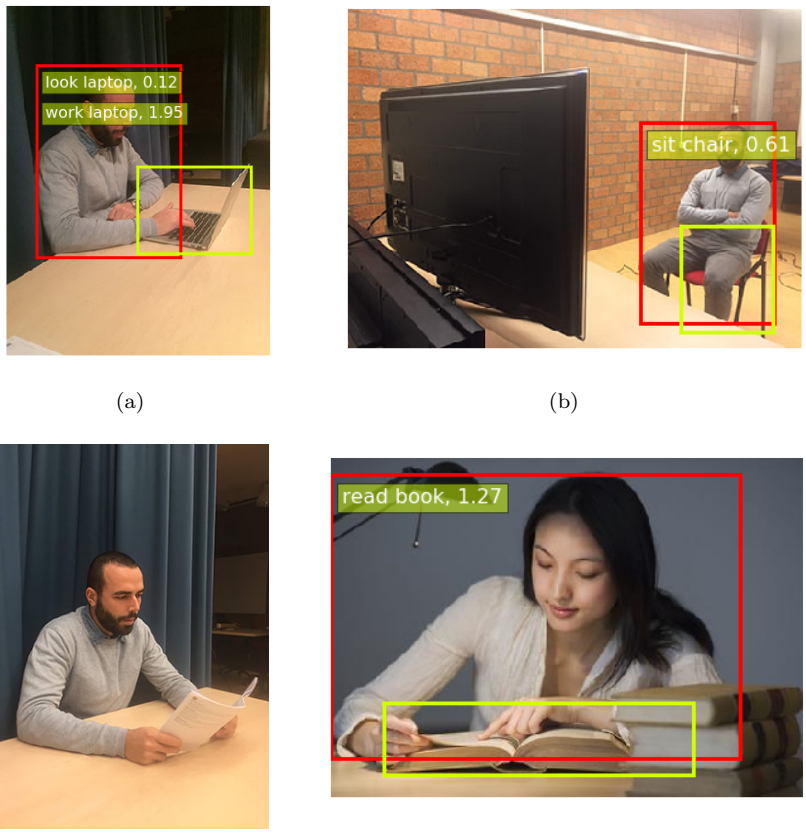
Figure 5.8: The results of object detection: (a) to (f) show the detection result of show the detection result of Faster R-CNN with inception model; (g) to (l) show the detection result of SSD with Mobilenet model.

5.7 Human-Object Interaction Recognition and Result on Indoor-activity-scene Dataset

For the implementation of human-object interaction, we use the framework provided by iCAN (Gao et al., 2018), running on one NIVIDA Tesla P100, the time to detect the H-O-I relationship is 600ms per image (300×400). However, in the original paper (Gao et al., 2018), this method takes less than 75ms to process an image of size 480 × 640 in HICO-DET (including ResNet-50 feature extraction, multistream network, attention-based feature extraction, and HOI recognition). The difference may come from different images source. The key of human-object interaction detection is to build the relationship between the individual object and person. First, they use Faster R-CNN to detect all individual person/object, bounding boxes and scores of all the detected person/object instance are stored, as shown in Figure 5.8; second, all possible person and object bouncing boxes are evaluated and interaction score is predicted for each pair. The multi-stream model consists of a human stream, an object stream, and a pairwise stream. The function of each stream is to detect and score humans, detect and score objects, and encode the spatial layout between human and object bounding boxes, respectively, in the end, each $< human, verb, object >$ triplet is scored as $S^a_{h,o}$, which is based on these items: confidence on each object in object detection s_h and s_o; the action prediction score based on the human s^a_h and the object s^a_o; the prediction action score based on the spatial relationship between the human and the object s^a_{sp}. $S^a_{h,o}$ is expressed as:

$$S^a_{h,o} = s_h \cdot s_o \cdot (s^a_h + s^a_o) \cdot s^a_{sp}$$

The result of applying H-O-I detection is shown in Figure 5.9, iCAN can recognise 'work laptop' perfectly, while failed to recognise 'watch TV' and 'read book' on indoor-activity-scene dataset. First we ascribed the 'book' we used in our dataset, it's a thin booklet. We change another picture (see Figure 5.9d), then this picture can be recognised correctly with a high score. The reason of failing to recognise 'watch TV' is the low score of object detection (TV: 60%) and the spatial relationship between the human and the object s^a_{sp} due to the distance. The detector can not recognise human behavior from the back due to lacking human appearance features. Thus, this H-O-I detector raises some constraints during in application: the H-O-I behavior can not be detected from the back; the human-object distance should be short (within 1 meter); the object should be detected with high confidence; a system equipped with one GPU. The object detection confidence problem can be solved by building own training dataset. The human-object distance problem may also be somewhat tackled by an own training dataset. And, for calibrating the s^a_{sp}, we suggest to revise it based on human's head orientation. When the robot approaches from the back, at the early phase, the H-O-I behavior is not that important compared with body orientation detection, therefore, the robot can mute H-O-I detection while no human appearance feature is detected.



(a) (b)

(c) (d)

Figure 5.9: The results of human-object interaction detection: (a) is recognised as 'work laptop' with a high confidence; (b) is only recognised as 'sit chair'; (c) is not recognised as any behavior; with containing a thick book, (d) is recognised as 'read book' with a high confidence

Chapter 6

Results

6.1 Individual-person PS model

Torta, Cuijpers, and Juola (2013) compared different approaching angles and human status (sitting or standing), and established a parametric model that describes the personal space model of a single person sitting or standing in an open space, which is written as $PS(\rho, \theta|\alpha)$. For a position x=[ρ, θ]expressed in polar coordinates with respect to the users reference frame, the expression is given in equation.(X), where represents the distance to the human position, represents the angular distance, represents the users state (sit or stand):

$$PS(\rho,\theta|\alpha) = \alpha \cdot \eta_A \cdot \exp\left(-\frac{(\rho-\gamma_A(\theta))^2}{\delta^2_{\gamma_A}}\right) + (1-\alpha)\cdot\eta_B\cdot\exp\left(-\frac{(\rho-\gamma_B(\theta))^2}{\delta^2_{\gamma_B}}\right) \tag{6.1}$$

We use the parameters Torta, et al. acquired from their user studies, detail can be seen in Table. 1, where mean value of γ_B is (=173 cm, SD=68.3 cm), mean value of γ_A is (=182 cm, SD=35.9 cm).

Experiment	Pol. ID	Dependent variable	ANOVA	R^2	Coefficients
Sitting	γ_A	Distance [cm]	F(2,122)=5.3, p=0.006	0.08	[0.005,0.029,194.757]
Standing	γ_B	Distance [cm]	–	-	[173.528]
Sitting	η_A	Preference	F(2,122)=27.64, p¡0.001	0.312	$[0.254,3.543,4.143]\cdot 10^{-3}$
Standing	η_B	Preference	F(2,125)=3.39, p=0.036	0.052	$[0.054,4.179,3.953]\cdot 10^{-3}$

Table 6.1: Summary of the parameters of the approximating curves that relate distance (γ) and preference (η) to direction of approach for the two experiments used in individual-person PS model

We visualize the individual-person PS model, which is depicted in Figure 6.1.

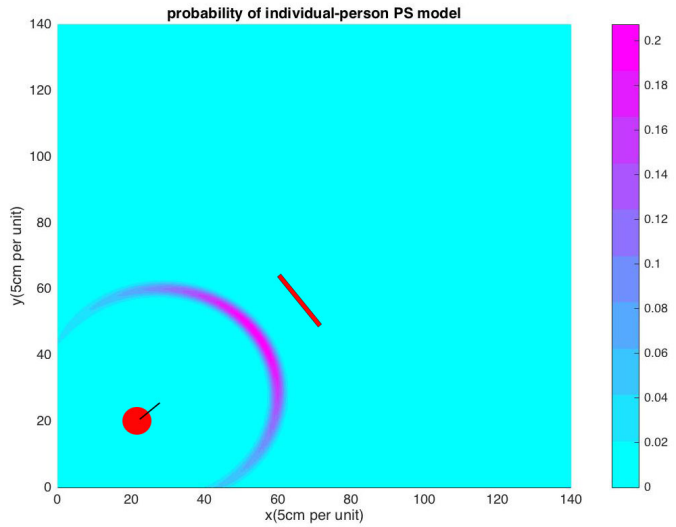


Figure 6.1: The probability of individual-person PS model: human is located at (20, 20) as a red circle, facing angle is 60 degrees; the TV is placed at the polar coordinates (60, 3) with respect to the users reference frame, with width of 90 cm and height of 125 cm.

