IMPROVING VISUAL FOCUS OF ATTENTION OF CHILDREN WITH ASD DURING
PROMPTED TASK EXECUTION USING NAO HUMANOID ROBOT

by

Di Xue

A thesis submitted in conformity with the requirements
for the degree of M.A.Sc
Graduate Department of IBBME
University of Toronto

Abstract

Improving Visual Focus of Attention of Children with ASD during Prompted Task Execution using
NAO Humanoid Robot

Di Xue

M.A.Sc

Graduate Department of IBBME

University of Toronto

2015

Contents

Chapter 1

Introduction

1.1 Motivation

1.1.1 Children with ASD and the need for ATCs

Autism Spectrum Disorder (ASD) is a complex neurological and developmental condition that is characterized by a triad of impairments: impaired social interactions, impaired non-verbal and verbal communication, and restricted or repetitive patterns of behaviour (Frith & Happ, 2005). Recent research suggested that the prevalence of ASD, when the broad spectrum is considered, is as high as 1:91 in the United States (Kogan et al., 2009).

Children with ASD are often slower than their neurotypical peers in learning self-management skills, and may need constant assistance from caregivers for carrying out activities of daily living (ADLs). Often such burden is carried by an informal caregiver, such as an immediate family member. Due to the overwhelming burden and stress, decreasing quality of life may be a result (Burgess & Gutstein, 2007).

The alternative to caregiver assistance is assistive technologies for cognition (ATCs). Such technologies aim to assist, instruct, train, or educate individuals with cognitive disabilities to overcome challenges in life, including executing activities of daily living (ADL). Especially, autonomous or self-operated ATCs promote independence of children with ASD, alleviating caregiver burdens. Commercially available devices such as computers, video players, audio players, cellphones, etc. have all been exploited as platforms for ATCs.

1.1.2 COACH and its Challenges

The COACH (Cognitive Orthosis for Assisting with aCtivites in the Home) system, developed by Mihailidis et al., is an autonomous prompting system (Mihailidis, Boger, Craig, & Hoey, 2008). It uses computer vision and artificial intelligence to automatically detect user actions when executing ADLs, and prompts appropriately when user needs assistance. It was first developed for the dementia population, but a version appropriate to the ASD population was recently adapted and tested in a pilot study (Bimbrahw, Boger, & Mihailidis, 2012).

The system currently uses audio and video prompts using an LCD monitor screen as its primary prompting modalities. In the pilot study, the hand-washing activity is used as an example to test the system's effectiveness because of the simplicity of its tasks as well as the washroom settings being easily

controlled. The hand-washing activity is broken into five steps, with verbal prompts being: "turn on the water and wet your hands", "put soap on your hands", "scrub your hands", "rinse your hands", "turn off the water and dry your hands". At each step, if the child does not successfully execute the correct action within a time threshold, the system prompts. The prompting consist of first displaying a still image attention grabber on screen that attempts to capture child's attention, then playing the pre-recorded audio and/or video prompts. The video prompts are videos recorded from point-of-view (first person) perspective of a person executing the step. If child successfully executes a step with or without prompting, the system congratulates by saying "Good job". If the child doesn't get it right within a time threshold, the system prompts again. This is repeated until a predetermined maximum number of times before calling over the caregiver to assist.

The pilot study of COACH for ASD showed good acceptance by the children with ASD and their caregivers (Bimbrahw et al., 2012) . System effectiveness was shown in that 78% of the hand-washing steps were completed without caregiver assistances by the children themselves, all of whom did not know how to wash-hands on their own before.

The major area of improvement identified in the study was to increase child's engagement during prompting and task execution. Firstly, almost half of the system's prompts were ignored by the children. Assuming that these children were able to understand the prompts, as seen by their correct actions to the not ignored prompts, their noncompliance is a reflection of their disinterest in the prompts. Secondly, several occasions of child being distracted or disinterested to tasks were observed.

To tackle this problem, two approaches can be applied: 1) change the prompting modality to one that's inherently more engaging to the child (e.g. humanoid robot); 2) track the child's visual focus of attention in real-time, and issue attention grabbers when child is distracted.

1.2 Roadmap

This thesis will address how to improve the visual focus of attention of children with ASD when prompted by COACH. It will approach the problem by incorporating a humanoid robot, NAO, and implementing algorithms to track and react to the child's gaze, in attempts to better engage children during ADL prompting sessions.

The thesis report will be broken into the following sections:

In Chapter 2, we first review the relevant literature on the established frameworks for ASD interventions, especially those that teach skills of daily living, and relate it to our prompting framework. Next, we review research for ASD interventions that utilize technologies such as videos, computers, and robots, confirming this thesis' approach and identifying the research gap. Also, our research direction is put into context in the Human Robot Interaction (HRI) field. Lastly, we review state of the art research in visual focus of attention estimation, targeting key methodologies appropriate for this thesis's goal.

In Chapter 3, the research objectives and hypotheses are summarized.

In Chapter 4, the Wizard of Oz pilot study is presented. The study setup, protocol, data collected, analysis and results are discussed in details.

In Chpater 5, the investigated algorithms for tracking a person's visual focus of attention are presented. Specifically, the problem is broken into head pose estimation, eye pose estimation, and object identification. The methods and evaluation for each are discussed and evaluated.

In Chapter 6, we summarize and conclude the thesis.

Chapter 2

Literature Review

2.1 ASD Interventions

Currently, there is no cure for ASD, with all interventions aimed to help children with ASD adjust more effectively to their environment (Francis, 2005). There are many approaches to ASD interventions, such as behaviour based, dietary based, pharmaceutically based, speech therapy based, physical therapy based, etc. One reason for such variety is because of the varying nature of ASD disabilities across individuals, making one single method not effectively for the whole population, and sometimes not sufficient for an individual. Consequently, it remains a challenge to show effectiveness of a method in a clinically significant sense, even if it is truly effective to a small sample of the population. The most clinically tested, commonly practiced, and recommended methods are those based on the principles of Applied Behaviour Analysis (Foxx, 2008).

2.1.1 Applied Behaviour Analysis

Applied Behaviour Analysis (ABA) is based on scientifically proven theories of human behaviours, and is widely used for treating inappropriate behaviours and teaching new behaviours to people with mental disabilities. Its strategies for treating inappropriate behaviours include: finding and changing antecedents to inappropriate behaviours, ignoring the behaviour, or negative reinforcement through punishment (Foxx, 1982). Its strategies for teaching new behaviours include: giving stimuli to child to elicit a new behavior, positive reinforcement (e.g. praise), and maintenance and generalization strategies to make sure the newly learned behaviour is retained across time and settings (Foxx, 1982). For cueing stimuli, high level of consistency is needed. For positive reinforcement, individually selected and strategically used motivators (e.g. praise, a hug, a check mark, a favourite activity) should be given immediately after appropriate behaviours. For maintenance and generalization strategies, prompt fading and testing across context and settings can be used.

For children with ASD, research has shown that early ABA based intervention with persistent and intensive sessions of minimum 30 hours a week for 2 years are proven to be successful for improving ASD outcomes (Howlin, Magiati, & Charman, 2009).