6.2 Activity and Affordance Space

The shape of activity and affordance space is not defined by any previous work (Rios-Martinez et al., 2014). Since the focus area of the human eye can be adjusted real-time with the interacting object changes, when a person is reading a book, using a laptop, or watching TV, the object is regarded as a rectangular plane from the persons point of view. Thus, the borders of the activity space can be drawn from the edges of this rectangular plane to human eyes. If we ignore the distance between human eyes, the activity space in our model is described as a triangle space with smooth borders. Since Its unclear where the human the focus area ends, its acceptable to describe each border of this triangle area as Gaussian functions with low variance, the inner area in activity space is a forbidden area which contains the highest human-object interaction density (as seen in Figure 6.5, the triangle area). We formula the model in the polar coordinates with respect to the humans reference frame, for the points x=[ρ, θ], represents the distance to the human, and represents the angular position in polar coordinates. When is smaller than the distance between human and the object, the activity space is modelled as following expressions:

$$f(\theta \mid \theta_1, \theta_2) = \begin{cases} 1 & \theta_1 \leq \theta \leq \theta_2 \\ \exp(-\frac{(\theta-\theta_1)^2}{2\cdot\sigma_1^2}) & \theta < \theta_1 \\ \exp(-\frac{(\theta-\theta_2)^2}{2\cdot\sigma_1^2}) & \theta > \theta_2 \end{cases} \tag{6.2}$$

The term θ_1 represents the angular position of the left border of activity space, the term θ_2 represents the angular position of the right border of activity space, which equal to the angular positions of the bottom vertexes of the object plane with respect to the human's reference frame. σ_1^2 represents the variance associated with buffer regions around the two borders. As shown in Figure.6.5, the yellow triangle region is the activity space, which is forbidden for robot. However, the borders are as sharp as the sides of rectangle wave if we see from the cross section (see the rectangle box in Figure 6.3). When the robot is close to the border, it would suddenly brake and stop, which is an unnatural behavior from the humans perspective and also harmful to the robots hardware. Therefore, we applied Gaussian smoothing to the borders. Another question

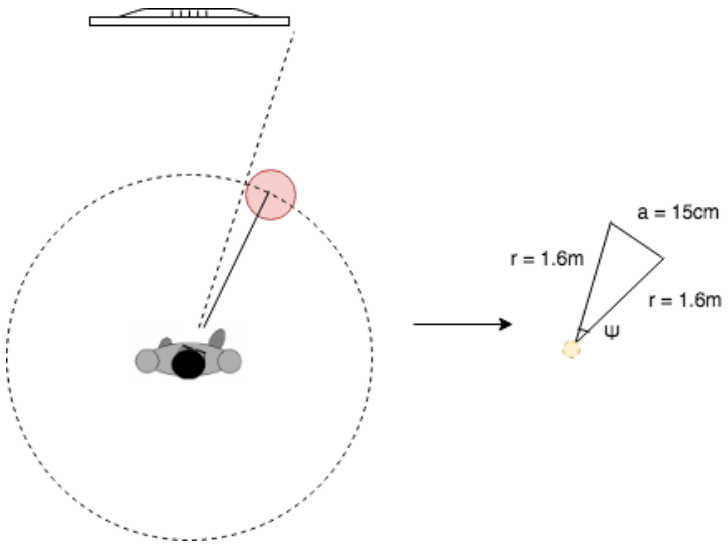


Figure 6.2: The left image shows a Nao robot stops just next to the border and in the personal space; the right image translates the situation shown in the left image into a mathematical problem to derive the value of ψ

raised when using the Gaussian smoothing: what value should the variance σ_1 be? The situation is depicted in Figure. 6.2 (left image). Ideally, the optimal stopping position for the robot is to stand just beside the border. Since we use the Nao robot in our cases, the width of this robot is about 30 cm. The mean stopping distance can be extracted from the study of Torta et.al (2013), we set it 1.6 meters. As shown in Figure. 6.2 (right image), ψ can be derived from the Cosine Law: $\arctan\left(1 - 0.15^2/2/1.6^2\right) = 1.54$, which is a very small value. Moreover, during the navigation, the robot is constrained by a force to find the shortest path, if the variance is large, the robot is highly possible to stop at the position which is far away from the border. Thus, we should set the variance σ_1 a very small value to make the border smooth and also counter the negative impact. To keep the variance reasonably small, we chose σ_1 the constant pi/36. The influence of the angular position on the model is shown in Figure 6.3.

As elaborated in chapter 3.2, in our opinion of view, the affordance space of one object is independent from the current human activity, but the current human activity also provides the clew whether this object will be engaged in human-object interaction in a short term. Since the shape of affordance space for different objects vary, an affordance space is generally a small area around the object. The robot can take the object as a normal obstacle, which is controlled by an obstacle avoidance algorithm. Therefore, the affordance space of an obstacle in the probabilistic model equals 0. When an object near the human is possible to be interacted with this human in a short time, like the chair near a standing person, the affordance space will be the same as the activity space, namely, the affordance space is the sector area centered on the human position. Generally, if an object would be in the potential interacted object in the short run, the affordance space of this object equals to activity space, which is $f(\theta)$; if an object would not have interacted in the short run, the affordance space of this object equals 0. We suppose a potential interaction mode ξ for any object, which is a confidence level of potential interaction. Thus, the affordance space of one object is expressed in equation 6.3:

$$a(\theta, \xi) = f(\theta) \cdot \xi, \tag{6.3}$$

where

$$\xi = \begin{cases} 0 & \text{the human is about to interact with the object} \\ 1 & \text{the human is not about to interact with the object} \end{cases} \tag{6.4}$$

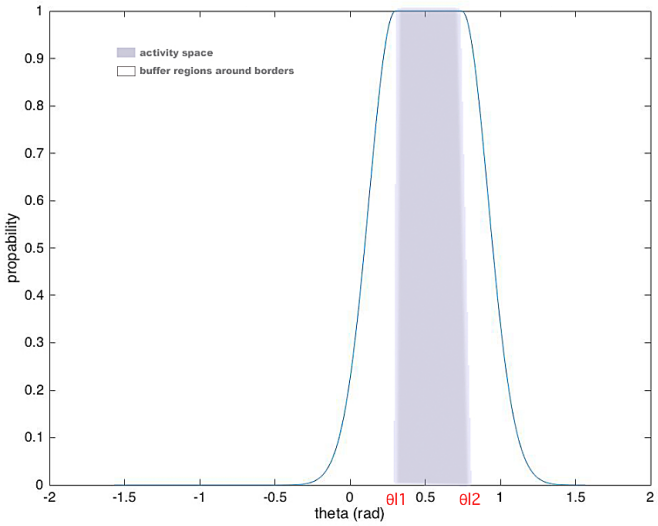


Figure 6.3: The Probability of Activity Space Model Changing Along with Angular Positions

We'll take an example to explain how to decide the status of an object. In our simple scenario: a person is watching TV sitting on the couch in the environment of the living room, the lamp can be regarded as an obstacle, and the ξ of the lamp is off. While the status of the empty chair near the human is related to the human pose. If the person is sitting, the ξ of the chair is off; if the person is standing and face to the chair, the ξ of the chair is on.

6.3 Blind Area

For any object the human is staring at, a blind spot is generated when the object blocks the line of sight. As shown in Figure 6.4, the line of sight and the ground converge to a point. In certain cases, when the object almost reaches the height of human eyes or even higher, for instance when a person is using a personal computer, the far point could be exceptionally far or even not exist. In other cases, the shape of the blind spot is geometrically seen as the extension of the activity space. The space is related to the width and height of the object, the height of the human, and the height of the robot. Similar to the activity space formula, for a position x=[ρ,θ] expressed in polar coordinates with respect to the users reference frame, when ρ is bigger than the distance between human and the object d_{ho}, the blind space should be downward to the floor:

$$b(\rho \mid \rho_1, \rho_2, \theta_1, \theta_2) = \begin{cases} 1 & \rho \leq d_{ho} \\ exp(-\frac{(\rho - d_{ho})^2}{\eta \cdot d_{far}}) & \rho > d_{ho} \end{cases} \tag{6.5}$$

where d_{ho} represents the Euclidean distance between the object and human, namely, the distance between $[\frac{\rho_1+\rho_2}{2}, \frac{\theta_1+\theta_2}{2}]$ and $[0, 0]$. d_{far} represents the distance between far point and human, which is expressed as $h_{human} \cdot d_{ho}/(h_{human} - h_{object})$. η represents the ratio of robot height to human height and is expressed as h_{robot}/h_{human}. Though the height of robots and human varies, in our case we chose the Nao robot (height: 0.58 meters) and set the human height to 1.7 meters, the η equals to 0.34. In sum, representing the area robots should avoid, activity space and blind spot can geometrically and mathematically combine into one. The model visualization of combination area is shown in Figure 6.5.

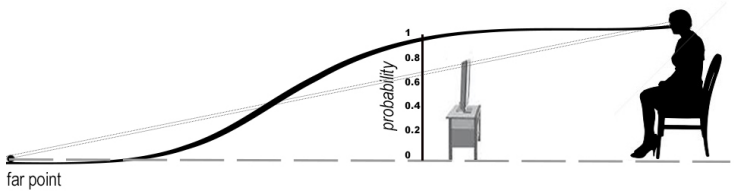


Figure 6.4: The Probability Distribution of the Blind Spot from Side-view

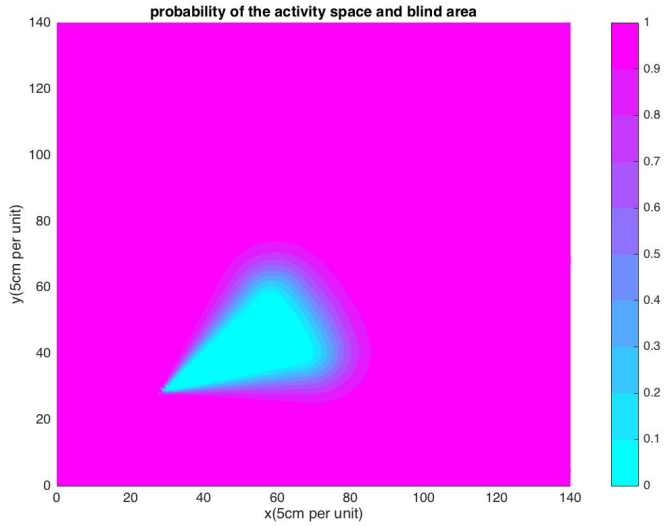


Figure 6.5: The Probability of the activity space and blind area when a human is watching a TV

6.4 Activity-based PS Model Visualization

Torta, Cuijpers, and Juola (2013) compare different approaching angles and human status (sitting or standing), and establish a parametric model that describes the personal space model of a single person sitting or standing in an open space, which is written as PS(ρ,θ,α). We use a uniform model to describe personal space based on different human-object interaction behavior. Compared with the model built by Torta and her colleagues (2013), our model takes into account the activity space and affordance space, as well as the blind spot. For a position x=[ρ,θ] expressed in polar coordinates with respect to the users reference frame, the expression is given in equation **??**, where ρ represents the distance to the human position, θ represents the angular distance, $[\rho_{i1}, \theta_{i1}]$ and $[\rho_{i2}, \theta_{i2}]$ represents the position of the bottom vertexes of the $i-th$ object plane with respect to the user's reference frame, S^a_{h,o_i} represents the score coming from the human-($i-th$)object interaction detection. The activity spaces for multiple objects is modelled in equation 6.6

$$AS_i(\rho, \theta|\rho_{i1}, \rho_{i2}, \theta_{i1}, \theta_{i2}) = b_i(\rho \mid \rho_{i1}, \rho_{i2}, \theta_{i1}, \theta_{i2}) \cdot f_i(\theta \mid \theta_{i1}, \theta_{i2}) \cdot S^a_{h,o_i} \tag{6.6}$$

The term i represents the the object number. Then the activity-based PS model is modelled in euqation 6.7

$$APS(\rho, \theta, \alpha) = PS(\rho, \theta, \alpha) \cdot \Pi_i(1 - AS_i(\rho, \theta)) \tag{6.7}$$

we concern multiple objects in our scenario. Human can also doing multiple activities (eg., drink and watch TV). However, when S^a_{h,o_i} is low, we ignore the interaction activity. Meanwhile, drinking water has no effect on personal space model. Which is invisible in our model visualization. Our model is visualized in Figure 6.6. Supposing the person standing at the same position [35, 70], among different human-object interaction behaviors, the size and position of objects changes. We suppose the stopping distance of the single personal space model changes slightly based on the distance between human and objects, since we assume that people prefer the shorter distance of attention shifting from the object to the robot. However, no related studies examines our speculation, the detail of APS model on using a laptop is introduced in chapter 7.2.

We plot the probability distribution cross section pictures of old model in Figure 6.7a and new model in Figure 6.7b. Since the position with highest probability will be chosen as the end point of robot navigation, in the old model the end point is located in the middle value and in the new model the end point is located near the borders of activity spaces.

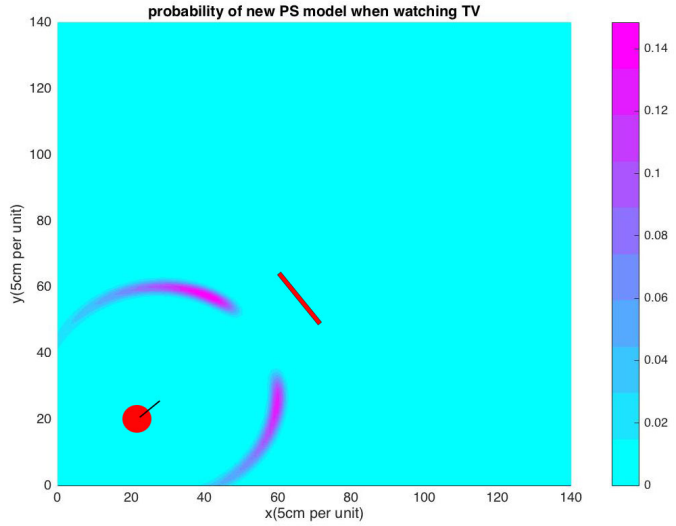
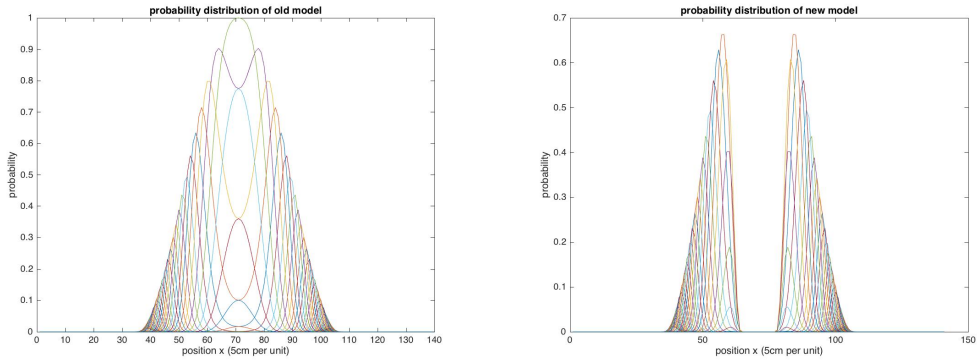