Discrete Trial Training for Skill Teaching

Discrete Trial Training (DTT) is an ABA based method widely used for teaching children with ASD new behaviours, and particularly useful in teaching skills of daily living. There are four steps to a DTT (Bogin, 2010):

1. Grabbing the child's attention

2. Discriminative stimulus: must be simple, clear, and concise, and wording must be consistent

3. Child's response: stop incorrect response by instructing again immediately or provide prompts

4. Prompt to aid the child if no response or wrong: this has the advantage of helping him quickly move through trials and avoid boredom and frustration. Always use least intrusive prompt that ensures correct response. Fade prompts to promote maintenance of effect and prevent dependence on intervention. Types of prompts include: verbal, gesture, modeling, visual, physical.

5. Reinforcement: given immediately after correct responses, tailored individually, always pair with verbal praise.

2.1.2 Discussion

To promote consistency and effectiveness, the implementations of ATCs for assisting children with ASD with daily living tasks should also follow the ABA framework, and model after the DTT steps. In fact, this is exactly how COACH prompts:

1. Attention Grabber

2. Verbal Instruction

3. Video Prompt

4. Child's Response

5. Verbal Reward

For prompting modalities, gesture and physical may be candidates for further exploration.

2.2 Assistive Technology for ASD

Lewis describes assistive technology as any technology that can enhance the performance of persons with disabilities by augmenting an individual's strengths or providing an alternative mode of performing a task (Lewis, 1998). In a review by Goldsmith et al., such technologies for ASD population include the following (Goldsmith & LeBlanc, 2004):

2.2.1 Tactile Prompting Devices

Tactile prompting was mainly used to prompt ASD individuals for initiating interactions with other people. Taylor et al. used such devices for prompting teens with ASD to seek assistance when lost (Taylor, Hughes, Richard, Hoch, & Coello, 2004). Shabani et al. showed increased verbal initiations when tactile prompting were used (Shabani et al., 2002).

2.2.2 Auditory Prompting Devices

Taber et al. used auditory prompting to decrease off-task behaviour for a student with ASD. Verbal prompts (e.g. keep working", "pay attention", etc.) were played in between recorded music in a prompting system. The result was successful decrease in off-task behaviour and teacher-delivered prompts (Taber, Seltzer, Heflin, & Alberto, 1999).

2.2.3 Computer-Aided Instruction

Computers have been used to teach a variety of skills, including how to recognize and predict emotions (Silver & Oakes, 2001), enhance problem solving (Bernard-Opitz, Sriram, & Nakhoda-Sapuan, 2001), improve vocabulary (D. Moore & Taylor, 2000), advance generative spelling (Kinney, Vedora, & Stromer, 2003), enhance vocal imitation (Bernard-Opitz, Sriram, & Sapuan, 1999).

Compared with live personal instructions, Chen et al. showed that computer-assisted instruction resulted in better motivation and fewer behaviour problems (Chen & Bernard-Opitz, 1993). Similar results were observed by Moore et al., where children with ASD were paying more attention when instructed by computers (M. Moore & Calvert, 2000). In Lahm's study, children with disabilities, including ASD, were shown to engage more frequently with the computer when the program contains higher interaction requirements, animations, sounds, and voice (Lahm, 1996).

2.2.4 Video-Based Modeling and Feedback

Video technology has proven useful as a tool for modeling appropriate behaviour, providing feedback, and creating discrimination opportunities for the child's own behaviour, and as a medium for presenting basic instruction that many children with ASD find engaging (Sturmey, 2003). Video based modeling and feedback has been effectively used to teach conversational speech (Sherer et al., 2001), increase task fluency (Lasater & Brady, 1995), increase play related statements (Taylor, Levin, & Jasper, 1999), improve social communication (Thiemann & Goldstein, 2001), teach daily living skills (Shipley-Benamou, Lutzker, & Taubman, 2002), improve perception of emotion (Corbett, 2003).

Here are some relevant results of video modeling studies: video modeling has been shown to promote generalization and maintenance of the skills taught (Charlop & Milstein, 1989) and to result in superior skill generalization compared to live modeling (Charlop-Christy, Le, & Freeman, 2000). Point-of-view perspective is preferred than third person perspective, and prompting videos showing before execution of each step is more effective than priming videos showing the whole sequence of steps altogether (Mason, Davis, Boles, & Goodwyn, 2013). Lastly, videos are more effective than still pictures (Van Laarhoven, Kraus, Karpman, Nizzi, & Valentino, 2010).

2.2.5 Discussion

As seen from the review, video prompts are particularly effective for teaching daily living skills. This is COACH's current primary prompting modality. It uses point-of-view perspective for recording the videos, just like the literature suggested. In our study, although we will explore and compare effectiveness of other prompting modalities, we believe video modeling would still be essential in conveying complex task execution techniques to the child.

2.3 Human Robot Interactions (HRI)

HRI is a recent field of research focusing on natural and efficient interactions between human and robot. It is a multidisciplinary field with contributions from human computer interactions (HCI), artificial intelligence, robotics, design, social sciences, etc. The study of HRI is important for designers of robot behaviours, especially if one pursuits a framework of user centered design, where user experience is given as much value as functionalities.

2.3.1 Socially Interactive Robotics (SIR)

SIR is first defined by Fong et al. as robots for the main purpose of interacting with user without physical contact (Fong, Nourbakhsh, & Dautenhahn, 2003). The application domains where SIR are desirable are: robot as mediator of human-human interactions, robot as representations of humans, robot as companions to human, robot as modeling tool for researchers studying embodied social behaviours. The kinds of social interactions SIR could simulate include: artificial emotions, speech, facial expression, body language and gesture. We will refer to these as different modes of human-robot interactions through out this thesis.

2.3.2 The Issue of Embodiment

One major issue in the study of SIR (and that of HIR in general) is the issue of embodiment, i.e. what effects does the robot's physical presence have on its interactions with humans. Ziemke (Ziemke, 2001) defined four levels of embodiment from least to most:

1. **Structural coupling**: the presence (doesn't have to be physical) of the agent can influence a human's state

2. **Physical embodiment**: the agent has a physical body

3. **Organismoid embodiment**: the agent's physical body is organism-like

4. **Organismic embodiment**: except the agent's organism-like body is of autopoietic, living systems

The impact of embodiment in SIR is significant, as argued by Mataric (Mataric, 2005). Because humans irrepressibly attribute human-like characteristics to embodied agents that are similar to them, the factors influencing the embodiment of the robot will impact significantly the engagement of the human in HRI.

Similar intuition is explored by Young et al (Young et al., 2011). As Young et al. pointed out, because the robot has an embodied presence in close proximity to the user, it has a greater influence than the disembodied counterpart on the user physically, emotionally, and socially. Thus, factors in embodiment has a large influence on how the user perceives the robot's identity and their relationship. Young et al. provided a three level perspective on how to analyze these factors in embodiment in context of HRI:

1. **Intrinsic level**: static factors regarding the appearance of the robot set the initial emotional response from the user. For example, a cute looking robot animal may create instant affinity from children towards the robot. Static qualities of the robot's voice and motion (e.g. amplitude, smoothness, speed, etc.) fall under this level, too. In addition, factors in this level contribute

to the role user perceives the robot to be in. For example, a humanoid is better than an animal looking robot as a supervisor or mentor.