Figure 6.6: the activity-based personal space model of watching TV: (1) the image shows the probabilistic model where the person wants the robot to stop; (2) the red line presents the TV, the person locates at [35, 70]



(a) (b)

Figure 6.7: (a) shows the probability distribution cross section pictures of old model; (b) shows the probability distribution cross section pictures of new model

6.5 Application and Navigation

6.5.1 Simulation

We simulate the robots movement starting from there points with the angle facing to the human, the position values in polar coordinates were $x = [2.5m, -35°], [2.5m, 0°], [2.5m, 35°]$. During the navigation, the robot is driven by multiple forces: target force is to attach the robot go to the end point; obstacle avoidance force is to avoid the obstacle; gaze target force is to turn the robots orientation to the end point. Its worth noting here that the personal space probability updates when the robot is approaching. For a wider range of application, if the starting position of the robot is on the opposite side of the initial target stopping position, the navigation algorithm should concern the length of trajectory and avoid crossing the personal space. Otherwise, the robot behavior could confuse the person, since a normal human would never go across others visual field. We consider a weight expression to add the force of choosing a shorter trajectory, and the weight is given by:

$$\begin{aligned} w_j &= 1/(0.2 + d_j/10) \\ WAPS(\rho_j, \theta_j) &= w_j \cdot APS(\rho_j, \theta_j) \end{aligned} \tag{6.8}$$

where d_j presents the distance between the robot and the $j-th$ particle, $PS(\rho_j, \theta_j, \alpha, \xi)$ represents the probability value at the $j-th$ particle. $WAPS(\rho_j, \theta_j)$ allows the model to prevent going across the humans visual field. As shown in Figure 6.8, when the robot is far away from the particle, the robot choose the highest value in $APS(\rho_j, \theta_j)$ as the target, when d_j is very low, the robot motion considers a lot more on the human preference.

In the following, we take the situations of starting from the right side of the human for example and record the trajectory. During navigation, personal space model is used to find the endpoint. We take navigation behavior based on the new model without path force as a benchmark, then analyzed on the new model with path force. Then the trajectory of approaching from the right point based on the new model is shown in Figure 6.9a and 6.9b. The Figure 6.9a is moving without the path force and 6.9b with path force. By comparing the two navigation trajectory, the behavior of walking behind the TV in Figure 6.9a is regarded redundant and unlike a human. To examine the influence of path force, we chose the moment when the robot was standing at the position [90,70], and plotted the probability distribution. The top view of this world at this moment is depicted in Figure 6.10. The probability of cross section at y = 60 is shown in Figure 6.11, where 6.11a shows the probability of the new model without path force. As the Figure 6.11a shows the points at [58,60] and [84,60] have the highest probability of 0.6629. Figure 6.11b shows the probability of new model without path force, where the point at [85,60] has the highest probability of 0.043, the point at [58,60] has the probability of 0.025 which is highest at the left side but much lower than the point at [85,60]. As a result, if the robot navigates based on the model without path force, though the robot is approaching from the right side, the robot is possible to move to the left side of this human. That is to say, the side at which the robot starts has no relation to the side where it stops. Thus we can hardly expect at the robots trajectory and end point, whereas normally a humans behavior is predictable. Thus, the robots behavior conflicts with its identity as a social agent. By contrast, when the robot navigates based on the model with path force and the robot is approaching from the right side, the robot has a dominant force to go to the right side of the human. The starting side reflects a strong inclination to approach and stop on the same side.

6.5.2 Simulation with Noises

The world in real life is not as ideal as we depicted in Chapter 6.5.1. For instance, the accuracy of body orientation detection from the right front is 67% when the human is standing. We used

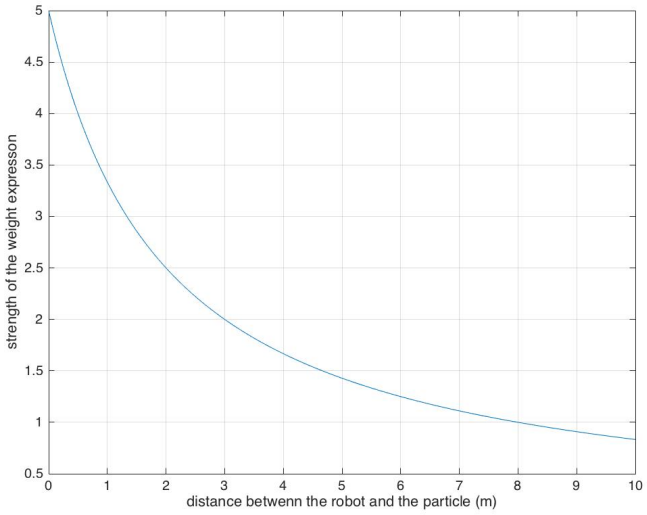
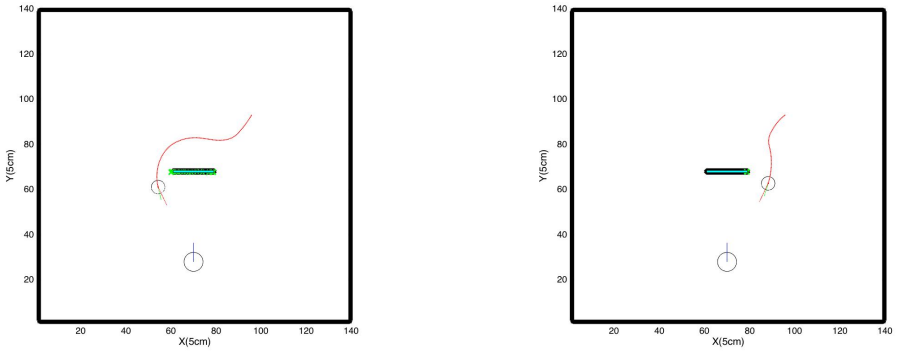


Figure 6.8: How the strength of the weight expression change along with the distance between the robot and particle



(a) (b)

Figure 6.9: (a) shows the trajectory of a robot approaching a human (located at [70,25]) based on the new model without path force; (b) shows the trajectory of a robot approaching a human (located at [70,25]) based on the new model with path force

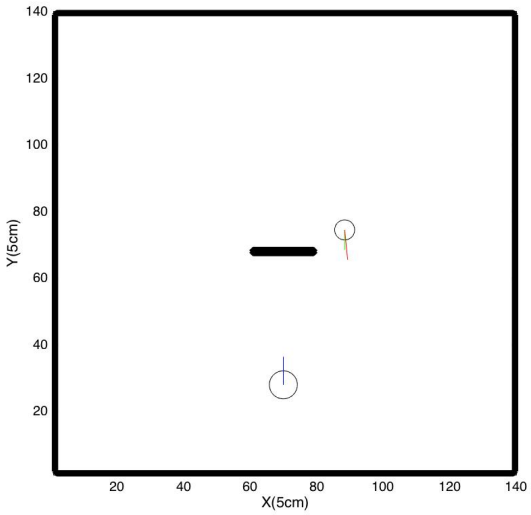
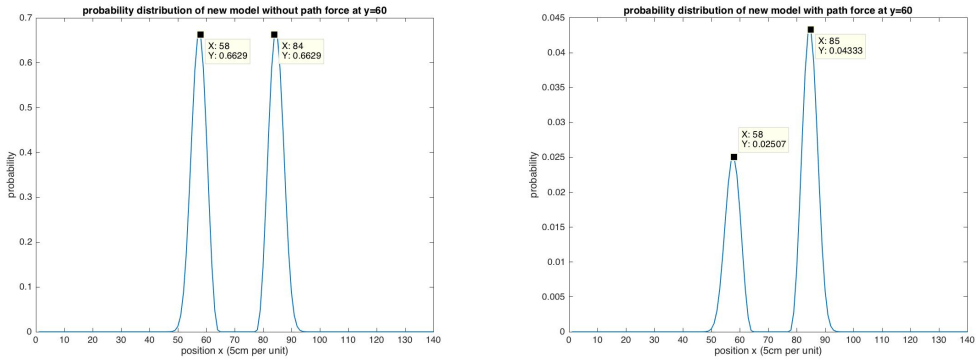


Figure 6.10: The top view of the world when the robot is located at [90,75]



(a) (b)

Figure 6.11: (a) shows the probability of new model without path force of the cross section at y=60, where the points at [58,60] and [84,60] have the highest probability of 0.6629; (b) shows the probability of new model with path force of the cross section at y=60, where the point at [85,60] has the highest probability of 0.043, the point at [58,60] has the probability of 0.025 which is highest at the left side but much lower than the point at [85,60].

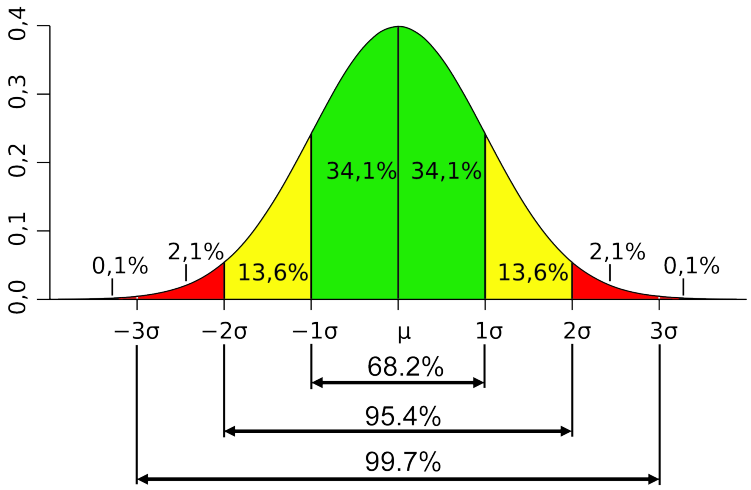


Figure 6.12: The probability density function of normal distribution. From "Standard deviation diagram" by J.Muelaner, 2013, http://www.muelaner.com/metrology/.

YSR (yaw success rate) as the criteria for orientation detection evaluation, which means the estimated yaw angle is within +/- 22.5 degree of the annotation. Thus, in reality, from the robot's perspective, the human's orientation jumps every frame within a certain range. Since the result in Chapter 5.4.1 shows that when the human is standing, the YSR is 66% seen from the right front and 75% from the left front. We examined the worse case, 66% of YSR, to test the usability of our detection method in our scenario. Given the 68-95-99.7 rule in normal distribution (see Figure. 6.12), the probability density between $-\sigma$ and $+\sigma$ is around 68.2%, which is slightly larger than our YSR 66%. Thus, we assumed the detected body orientation is a normal distribution with mean of the annotation and standard deviation of 22.5 degree. The simulation result of the robot starting from $x = [2.5m, -35°], [2.5m, 0°], [2.5m, 35°]$, which represent the left front, front, and right front respectively. From the simulation result (see Figure. 6.13), we find out a body orientation detection method with 68.2% success rate in detecting within 22.5 yaw degrees, robot can still conduct a reasonable approaching behavior. However, the optimistic result ignores the bad performance of human-object-interaction detection method when working on a distant interaction (like watching TV). Moreover, due to insufficient evaluation data, the distribution of detected results may not fit normal distribution. Thus, it's essential to build a more complete indoor-activity-scene dataset, to both train our algorithm and test.

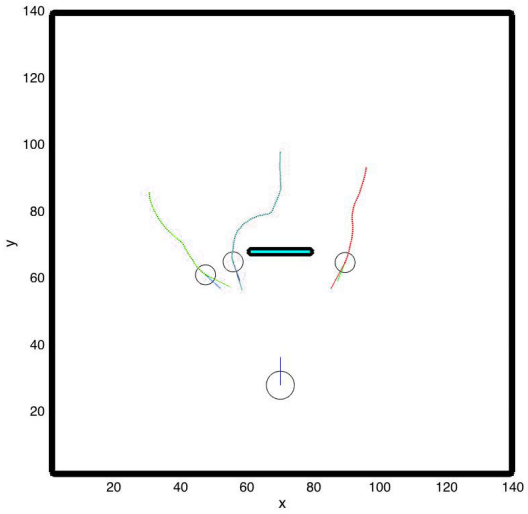


Figure 6.13: Trajectory of the robot's approaching the human from three sides. In these simulations, human body orientation is randomly generated from a normal distributions with mean -35 or 0 or 35 degrees, and variance 22.5 degree during the frames update.

Chapter 7

Discussion

7.1 Stopping Distance

Most previous studies chose stopping distance experiments to study the proxemic behavior of robots from human expectation. These stopping distance experiments yield the consistent result. When the robot approaches a human, the human tends to prefer the robot stopping in the personal space area (Obaid et al., 2016; Walters, 2008). The personal space model of a single person in open space has been reported by Torta et al. (Torta et al., 2011). When people are doing some activities, how does the optimal stopping distance change? In the same environment, when the human is engaged in doing other activities, like having a conversation with another(Ruijten & Cuijpers, 2017), the average optimal distance for a robot stop is smaller than single person setting (Torta et al., 2011). The stopping distance for the Nao robot is between 1.6m and 1.8m in the report of Torta et al. (2011), and 1.5m from the participants in the report of Ruijten and Cuijpers (2017). These results closely match, while the difference may be led by different human behavior. Because in the experimental setting of Ruijten and Cuijpers (2017), the subject is in the conversation, and the subject stands or sits still in the setting of Torta et al. (2011).

Though its widely accepted that most previous studies record the optimal stopping distance by asking the subjects to press the button to stop the robot (Walters et al., 2005; Torta, Cuijpers & Juola, 2013; Ruijten & Cuijpers, 2017). However, the procedure of this method is different from the scenario of a robot approaching a person to draw his attention. Visual attention is thought to be a two-stage process (Jonides, 1983) . In the first stage, attention codes the external visual scene uniformly and processes the information in parallel. In the second stage, attention is concentrated on a specific area of the visual scene. In many HRI cases, before the human notice the robot, they are engaged with other activities, in other words, theyre at the second stage of visual attention processing. Since the purpose of a robot approach is to draw peoples attention, namely, the robot is initially out of the visual spotlight. During this process, peoples spotlight, the item introduced by Eriksen and Hoffman (1972), will jump from the previous center to the robot (Eriksen & Hoffman, 1972). In our opinion, different from what happens in the stopping distance experiments, this draw-attention event will shorten (even leave out) the first stage of visual attention processing. In stopping distance experiments, subjects are performing a multitask which contains the primary task and observing robots. As a result, the robot is always caught by peoples spotlight during the experiments, just with less dwell time compared with the primary task.

Thus, a better experiment design is in demand. Our suggestion is to collect user evaluation on the robot approaching motion while the user performs tasks on the screen of different distance, the robot stopping distance is randomly programmed. Considering the time consuming, a VR envir-

onment may facilitate the experiment. Moreover, we can also examine whether the concentration level on human-object interaction affects stopping distance by manipulating the difficulty level of tasks.