2. **Behaviour level**: dynamic factors regarding the facial expressions, voice intonations, motion gestures and gaze, etc. modulate emotional responses from the user and interactions between user and robot. In addition, these factors contribute to the robot's perceived role as well.

3. **Role level**: the goal of the robot, and the planning and decision making of the robot's behaviours to achieve that goal (along with factors from previous two levels), determines the perceived role of the robot. The user may treat the robot as a supervisor, mentor, companion, cooperative peer, slave, tool, etc. (Goodrich & Schultz, 2007).

2.3.3   Socially Assistive Robotics (SAR)

Feil-Seifer and Mataric defined SAR as the intersection of socially interactive robotics (SIR) and assistive robotics (AR) (Feil-Seifer & Mataric, 2005). In the past, AR has been mainly been used to describe robots that assisted people with physical disabilities through physical contacts. But SAR arises from the need for robots to assist people with cognitive disabilities without using physical contacts. In this context, SAR assumes an expert (mentor / coach / supervisor / teacher / assistance) role, and prompts the user through executions of tasks. Feil-Seifer and Mataric pointed out that SAR has two (possibly conflicting) goals - engaging the user to HRI and engaging the user to executing tasks. SAR needs careful designing to satisfy both the social interaction goal and the assistive goal. However, on a deeper level, the SIR goal is serving the AR goal in that, better social interactions would cause better user engagement during the co-operative activity, and yield better user compliance to prompts and ultimately better task performance.

Wainer et al. conducted studies along this line of hypothesis, evaluating user engagement through self-reflective survey and evaluating task performance through optimality of move during the task of solving a Towers of Hanoi puzzle (Wainer, Feil-Seifer, Shell, & Mataric, 2007). They tested the difference between embodied robot versus its remote presence and its simulated virtual avatar (both disembodied versions are displayed on a computer screen). They have found that people prefer interacting with the embodied robot and reported it being more helpful and watchful than the other two disembodied counter parts. However, there were no significant improvements in task performances from using the embodied robot over the disembodied versions.

2.3.4   Discussion

This lack of impact on task performance in Wainer et al.'s study may be due to the learning effect when solving the puzzle and the novelty factor of using the robot. Also, the use of such SAR on the clinical population and on clinically relevant tasks may yield a better result because of greater need for user engagement and user compliance in that context. For our thesis, we will investigate the impact of embodiment on the clinical population of children with ASD in the task of hand-washing, where user's lack of compliance to prompts and of engagement in task executions are observed problems. One thing to note is, we cannot conduct self-reflective surveys due to the nature of the children with ASD population. Therefore, we will rely on objective measures for user engagement such as verbal responses and physical behaviours (proximity, posture, eye gaze and xation) (Mataric, 2005).

2.4 Robots in ASD Interventions

A recent popular area of research for ASD interventions is robot-based interventions. Clinical studies of this approach is reviewed by Diehl et al., and the studies have been categorized based on the study purposes (Diehl, Schmitt, Villano, & Crowell, 2012):

2.4.1 General Response to Robots

Several studies have investigated how individuals with ASD interact with robots in general, and have compared it with human counterparts. Dautenhahn et al. showed that some individuals with ASD prefer interactive robots compared to passive toys (Dautenhahn & Werry, 2004). Robins et al. showed that ASD individuals prefer robot-like characteristics over human-like in social interactions (Robins, Dautenhahn, & Dubowski, 2006). Pierno et al. found that children with ASD respond faster to robotic movement than human movement (Pierno, Mari, Lusher, & Castiello, 2008).

2.4.2 Eliciting Pro-Social Behaviours

Robins et al. showed that children showed increased social interaction behaviours towards the robot in some areas such as proximity to robot, eye gaze, touch, imitation (Robins, Dautenhahn, Te Boekhorst, & Billard, 2005). Feil-Seifer et al. suggested that social behaviours of children with ASD increased when the robot is acting contingently to child's actions, as opposed to randomly (Feil-Seifer & Matari, 2009). Ravindra et al. showed the potential for robot and ASD individuals to share joint attention towards objects (Ravindra et al., 2009).

2.4.3 Modeling, Teaching, or Practicing a Skill and Providing Reinforcements

Not many studies have been conducted for skill execution training, since most studies have focused on addressing the social aspects of ASD interventions. One preliminary study by Duquette et al. examined the use of humanoid robot to help children with ASD practice imitation behaviours (Duquette, Michaud, & Mercier, 2008). Positive reinforcements were also given through the robot raising arms and saying "Happy!" when the child successfully imitated the robot. In their study result, children with ASD showed more interest to the robot when the robot is mediating the training session, as compared to their interest during human mediated sessions. Also, in the robot's presence, fewer repetitive behaviours with their favourite toy were observed.

2.4.4 Discussion

Of particular interest to this thesis is Ravindra et al.'s study (Ravindra et al., 2009), in which they implemented a detector of object of visual focus of attention. Using head pose plus eye pose estimations, and mixture of Gaussian modeling of object locations, they were able to detect with 75% accuracy the object of the child's gaze (i.e. visual focus of attention). The detector is then used as a feedback to the robot's prompting system for instructing the child to interact with objects through verbal command and gesture prompt (robot points and gazes at the object).

Even though their study's purpose is to train children with ASD with joint attention skills, their study showed the plausibility of utilizing similar techniques for prompting the child to pay attention to or interact with any objects of interest. Particularly in our thesis, we aim to implement two aspects of their techniques to guide children with ASD through hand-washing:

- **Robot gesture prompt**: point and gaze at an object while instructing the child how to interact with the object.

- **Robot behaviour loop**: loop consisting of attention grabber (AG), prompt, reinforcement if instruction successfully followed, loops from AG again if failure (illustrated in Figure 1[FIGURE]).

In Diehl et al.'s review, it was pointed out that studies in skill training would benefit from integrating robots into the ABA framework. As discussed in 2.1.2[CHANGE TO SECTION LINK], this is what we propose any skill training ATC system should do, similar to COACH. Thus including a robot into our COACH system would fill this research gap.

In all, from these studies, we see a great potential for increasing the attention level of the ASD child if we incorporate a humanoid robot into our COACH system. The robot may serve as an attention grabber as well as a prompting agent.

2.5 Visual Focus of Attention Estimation

2.5.1 Visual Focus of Attention and Gaze

If we define Visual Focus of Attention (VFOA) as what a person is looking at, then given that we know what and where objects of interest are in the scene, the problem for VFOA estimation becomes estimating the direction and depth of a person's visual focus (i.e., gaze).

For consistency, we define the 3 degrees of freedom (DOFs) of head pose as pan, tilt, and roll (see Figure 2[FIGURE]), with reference direction for pan and tilt being frontal direction of the head facing the camera. We define 2 DOFs of eye pose as pan and tilt similar to head pose, with reference direction for pan and tilt same as head pose.