7.2 The Influence of Human-Object Distance

Given the single-person PS model reported by Torta, et al. (2013)

$$PS(\rho,\theta|\alpha) = \alpha \cdot \eta_A \cdot \exp\left(-\frac{(\rho-\gamma_A(\theta))^2}{\delta_{\gamma_A}^2}\right) + (1-\alpha)\cdot\eta_B\cdot\exp\left(-\frac{(\rho-\gamma_B(\theta))^2}{\delta_{\gamma_B}^2}\right) \tag{7.1}$$

Our model considers the situations when human is interacting with an object, which is more common in daily life compared with a person standing or sitting still with no activity. We chose the behavior of reading a book, using a laptop, and watching TV as examples due to two reasons: (1)they are the most frequently happened indoor human-object behaviors; (2)they represent different distances between object and human. We speculate that the stopping distance of the single personal space model changes slightly based on the distance between human and objects. For different human-object interaction activities, of which the distance between human and objects varies. The purpose of robot approaching is to arouse human attention, speaking from the two-stage visual attention process (Jonides, 1983), the human would zoom in their attention in the specific area of an object, and then search on the external environment uniformly, in the end, zoom out their attention on the robot. A term named attention shifting distance is generated in this procedure, which presents the distance of attention shifting from the object to the robot. We assume that people prefer the shorter attention shifting distance. As a result, the stopping distance ρ in Torta, et al. (2013) model, should change positively with the change of human-object distance. This influence can be viewed as the behavior force, and the expression is given by:

$$h(d_{ho}) = \begin{cases} d_{ho}/1.2 & 0.6 \le d_{ho} \le 1.2 \\ 0.6/1.2 & 0.6 > d_{h0} \\ 1.2/1.2 & 1.2 \le d_{h0} \end{cases} \tag{7.2}$$

where d_{h0} represents the distance between human and the object, the range of $h(d_{ho})$ is supposed to be above the value 0.4 times the of single-person PS model meters due to safety issue, below 1.4 times the of single-person PS model due to taking this position as the farthest point to be interactive. The equation 7.2 is visualized in Figure 7.1.

For a position x=$[\rho,\theta]$ expressed in polar coordinates with respect to the users reference frame, the expression is given in equation.(9), where ρ represents the distance to the human position, θ represents the angular distance, represents the users state (sit or stand), d_{ho} represents the human-object distance. The single-person PS model is modified now by equation 7.3 as a activity-based individual PS model:

$$PS(\rho,\theta|\alpha,d_{ho}) = \alpha\cdot\eta_A\cdot\exp\left(-\frac{(\rho-h(d_{ho})\cdot\gamma_A(\theta))^2}{\delta_{\gamma_A}^2}\right)+(1-\alpha)\cdot\eta_B\cdot\exp\left(-\frac{(\rho-h(d_{ho})\cdot\gamma_B(\theta))^2}{\delta_{\gamma_B}^2}\right) \tag{7.3}$$

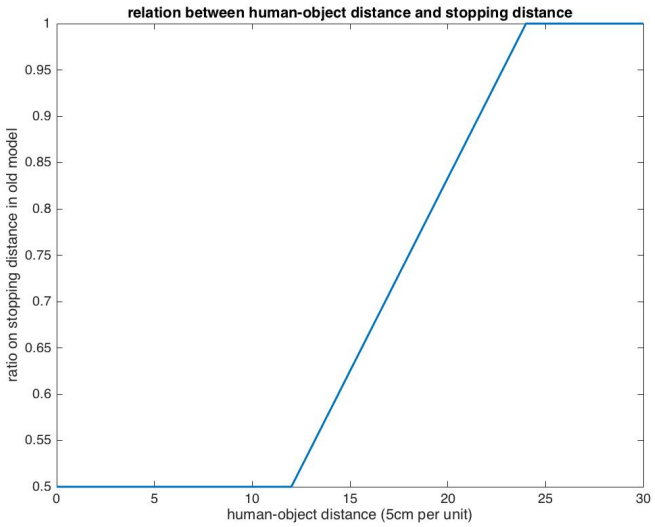
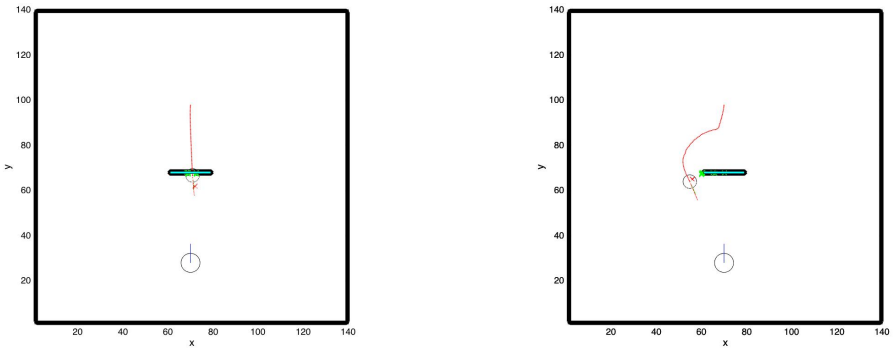


Figure 7.1: The force brought from the human-object distance

7.3 Robustness in Navigation Algorithm

An elusive error in obstacle avoidance force is found during our simulation when the robot starts from the middle point based on the old model. In Figure 7.2a, the robot starts from the middle point and goes straight to the endpoint crossing the TV, where the obstacle avoidance force seems invalid. In Figure 7.2b, the robot approaches based on the new model, the trajectory shows a smooth avoidance to the TV. We recorded the forces changing in the navigation process (see Figure 7.3), the obstacle avoidance is invalid because the turning force to any side is offset by the turning force to the other side, with the same magnitude and opposite direction. The exception happens when the line where the initial position of the robot and the endpoint are located is perpendicular to the plane of the obstacle (in our case, the TV). To fix the exception, we add a small positive number to the term of robots view direction in avoidance force function to make the robot always turn right when the exceptional condition happens. After our repair, the trajectory of navigation is shown in Figure 7.4a. The robot successfully avoids the TV and stops on the left side of the human. The force changing is shown in Figure 7.4b, once the avoidance force is valid, the robot will turn a small angle to the right. Gradually, the effect of the avoidance force accumulate, the robot can avoid TV under normal conditions. Compared with the navigation behavior based on the old model, the new model avoids the hidden flaw of previous avoidance algorithm. Even after fixing the flaw of avoidance algorithm, the Gaussian function of the activity space and blind area borders makes the robots trajectory quite smooth (see Figure 7.2b), which greatly prevents the hardware damage during motion, compared with the trajectory in new avoidance algorithm based on old model (see Figure 7.4a). Navigation is a behavior involving multiple forces, especially when running on a real robot, a small error may lead to chain-reaction of robots abnormal behaviors. Our new model is built by taking the inputs from computer vision, the behavior based on the new model is similar to how human perceive and conduct approaching behavior. In this way, our new model boost the robustness of navigation algorithm.



(a) (b)

Figure 7.2: (a) shows the trajectory of robot starts from the middle point based on old model, whereas the obstacle avoidance force is invalid; (b) shows the trajectory of robot starts from the middle point based on new model

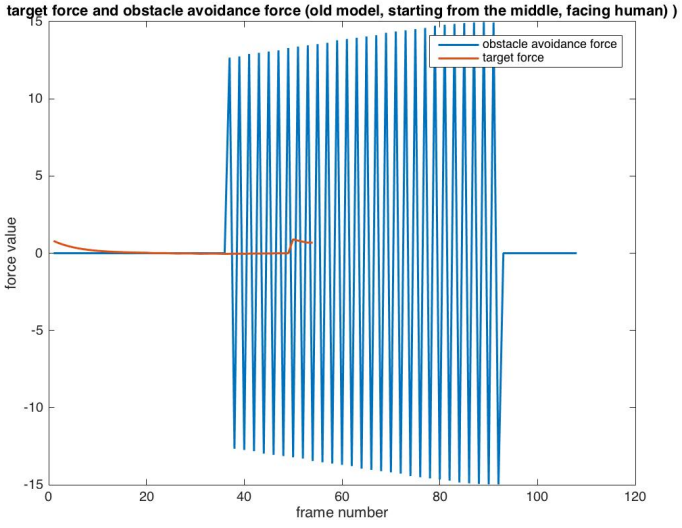
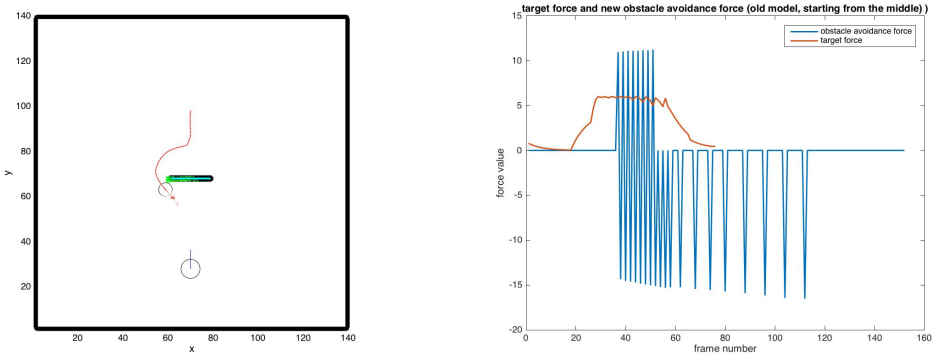