Then, gaze direction is given by the accumulated rotation of average eye pose on head pose (Mora & Odobez, 2013), gaze depth is given by the intersection of the left and right eye pose directions. Therefore, the problem can be divided into head pose estimation and eye pose estimation.

2.5.2 Head Pose Estimation

There has been extensive research for head pose estimation for a single image frame, or head tracking for a video stream of images. Many methods have been proposed. As reviewed by Murphy-Chutorian et al., the following categories of methods have been identified: appearance templates, detector arrays, nonlinear regression methods, manifold embedding methods, geometric methods, and flexible models (Murphy-Chutorian & Trivedi, 2009).

Appearance Templates and Detector Arrays Methods

Appearance templates methods and detector arrays methods are mainly for estimating discrete / coarse head pose. For our task of identifying the object of interest under visual focus of attention, we do not

know a priori where and how close to each other the objects are. Therefore, only continuous fine pose estimation methods are of interest and are reviewed in this thesis.

Nonlinear Regression Methods

Using a machine learning approach, a direct mapping from the cropped head image to pose can be learned. Because such mapping is highly nonlinear, nonlinear regression methods, such as Support Vector Regressors (SVR) (Li, Gong, & Liddell, 2000) and Neural Networks (Multilayer Perceptron (MLP) (Voit, Nickel, & Stiefelhagen, 2008), and Locally Linear Map (LLM) (Krger & Sommer, 2002)) have been applied in this context as supervised learning, using head images as inputs and head poses as ground truths.

> **report the head pose accuracy and operating range of the best neural network?**

Although neural networks are among the most popular and accurate methods in head pose estimation, they are prone to errors from poor head localizations. And since, in our application, we cannot restrict user to remain in the center of the camera's field of view at all times, we need to either seek out a separate localization method for cropping the head, or seek a head tracker that doesn't have this problem. Using face detectors for localization is one solution, but it's operating range is **BLAH**, bottlenecking the head tracker's operating range [REF].

Manifold Embedding Methods

Manifold embedding methods learn a projection from high dimensional image space to some low dimensional space. The methods are unsupervised learning methods that only require head images and not pose ground truth labels. Promising techniques include: Isometric feature mapping (Isomap) (Raytchev, Yoda, & Sakaue, 2004), Locally Linear Embedding (LLE) (Roweis & Saul, 2000), and Laplacian Eigenmaps (LE) (Belkin & Niyogi, 2003).

However, due to the fact that they ignore pose labels, the projected low dimensional space may not be capturing appearance variations due to head pose alone, but may also include appearance variations due to identity, lighting, etc., making pose prediction inaccurate. Although there are work around techniques in the manifold embedding methods to tackle the appearance variations due to identity, more systematic ways to deal with it are seen in the following two methods: geometric methods and flexible model methods.

Geometric Methods

Geometric methods use person independent facial features (e.g. corners of eyes and mouth, tip of nose, etc.) to predict head pose. The relative locations of these features are exploited with geometrical assumptions of a person's face such as parallelism, symmetry and proportion (Wang & Sung, 2007).

A major caveat of these methods is that their accuracy relies on accuracy of features tracking. For our scenario of a child washing hands at the sink, where a moderate resolution camera captures a mid-range field of view, features are not guaranteed to be tracked with ease. Thus, it is better to select head pose estimation methods that don't require local features tracking.

Flexible Model Methods

Flexible model methods explicitly models the identity as well as pose of a person's face. Given a representation of the face, flexible models for shape and texture can be constructed using PCA from a face database to represent the directions in which the face most likely (naturally) deforms. Using this model, a new face image can be fitted through optimizing both an identity parameter and a pose parameter. This way, pose can be extracted independent of identity.

There are two popular and related flexible model methods: Active Appearance Model (AAM) and 3D Morphable Model (3DMM). They differ in the following ways:

Face Representation   Although both models use triangulated mesh representations of the face, AAM uses a 2D representation while 3DMM uses a 3D one. Also, AAM uses a sparse mesh with vertices at local features while 3DMM uses a dense mesh with vertices at the pixel level. These make AAM more computationally efficient, but on the other hand 3DMM more robust to low resolution image, partial occlusions, lighting variations, and large head rotations.

Face Fitting   Because AAM operates in 2D, fitting simultaneously the pose as well as the identity parameters requires us to recover a 3D model of the face using a structure-from-motion algorithm and then use the 3D model's weak perspective projection to constrain 2D fitting (Xiao, Baker, Matthews, & Kanade, 2004). 3DMM fitting is less convoluted if using a RGB-D camera (e.g. the Kinect camera). The pose and identity can be fit simultaneously using a non-rigid iterative closest point algorithm (Optimal Step Non-rigid ICP) (Paysan, Knothe, Amberg, Romdhani, & Vetter, 2009).

Kinect Fusion

Very similar to 3DMM, the Kinect Fusion algorithm developed by Microsoft also can be used to build a 3D head model, and track it's orientation through ICP [REF]. The main difference between the two algorithms lies in that 3DMM uses the optimal step non-rigid ICP, which uses a parameterized model of a person's face generated by doing PCA of a face model database. On the other hand, the head model generation from Kinect Fusion simply rely on building a point cloud of the head, thus its model is not parameterized. Besides this head model generation step that's different, everything else used in head pose tracking are the same – rigid ICP alignment. However, since 3DMM focuses on modeling the face while Kinect Fusion is capable of modeling the whole head, Kinect Fusion may perform better in tracking extreme head poses where the face is largely obstructed. 3DMM, on the other hand, can be tolerant of dynamic expressions on a person's face [REF], while Kinect Fusion may yield poorer tracking when the head model deforms.

2.5.3   Eye Pose Estimation

Eye pose estimation for a single image frame, or gaze tracking for video stream of images, have also been extensively researched. Here we present results from a recent review by Hansen et al. (Hansen & Ji, 2010). Eye pose estimation methods can be categorized as shape-based, feature-based, and appearance-based:

Shape-Based Methods

Shape-based methods are seeking to contour fit the shapes of iris, pupil, eye, etc. Some examples include: a simple model of ellipse fitting the shape of iris or pupil region, proposed by Valenti et al. (Valenti & Gevers, 2008); a complex model of deformable template fitting the eye and pupil shape, proposed by Colombo et al. (Colombo & Del Bimbo, 1999).

Feature-Based Shape Methods

Feature-based shape methods seek eye structure localization through identifying a set of distinctive local features. One can use features directly in the intensity image, with features found for example in limbus, pupil, cornea reflections, eye corners, etc. An example for this is the neural network method by Reinders et al. (Reinders, Koch, & Gerbrands, 1996). One can also use features in a filter response of the image, for example seen in the method of A Sirohey et al. (A Sirohey & Rosenfeld, 2001).