Figure 7.3: The target force and obstacle avoidance force changing during the navigation starting from the middle point based on old model

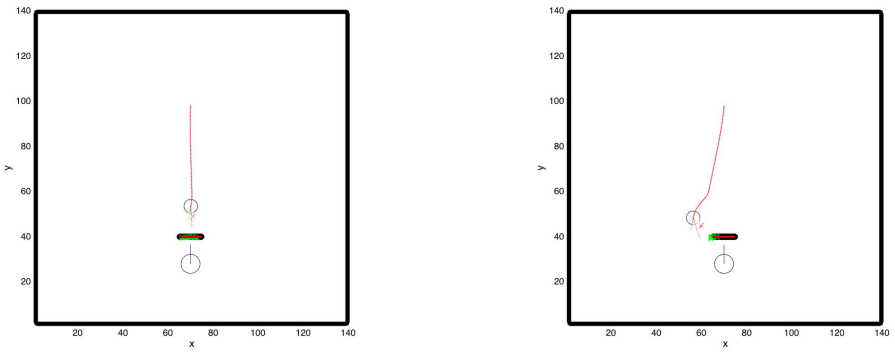


(a) (b)

Figure 7.4: (a) shows the trajectory of a robot approaches a human based on the old model with new obstacle avoidance force; (b) shows the obstacle avoidance force and target force changing during the approaching process.

7.4 Navigation Results on Using a Laptop

According to our speculation of the influence coming from different type of behavior(see chapter 7.2), we examined individual-person PS model and our model when the human is using a laptop. The robot stops just behind the laptop and is blocked by the screen(see in Figure 7.5a) based on improved old model, Figure 7.5b) shows the robot avoid the blind area based on new model. The new model makes the robots behavior more natural and human-like again. Due to the complexity of the real life (e.g., the diversity of environment and human behavior), we need a robot has the ability to adapt and work robustly, and our new model lifts robot's abilities.



(a) (b)

Figure 7.5: (a) shows the robot approaches the human from the middle point based on old model when the human is using a laptop; (b) shows the robot approaches the human based on new model from the middle point when the human is using a laptop; we applied path force and improved avoidance algorithm in both cases.

Chapter 8

Future work

There are some limitations to the method used for this thesis. As we discussed, our model is based on a limited stopping distance experiment, and we can improve the model with a new experimental method. Apart from improvements in stopping distance parameter, we supposed the relationship between human-object distance and stopping distance, and its yet to be examined in more experiments in the future. Currently, we only integrate the human detection (head orientation, body orientation, and pose estimation), while our thesis proposes the integrated detection framework by extracting the result of all the detection methods, which should be technically implemented in the future. To improve the performance of detection and better evaluation, building a larger dataset tailor to our application scenario for training is efficient and most recommended. Due to the common drawback of these computer algorithms, slow processing speed, a lot of effort will be spent on transplanting the algorithm into the robot hardware. To maximize the extension of robot hardware, we should consider to build a cloud platform for computation. After implementation on robots, we should validate the personal space model experimentally. The aim of the experiment is to show that the robot adapts its behavior depending on the participants activity. The participant will assess the robots behavior in the new and baseline model of personal space respectively.

Chapter 9

Conclusions

The thesis addresses the problem of a robot approaches a human naturally by detecting the environment.In chapter 2 and 3 , based on previous personal space models and F-formation model, we first discusses the configuration of personal space. We modified the previous personal space model by merging the influence of human-object interaction activity. We also state the importance of detecting humans activity, pose, and orientation to conduct a natural approaching movement based on our personal space model. In chapter 4, we discuss the related method for detection, and the methods we choose for our study. In chapter 5, we implement head orientation, body orientation, pose estimation, object detection, and human-object interaction detection. We evaluate the methods on our indoor-activity-scene data respectively. The result shows a proper accuracy for human detection, and a restricted human-object-interaction detection, which fails to detect the interaction between small objects and the distant interaction. In chapter 6, we visualize our personal space model and simulate robot navigation. By comparison, our model makes the robots approaching behavior more natural and human-like. Moreover, our model boosts the robustness of the navigation algorithm.

The simulation with noises still shows social navigation behavior from the robot, while the current evaluation result is pilot. First, we recommend to build a larger dataset for training and testing. Then,we suggest a combination of OpenPose skeleton detection result to distinguish the back and the front of human body. In the last, aiming at improving the human-object interaction detection, we suggest a revised spatial action score s^a_{sp} based on head orientation.

Our study applies the state-of-art computer vision methods to the robots social navigation, and the results show a more socially acceptable robot behavior. In the meanwhile, our study builds the structure of robot social navigation based on computer vision, and the further works can implement the visual SLAM on the robot to reach the ultimate goal – move like a human.

References

Ahn, B., Park, J. & Kweon, I. S. (2014). Real-time head orientation from a monocular camera using deep neural network. In *Asian conference on computer vision* (pp. 82–96). 22

Belhumeur, P. N., Jacobs, D. W., Kriegman, D. J. & Kumar, N. (2013). Localizing parts of faces using a consensus of exemplars. *IEEE transactions on pattern analysis and machine intelligence*, *35*(12), 2930–2940. 21

Bishop, C. M. et al. (1995). *Neural networks for pattern recognition*. Oxford university press. 21

Borkowski, A., Siemiatkowska, B. & Szklarski, J. (2010). Towards semantic navigation in mobile robotics. In *Graph transformations and model-driven engineering* (pp. 719–748). Springer. 19

Cao, Z., Simon, T., Wei, S.-E. & Sheikh, Y. (2016). Realtime multi-person 2d pose estimation using part affinity fields. *arXiv preprint arXiv:1611.08050*. 11, 22

Chao, Y.-W., Liu, Y., Liu, X., Zeng, H. & Deng, J. (2018). Learning to detect human-object interactions. In *2018 ieee winter conference on applications of computer vision (wacv)* (pp. 381–389). 23

Choi, J., Lee, B.-J. & Zhang, B.-T. (2016). Human body orientation estimation using convolutional neural network. *arXiv preprint arXiv:1609.01984*. 22

Chung, S.-Y. & Huang, H.-P. (2010). A mobile robot that understands pedestrian spatial behaviors. In *Intelligent robots and systems (iros), 2010 ieee/rsj international conference on* (pp. 5861–5866). 11

Ciolek, T. M. & Kendon, A. (1980). Environment and the spatial arrangement of conversational encounters. *Sociological Inquiry*, *50*(3-4), 237–271. 20

Eriksen, C. W. & Hoffman, J. E. (1972). Temporal and spatial characteristics of selective encoding from visual displays. *Perception & psychophysics*, *12*(2), 201–204. 51

Gao, C., Zou, Y. & Huang, J.-B. (2018). ican: Instance-centric attention network for human-object interaction detection. *arXiv preprint arXiv:1808.10437*. 12, 23, 37

Gavrila, D. M. & Davis, L. S. (1996). 3-d model-based tracking of humans in action: a multi-view approach. In *Proceedings cvpr ieee computer society conference on computer vision and pattern recognition* (pp. 73–80). 22

Girshick, R. (2015). Fast r-cnn. In *Proceedings of the ieee international conference on computer vision* (pp. 1440–1448). 23

Gkioxari, G., Girshick, R., Dollár, P. & He, K. (2017). Detecting and recognizing human-object interactions. *arXiv preprint arXiv:1704.07333*. 23

Gourier, N., Hall, D. & Crowley, J. L. (2004). Estimating face orientation from robust detection of salient facial structures. In *Fg net workshop on visual observation of deictic gestures* (Vol. 6, p. 7). 26, 28