Both the shape-based and feature-based methods are inaccurate for moderate resolution mid-range field of view images – they often require a close up view of an eye. The problem with large field of view containing not only the eye regions is two fold. First, lower resolution for the eye regions are available, making the above methods less accurate. Second, without a proper localization of the eyes, false positives arise. Methods combining head pose and eye pose estimation is an effective way to combat the false positives in a large field of view as well as dealing with large head poses. It improves accuracy through the two estimators constraining each other in localizations (Valenti, Sebe, & Gevers, 2012). However, a better way, as seen in the head pose tracking section of the literature review, is to use appearance-based methods.

Appearance-Based Methods

Another approach to deal with moderate resolution mid-field images is through appearance-based methods. One can directly model the mapping from eye appearance to eye pose, seen by the neural network method of Baluja et al. (Baluja & Pomerleau, 1994). However, direct modeling requires large number of training images, the collection of which is a tedious process. One way to reduce number of training images needed is through template matching with local linear interpolation. An example of this is the Adaptive Linear Regression (ALR) method of Lu et al. (Lu, Sugano, Okabe, & Sato, 2011).

The major caveat with these appearance-based methods, as similarly to head pose estimation, is the inability to separate appearance variations in identity from pose. One way to tackle this is through keeping a bank of personal models, seen in Mora et al. (Mora & Odobez, 2013). A new person's eye images can then be represented by a linear combination of exemplar eye images from people in the bank using ALR. Then, to ease computational cost of ALR, we only keep the top few models that have their exemplar eye images used most often, decreasing the search space during ALR's optimization step.

2.5.4 Using RGB-D Camera for Gaze Estimation

The works of Mora et al. are good fits to our application (Funes Mora & Odobez, 2012; Mora & Odobez, 2013). Specifically, their approach uses the 3DMM head pose estimation together with ALR eye pose estimation. One thing to note is that the commercial grade RGB-D camera, Kinect, is used. Although 3DMM works with 2D camera as well, getting a direct sensor input on the depth information is advantageous in this context because it avoids inferring the 3D structure from 2D images, reducing

inaccuracies and computations. Also, it enables 3DMM to estimate pose through shape alone, ignoring texture fitting altogether, reducing computations and increasing robustness to lighting.

2.5.5   Discussion

From the reviews, we see that Kinect Fusion and ALR are ideal for our application. They yield many advantages: Firstly, usage of appearance-based methods means we are moderate resolution mid-field scenario ready. Secondly, only a simple calibration step is needed to generate the head model for Kinect Fusion and a bank of people's exemplar eye images are needed for ALR. For the children with ASD population, having relatively simple calibration step that doesn't require the child to behave in a certain way is very useful. Next, once the person's head model is generated, pose can be fitted without fitting identity, reducing computations. Lastly, the head image can be restored to the frontal head pose and a consistent frontal eye region cropped out, making eye pose estimation robust to large head poses.

Chapter 3

Research Objectives

3.1   Overall Goal and Approach

Our overall goal is to increase children with ASD's engagement level during COACH prompting and task execution, and thus improving prompt compliance and task completion rate. Our approach is:

1. to incorporate a humanoid robot, NAO, into the current COACH setup, capable of delivering verbal and gesture prompts and attention grabbers.

2. to automatically track the VFOA of child for more effective attention grabbers.

3.2   Central Hypothesis

We hypothesize that the incorporation of an embodied agent, such as the humanoid NAO, is sufficient in better engaging the child and better capturing and maintaining attention during prompting and task execution, and ultimately yields higher prompt compliance and task completion rates when assisting children with ASD through ADLs such as hand washing.

3.3   Specific Objectives

Our objectives are:

1. To explore the different modes of interactions between NAO and the child when prompting the hand-washing activity using a Wizard of Oz setup, focusing on verbal, gestures, and gaze for the modes of interactions

2. To investigate the impact on child's engagement, compliance, and task performance to an embodied prompting agent, comparing the robot, NAO, against the child's parent

3. To implement a real-time algorithm for tracking the child's VFOA

Our hypotheses are:

1. Gestural, gaze, and verbal are the essential modes of interactions present in the hand-washing prompting scenario between children with ASD and the embodied agent NAO.

2. Child has higher percentage of time looking at the correct target (i.e. robot or monitor during prompting and sink object or hands during task execution) when assisted by NAO than by the parent.

3. Child exhibits higher prompt compliance and better task completion when prompted by NAO than by the parent.

4. Attention grabber prompts yield higher percentage of time looking at prompting agent and higher compliance rate from child for both NAO and the parent

5. Child grows more confident in hand-washing as the trial goes on, and this confidence is independent of who is prompting

6. Using 3DMM and ALR for estimating head pose and eye pose, and using the Kinect camera, a classification rate of more than 80% is achieved for estimating child's VFOA on NAO, monitor screen, soap, towel, tap region, hands, and idling.

Chapter 4

Wizard of Oz

One major objective of this thesis is to investigate the impacts that using a humanoid prompting agent has on the visual attention, prompt compliance, and task performance of children with ASD during hand-washing activities.

This is the first research of its kind in the field of humanoid robot prompting agent guiding children with ASD through an activity of daily living. Therefore, it is wise to begin with a pilot study, the purpose of which is to show plausibility of the key underlying assumptions of our hypotheses, and to probe what questions are important to be answered later in a more rigorous randomized control trial. For this reason, the pilot study should be exploratory in nature, having a flexible experiment design, and relatively low experiment setup cost.

4.1   Wizard of Oz Experiment Design

The Wizard of Oz (WoZ) is an experiment design widely used in Human Computer Interaction (HCI) and Human Robot Interaction (HRI) researches[REF]. In a typical WoZ study, there is an interactive agent that is not yet fully autonomous, and is remotely controlled by a human wizard (operator), and this fact is concealed from the user being tested until after the study. The wizard may control one or many parts of the agent, such as speech recognition and understanding, affect recognition, dialog management, utterance and gesture generation and so on [REF]. The advantage of a WoZ study is that it does not require a large amount of work spent in implementing the artificial intelligence (AI) behind the agent – it is taken care of by the wizard. This is great for testing hypotheses early on in the design loop, enabling us to obtain feedbacks from users, learn, and iterate through design cycles faster. Of course, care needs to be taken to ensure the mocked up part of the AI is implementable in the near future, since the real purpose of the mock up is to have an early knowledge of the real design constraints, not trying to provide a less constrained solution.

The characteristics of a WoZ study fits our pilot study requirements, where we want to learn early the important design questions regarding building an effective ADL prompting robotic agent for the children with ASD population. Therefore, we will conduct a WoZ study, in which a humanoid robot whose motions and speech are preprogrammed, but the decision and timing of their executions are controlled remotely by the researcher. This is mocking up computer vision algorithms that understands the child with ASD's actions, speech recognition algorithms that recognize the child with ASD's verbal

interactions, and the AI decision making algorithms that decides what prompts to deliver and when to deliver them.

During each WoZ study session, the child with ASD will be asked to complete the hand-washing activity in the washroom with the supervision of one of his/her parents, with the help of the NAO robot, or with the help of both the parent and the robot. The researcher, and the parent if the child is to be assisted only by the robot, will be in an adjacent room out of the child's view to observe his/her hand-washing activity. However, the parent may enter the washroom if the child needs physical assistance to complete a step. An interface running on a laptop, connected wireless to the robot, will be used by the researcher to control the robot, as well as to monitor the progress and responses of the child through the video feeds of the cameras installed in the washroom.

4.2   Recruitment

Participants will be recruited from a previous autism study who indicated that they would be interested in participating in future studies related to the development of the COACH prompting system.

Participants will be children between the ages of 4 to 15 with a diagnosis of ASD, and their parent. Six children will be recruited. This sample size is typical for studies of this nature for children with ASD. For example, a pilot study by Bimbrahw et al. [REF] and a Wizard of Oz study by Bhargava et al. [REF] both involved a similar sample size of the children with ASD in their studies. We chose six children for this pilot study in order to equally explore the two permutations of experimental conditions (i.e. A-A-B-B-C-C and A-A-C-C-B-B, see [REFsection]). Participant demographics will be recorded and will include age, sex, and the Social Responsiveness Scale (SRS) test results. The SRS is a commonly used tool to identify the presence and estimate the severity of ASD [REF]. The results of the SRS will allow the research team to substantiate a diagnosis of an ASD for the child participants before proceeding with the study.

The inclusion criteria for enrolling in the study are as follows:

- Boys and girls between the ages of 4-15

- Parent report of a clinical diagnosis of an ASD to be confirmed through administration of the Social Responsiveness Scale (SRS)

- Has difficulty independently completing self-care activities, specifically hand-washing

- Has the ability to follow simple, one-step verbal instructions

- Ethical consent granted by parents or primary guardian

- Does not exhibit severely aggressive behavior

Each participating family will be given a $200 honorarium per child subject upon completion of the study (please see Appendix K Study budget sheet). All participants will be able to withdraw from the study at any time. The honorarium will be adjusted to be proportionate to the number of visits completed (e.g. completing 3 visits means the participated child will receive $100 ($200 * 3 / 6 = $100)). This will be made clear to participants at the time of consent.

4.3   Humanoid Robot NAO

We chose the half-torso version of the commercially available humanoid robot NAO from Aldebaran Robotics as our robotic prompting agent. NAO is a humanoid robot about half a meter high in full torso [FIGURE of full torso NAO]. It is designed by Aldebaran Robotics to primarily serve in educations for children and in academia research in robotics. Because of this, NAO is designed to have a very likable appearance – one with baby like facial features. Also, NAO is equipped with the state of the art mechanical, electrical, embedded, control, and local network communication systems. It also has cameras and sonar sensors for computer vision algorithms for scene understanding, path planning, and obstacle avoidance. The software development kit (SDK) provided is very easy and powerful to program with. Also, an even easier graphic user interface (GUI) for robot behavior programming, the Choreographe software, is also available. One caveat of using NAO for SAR HRI researches is that it is only equipped with a single degree of freedom finger dexterity, though other joints in its body are much more mobile. It is more than enough for doing simple pointing and other non-contact gestural prompts in sync with verbal interactions. It just cannot perform detailed hand gesturing. This makes NAO less capable in demonstrating a hand-washing step in high detail.

From a HRI research perspective, Aldebaran Robotics took care of designing the intrinsics level of HRI, where NAO has a likable appearance and child like neutral gender voice, although it's incapable of facial expressions. The design decisions we face when using NAO for this thesis is on the behavior level of HRI. Design decisions such as voice intonation choice, verbal prompts, motion gestures and gaze, and eye blinks using LEDs in eye regions are made in this thesis. The objective of this thesis is then to ultimately find out if the lower two levels of design decisions made are able to cumulate to the child with ASD perceiving NAO as a role model / supervisor / assistant during hand-washing.

For our pilot study, we use the half-torso version of NAO because we do not require any mobility from NAO – it is fixed on the sink table top [FIGURE]. The relevant functionalities of NAO we will utilize for delivering prompts include:

- Verbal prompting through its bilateral loud speakers on the head and speech synthesis functionality

- Body gesturing through its moving head and arms (although its fingers are not capable of hand gesturing)

- Flashing LEDs on the eyes and ears

4.3.1   Verbal Prompts

We used the text-to-speech engine from NAO to synthesize the verbal prompts. The pitch of NAO's voice is changed to a lower one than default for the verbal prompts to give a more authoritative feeling. The reward verbal prompt remains the default pitch, though, to give an exciting praise. The verbal prompts are worded as short, three or four word phrases, such as "turn on the water" or "rinse your hand", and a pause is put between the action and the subject so that the prompts sound clearer and is easier to understand to children with ASD.

4.3.2   Gesture Prompts

There are several kinds of gesture prompts NAO needs to perform:

- **Attention grabber (AG)**: When prompting is needed but child is not looking at NAO, NAO waves at child to grab the childs attention.

- **Motion demonstrating prompt (MoDemo)**: NAO demonstrates to child the motion of interaction (e.g. turning tap, scrubbing, rinsing, etc.).

- **Object pointing prompt (ObjPt)**: NAO points to the physical object of interaction.

- **Reward (REW)**: After a task is successfully completed, NAO flashes LEDs and celebrates.

The gaze behaviour of NAO during gesture prompts is also important and is grouped as: looking at child (when delivering AG, MoDemo, REW), and looking at object (AR, ObjPt). The gesture and gaze motions can be programmed using NAO's software, Choregraphe.

4.3.3   Wizard of Oz Remote Control

The WoZ experiment setup involves controlling the robot remotely behind the scene by a human operator, the wizard. A touch screen laptop will be used as the user interface for the operator, and the behaviors of the robot are presented as buttons on the screen, with the camera views displayed along side. Keyboard accelerators are also implemented for faster access to robot actions.

4.4   Surveys

entrance survey, SRS, post intervention survey for parent, for child, exit survey

4.5   Protocol and Setup

4.5.1   Entrance Survey and SRS

Prior to their first HomeLab visit, the parent will be asked to complete the Social Responsiveness Scale (SRS) [REFsection]. If the child meets the SRS score (minimum of 76 T-score), the same parent will then be asked to complete the entrance survey before their first visit of the HomeLab. This is to capture the childs demographics, his/her hand-washing ability level and to gather information to help the research team configure the system to the childs preferences [REFsection]. The same parent who has completed the entrance survey should accompany the child through all the HomeLab visits.

4.5.2   Protocol Overview

Each child will visit the HomeLab on the 12th floor of Toronto Rehab Hospital once a week with a total of six visits with his/her parent. The six visits will be evenly divided into three phases. The three phases are the baseline phase (Phase A) and the intervention phases (Phase B and Phase C). In Phase A, the child will be asked to wash hands by him/herself as independently as possible. The parent will be instructed to provide assistance to the child only when necessary (as outlined below). In Phase B, the child will be assisted by both the robot NAO and the parent during hand-washing. In Phase C, the child will be assisted by NAO alone. During each of the intervention trials, the parent may decide when

to enter the washroom and provide assistance to the child, and will be providing assistance only for the specific step that the child is having difficulties with (as outlined below).