Gupta, S. & Malik, J. (2015). Visual semantic role labeling. *arXiv preprint arXiv:1505.04474*. 23

Hayduk, L. A. (1981). The shape of personal space: An experimental investigation. *Canadian Journal of Behavioural Science/Revue canadienne des sciences du comportement*, *13*(1), 87. 13

Huang, J., Rathod, V., Sun, C., Zhu, M., Korattikara, A., Fathi, A., ... others (2017). Speed/accuracy trade-offs for modern convolutional object detectors. In *Ieee cvpr* (Vol. 4). 23

Hüttenrauch, H., Eklundh, K. S., Green, A. & Topp, E. A. (2006). Investigating spatial relationships in human-robot interaction. In *Intelligent robots and systems, 2006 ieee/rsj international conference on* (pp. 5052–5059). 14

Jonides, J. (1983). Further toward a model of the minds eyes movement. *Bulletin of the Psychonomic Society*, *21*(4), 247–250. 51, 52

Kendon, A. (1976). The f-formation system: The spatial organization of social encounters. *Man-Environment Systems*, *6*, 291–296. 17

Kuzuoka, H., Suzuki, Y., Yamashita, J. & Yamazaki, K. (2010). Reconfiguring spatial formation arrangement by robot body orientation. In *Proceedings of the 5th acm/ieee international conference on human-robot interaction* (pp. 285–292). 14

Le, V., Brandt, J., Lin, Z., Bourdev, L. & Huang, T. S. (2012). Interactive facial feature localization. In *European conference on computer vision* (pp. 679–692). 21

Lindner, F. & Eschenbach, C. (2014). Affordances and affordance space: A conceptual framework for application in social robotics. In *Sociable robots and the future of social relations, 1st international conference, robo-philosophy* (pp. 35–45). 18

Mikic, I., Trivedi, M., Hunter, E. & Cosman, P. (2001). Articulated body posture estimation from multi-camera voxel data. In *Computer vision and pattern recognition, 2001. cvpr 2001. proceedings of the 2001 ieee computer society conference on* (Vol. 1, pp. I–I). 22

Mitsuhashi, N., Fujieda, K., Tamura, T., Kawamoto, S., Takagi, T. & Okubo, K. (2008). Bodyparts3d: 3d structure database for anatomical concepts. *Nucleic acids research*, *37*(suppl_1), D782–D785. 30

Newell, A., Yang, K. & Deng, J. (2016). Stacked hourglass networks for human pose estimation. In *European conference on computer vision* (pp. 483–499). 22

Obaid, M., Sandoval, E. B., Złotowski, J., Moltchanova, E., Basedow, C. A. & Bartneck, C. (2016). Stop! that is close enough. how body postures influence human-robot proximity. In *Robot and human interactive communication (ro-man), 2016 25th ieee international symposium on* (pp. 354–361). 11, 51

Ostermann, F. & Timpf, S. (2007). Modelling space appropriation in public parks. In *Proceedings of the 10th agile international conference on geographic information science.* 17

Park, S. & Trivedi, M. M. (2007). Multi-person interaction and activity analysis: a synergistic track-and body-level analysis framework. *Machine Vision and Applications*, *18*(3-4), 151–166. 13

Redmon, J., Divvala, S., Girshick, R. & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 779–788). 22

Rios-Martinez, J., Spalanzani, A. & Laugier, C. (2014, sep). From proxemics theory to socially-aware navigation: A survey. *International Journal of Social Robotics*, *7*(2), 137–153. Retrieved from https://doi.org/10.1007/s12369-014-0251-1 doi: 10.1007/s12369-014-0251-1 11, 17, 40

Ruijten, P. A. M. & Cuijpers, R. H. (2017, aug). Stopping distance for a robot approaching two conversating persons. In *2017 26th IEEE international symposium on robot and human interactive communication (RO-MAN).* IEEE. Retrieved from https://doi.org/10.1109/roman.2017.8172306 doi: 10.1109/roman.2017.8172306 51

Sagonas, C., Tzimiropoulos, G., Zafeiriou, S. & Pantic, M. (2013, dec). 300 faces in-the-wild challenge: The first facial landmark localization challenge. In *2013 IEEE international conference on computer vision workshops.* IEEE. Retrieved from https://doi.org/10.1109/iccvw.2013.59 doi: 10.1109/iccvw.2013.59 21

Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., ... Blake, A. (2011, jun). Real-time human pose recognition in parts from single depth images. In *CVPR 2011.* IEEE. Retrieved from https://doi.org/10.1109/cvpr.2011.5995316 doi: 10.1109/cvpr.2011.5995316 22

Smith, B. A., Yin, Q., Feiner, S. K. & Nayar, S. K. (2013). Gaze locking: passive eye contact detection for human-object interaction. In *Proceedings of the 26th annual acm symposium on user interface software and technology* (pp. 271–280). 21

Sugano, Y., Matsushita, Y. & Sato, Y. (2014, June). Learning-by-synthesis for appearance-based 3d gaze estimation. In *The ieee conference on computer vision and pattern recognition (cvpr).* 22

Sutherland, I. (1974). Three-dimensional data input by tablet. *Proceedings of the IEEE*, *62*(4), 453–461. Retrieved from https://doi.org/10.1109/proc.1974.9449 doi: 10.1109/proc.1974.9449 28

Tateno, K., Tombari, F., Laina, I. & Navab, N. (2017). Cnn-slam: Real-time dense monocular slam with learned depth prediction. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)* (Vol. 2). 12

Tompson, J. J., Jain, A., LeCun, Y. & Bregler, C. (2014). Joint training of a convolutional network and a graphical model for human pose estimation. In *Advances in neural information processing systems* (pp. 1799–1807). 22

Torta, E., Cuijpers, R. H. & Juola, J. F. (2013, may). Design of a parametric model of personal space for robotic social navigation. *International Journal of Social Robotics*, *5*(3), 357–365. Retrieved from https://doi.org/10.1007/s12369-013-0188-9 doi: 10.1007/s12369-013-0188-9 51

Torta, E., Cuijpers, R. H., Juola, J. F. & van der Pol, D. (2011). Design of robust robotic proxemic behaviour. In *Social robotics* (pp. 21–30). Springer Berlin Heidelberg. Retrieved from https://doi.org/10.1007/978-3-642-25504-5_3 doi: 10.1007/978-3-642-25504-5_3 11, 51

Walters, M. L. (2008). *The design space for robot appearance and behaviour for social robot companions* (Unpublished doctoral dissertation). 11, 51

Walters, M. L., Dautenhahn, K., Koay, K., Kaouri, C., Woods, S., Nehaniv, C., ... Werry, I. (2005). The influence of subjects' personality traits on predicting comfortable human-robot approach distances. In *Proceedings of cog sci 2005 workshop: Toward social mechanisms of android science* (pp. 29–37). 51

Weinrich, C., Vollmer, C. & Gross, H.-M. (2012). Estimation of human upper body orientation for mobile robotics using an svm decision tree on monocular images. In *Intelligent robots and systems (iros), 2012 ieee/rsj international conference on* (pp. 2147–2152). 22

Wilson, H. R., Wilkinson, F., Lin, L.-M. & Castillo, M. (2000). Perception of head orientation. *Vision research*, *40*(5), 459–472. 21

Wood, E., Baltrusaitis, T., Zhang, X., Sugano, Y., Robinson, P. & Bulling, A. (2015). Rendering of eyes for eye-shape registration and gaze estimation. In *Proceedings of the ieee international conference on computer vision* (pp. 3756–3764). 22

Yao, B. & Fei-Fei, L. (2010). Modeling mutual context of object and human pose in human-object interaction activities. In *Computer vision and pattern recognition (cvpr), 2010 ieee conference on* (pp. 17–24). 23

Zhang, X., Sugano, Y., Fritz, M. & Bulling, A. (2015). Appearance-based gaze estimation in the wild. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 4511–4520). 22