It will take about an hour to an hour and a half for each visit. The child will be asked to wash his/her hands eight times for every visit, for a total of forty-eight trials per child. The child and his/her parent may take short breaks after hand-washing session. The break may last as long as the child needs until he/she is willing to continue the trial. If the parent feels the need, they may leave and come back to finish the activities another day. They will not be withdrawn from the study unless requested.

4.5.3 Specific Protocol

The specific protocol for each hand-washing session is as follows:

The hand-washing activity will be broken down into seven tasks: turn on the water, wet your hands, squeeze out the soap, scrub your hands, rinse your hands, turn off the water, and dry your hands. These tasks are modified based on Bimbrahw et al. pilot study [REF]. These constitute the same tasks as Bimbrahws except that the first (i.e. turn on the water and wet your hands) and the last task (i.e. turn off the water and dry your hands) are now four individual tasks to ensure that each task only involves one action.

All phases will be video recorded by the overhead, the scene, and the Kinect cameras and will be audio recorded by the microphone from the scene camera. The researcher will be in the room adjacent to the washroom out of view of the child for all phases. The researcher will remotely control the robot and the virtual avatar in the intervention phases. The parent will be with the child in the washroom for the baseline phase, and will be either with the researcher or with the child for the Intervention Phases, depending on if the child needs physical assistance from the parent.

Phase A (Baseline Phase)   The first two visits will be the baseline phase and will include sixteen trials of hand washing with eight trials for each visit. The child will be asked to complete the hand-washing as independently as possible. During this phase, the parent will be present in the washroom while the child is completing the hand-washing tasks. The parent will verbally and/or physically assist and give reinforcement to the child whenever the parent feels necessary.

Phases B and C (Intervention Phases)   The rest of the four visits will be the intervention phases and will include thirty-two trials of hand washing with eight trials for each visit. The child will be asked to wash his/her hands with the help of NAO or of both NAO and the parent in the washroom. During each trial, NAO and the parent will wait for the child to start each task. If the child has trouble an appropriate prompt will be delivered from NAO in order to help the child complete the task. If the child does not respond to NAO's prompt, an attention grabber will be delivered to capture the childs attention from the prompting agent. The attention grabber may be repeated for the second time to the child if he/she fails to respond to it. A verbal reward will be delivered to the child once he/she completes the task.

The parent's role in phase B and C differ in that, in phase B, the parent takes a more active role to prompt the child of what to do. On the other hand, in phase C, the parent should take more of a back seat role, prompting only for the purpose of reminding the child to listen to the robot, but leaves the specific step to be prompted by the robot. Of course, if the child doesn't respond to any of the prompts, the parent will need to physically intervene and complete the task together, just like in phase A. After

the physical intervention, the parent will then instruct and encourage the child to continue the rest of the hand-washing tasks on his/her own.

There are three prompt categories that the NAO robot will deliver when interacting with the child (please see [TABLE] for the specifics of each prompt used):

1. **Task Prompt** (to prompt the child through a hand-washing task):

   A verbal prompt will be delivered, such as Please [task name] (e.g. Please turn on the water.). Synchronous to the verbal, a visual prompt will also be delivered. This is a two-part gesture prompt of: first, demonstrating the motion of interaction while looking at the child; second, pointing to the sink object (e.g. the tap) while looking at the object. A maximum of two prompts will be given to the child. If the child does not respond to the second prompt or has started the task but does not complete the task within the task execution timeout, the parent will be asked to help the child complete the task.

2. **Attention Grabber** (to catch the childs attention to the NAO robot or the avatar):

   A verbal prompt will be delivered, such as Hi, [childs name]! Synchronous to the verbal, a visual prompt will also be delivered. This is an attention grabbing gesture of waving and looking at the child. A maximum of two attention grabbers will be given to the child in order to get his/her attention to look at the robot/avatar. The parent will be asked to instruct the child to look at the robot/avatar if he/she does not respond to the second attention grabber.

3. **Reward** (to provide positive reinforcement when the child attempts a task without the help from his/her parent):

   A verbal reward (i.e. Great!) will be delivered while looking at the child and switching back and forth the colors of the light-emitting diodes (LEDs) on the eyes after successfully performing a task.

For each trial, in addition to the three prompt categories stated above, the NAO robot and the virtual avatar will also deliver a short introduction before the start of each trial, a re-intro after the parent assisting the child through a task, and an outro at the end of each trial. The introduction is a two-part prompt. The first part is an attention grabber. The second part consists of a verbal prompt (i.e. Lets start washing hands.) with a simple conversational gesture. The re-intro is a verbal prompt (i.e. Lets continue washing hands.) with a simple conversational gesture. Same as the introduction, the outro is a two-part prompt. The first part consists of a verbal prompt (i.e. Good job, [childs name]!) with a gesture of fist pumping in the air. The second part consists of a verbal prompt (i.e. You are all done.) with a gesture signifying all the hand-washing tasks have been done.

4.5.4 Post-intervention Survey and Exit Survey

During the last visit, the same parent who has completed the entrance survey will be asked to fill out the post-intervention survey and the exit survey [REFsection], which will allow him/her to provide the research team with his/her feedback regarding the device. A variation of the post-intervention survey will be verbally administered by the researcher to the child participant to capture his or her views of the system [REFsection]. This information will be used by the research team to better understand which aspects of the system are effective, which are not, and how, if in any way, the system should be changed.

4.5.5   Data Collection

Three kinds of video data will be collected from the three corresponding cameras overhead, scene, and Kinect. The overhead and scene video data will be reviewed and annotated by two researchers. The inter-rater reliability will be calculated using Cohens Kappa [14]. The overhead video data will be used to score the participants prompt compliance and hand-washing performance. The scene video data will be used to evaluate the participants engagement during the whole activity. The effect of embodiment on engagement, compliance, and performance will then be explored qualitatively and quantitatively.

The Kinect video data will not be annotated. Instead, it will be used to evaluate the automatic gaze estimation algorithm that we developed. Specifically, the Kinect video data will be used by the gaze estimation algorithm as input and the output predictions will be compared with annotations of the scene video data to derive the algorithms prediction accuracy.

4.5.6   Ethics

The WoZ is approved by the Research Ethics Board (REB) of University Health Network (UHN), belonging to which is the Toronto Rehab Hospital, where the study is conducted.

Consent and Assent   Participants will be given a package of consent/assent forms prior to starting the study [REFsection]. One of the parents will need to provide their consent for their child and themselves to participate in the study. In addition, child participants will need to provide their assent to participate in their every visit of the study.

Interested families will receive an information/consent package (please see Appendices F to H) prior to starting of the study. This package includes consent/assent forms for participation in the study for the parent and child with ASD (these forms include study details and research contact information) as well as consent to be videotaped for the parent and child with ASD. Consent from the parent and assent from the child with ASD will be given if and when they feel comfortable that they understand the information presented. Potential participants of both parents and children will have up to a week to decide if they would like to participate, although they may consent to participate as soon as they feel comfortable doing so. Parents will need to provide their consent for their children (please see Appendix F) and themselves [REFsection] to participate in the study. In addition, child participants will need to provide their assent in their every visit of the HomeLab during the study (please see Appendix H) to participate. Parents will be required to consent to having their children and themselves videotaped during the study. The parents will be informed that they and their children may withdraw from the study at any time without penalty.

Confidentiality   Each participating family (parent and child with ASD pair) will be assigned a code number when they sign the consent/assent. All data in the study will be labelled with these code numbers only - the names of the participants will appear only on the information and consent/assent forms and will be kept confidential. Consent forms will be placed in a secure and locked area in the PIs laboratory, with access exclusively restricted to the research team. All forms will be destroyed seven years after the study publication.

The information and data collected will remain strictly confidential and will not affect any of the participants (both the parent and the child) employment, care, or treatment in any way. A code number

will be assigned to each parent and child participant when they give consent. This code number, instead of their name, will be used for all data collection and analysis. Direct quotes may be included in the final research paper but names will not be used in any report or publication. Privacy of participants (both the parents and the children) will be ensured by omitting all participant information from participant data, by employing data encryption, and by storing data on a secure server. If and only if participants consent, participants (both the parents and the children) video data may be presented for educational purposes. If any images or videos are used in presentations and publications, faces and other identifiable features will be masked.

Both the video and audio data will be stored temporarily on the touchscreen laptops hard drive during each childs visit. The data will be encrypted and transferred to the TRI servers as soon as after each childs visit. The portable devices, such as USB sticks, will be used to transfer the data to the TRI servers. All files stored in the portable devices will be password protected and encrypted. The data on the laptops hard drive and the portable devices will then be purged immediately after transfer.

Data Storage   All soft (electronic) data will be encrypted before any transfer is made. All data will be password protected and be stored on the TRI servers with access restricted to the research team. The laptop used for the study will be password protected so that only the research team has the access to it. All computerized data will be password protected. All survey data will be stored in a locked cabinet different from where the consent forms are stored. Access to all the data will be restricted only to the supervisor and researchers involved in the project.

After the study is completed and the results of the study are published, data will be stored for at least seven years from study closure. All data will be destroyed seven years after the study closure. Data contained on paper material will be destroyed by shredding the material. Data contained on electronic media will be destroyed by erasing or other removing the data in such a way that it cannot be retrieved.

4.6   Video Annotations

4.6.1   Annotation Framework

Only the scene camera videos will be annotated, since this view alone suffices in telling us the progress and response of the child in hand-washing steps and to prompts.

Each video file usually contains one hand-washing session, sometimes two. The annotator needs to scroll through each video until the scene of the child entering the washroom, marking it as the start of a session. The child leaving the washroom marks the end of a session.

A session contains many hand-washing steps. For each step, the annotator describes the video using a 3-segment scheme. The first segment describes the child's actions before any prompts, the second describes the prompting agent's prompts, and the third describes the child's actions after the prompts. We call this 3-segment scheme a "prompt section". The items to be annotated in each segment of the prompt section can be seen in TABLE BLAH.

A hand-washing step could have multiple prompt sections. Take for example the following scenario: The child executes the wrong step, so the parent prompts the correct step, but the child ignores the parent and continues the wrong step. This constitutes one prompt section. Then the parent prompts again, and the child finally follows the prompt and executes the correct step. This constitutes then another prompt section. Note that for this example, because the parent prompts a second time without

waiting for the child to stop his/her current action, the child can take no actions before prompt in the second prompt section. Thus, in annotation, the second prompt section should have a blank 1st segment.

A hand-washing step could also have multiple prompt sections because of the step's nature. For the "extended steps" (i.e. scrubbing, wetting, rinsing, and drying), even when the child is executing the correct step, the prompting agent may deliver more prompts to encourage the child to keep doing the same step for an extended period of time. This is in contrast to the non-extended steps (e.g. turning on the water), where a single action from the child marks the completion of that step. An example of an extended step with multiple prompt sections is: The child starts rinsing. The parent tells the child to keep rinsing. The child continues to rinse, and the parent says again "keep rinsing". The child rinses more and then decides to stop, so the parent prompts again to rinse more. The child follows. After a while, the parent decides this is enough and prompts for the next step. There are two prompt sections in this scenario. The first section consists of the child doing the correct step, the parent prompts, and the child follows prompt and stops on his/her own before next prompt. The second section consists of the child being idle before prompt, the parent prompts, and the child follows prompt. One thing to note for this example is, although the parent prompted twice before the child stopped rinsing, both of those prompts should be merged and annotated as a single prompt section. In fact, any consecutive prompts resulting in the same actions from the child should be merged together in one prompt section. This way, "task completion rate" would not be inflated by extended steps where many prompts were given while the child continued the same step. On the other hand, a separate measure, "number of prompts until child stops step", is used to reflect the prolonging of an extended step due to prompts. Also note that merging multiple prompts that have same child reactions into one prompt section applies to prompts that the child is ignoring as well.

Lastly, getting soap is a step similar to the extended steps such that it itself takes an extended period of time to execute. However, it is different because of the child's tendency to always get more soap than needed. So after the child starts getting soap, any prompts given are to tell the child to stop the step (as opposed to prolonging the step, as in the other extended steps' cases). This means the measure, "child stops step before next prompt", is marked true if and only if no prompts are given after the child starts getting soap. In general, "child stops step before next prompt" is marked true if the child stops the step before any prompt is given that is either telling him/her to stop (e.g. a verbal reward) or to go on to the next step.

4.6.2   Annotation Tools

The videos are played back by the software Media Player Classic - Home Cinema (MPC-HC 64-bit v1.7.8), where timestamps of millisecond resolution can be obtained. The annotations are recorded onto Microsoft Office Excel spreadsheets, and each sheet exported to CSV files to be analyzed.

4.6.3   Annotators and Inter-rater Agreement

- number of annotators - percentage of overlap - inter-rater agreement calculation (method, what's good enough)

4.7 Measures

4.7.1 Video Data Measures

To measure the child with ASD's engagement level, visual focus of attention, prompt compliance, task completion, and self confidence, and to prove or disprove our hypotheses regarding them, the following metrics are calculated from the annotated video data:

General Engagement Level

- % of prompts that child **smiles**

- % of prompts that child **murmurs**

- % of prompts that child is **distracted**

Visual Attention Correctness

- % of prompts that child **looks at prompting agent or prompted object** when prompted

- % of attempted steps that child **looks at step attempted** during execution

Prompt Compliance

- % of prompts that child **complies**

- avg # of prompts till child **complies**

- avg time till child **complies**

Task Completion

- % of attempted steps that child **successfully completes**

- avg # of prompts till child **stops a step before prompted** to stop (out of all prompt sections that child stops a step before prompted to)

- avg time for **extended steps**

- avg time for **soap steps**

- avg # of presses for **soap step**

- avg # of steps that child **requires parent's physical intervention**

Self Confidence

- avg # of prompts that child **starts the step before prompt**

- avg # of prompts that child **stops the step before prompted** to stop (out of all prompt sections that child stops a step before prompted to)

4.7.2   Scope of Measures

The scope of the measures used above are meant as follows:

- **% of prompts**: percentage of prompt sections out of all the prompt sections in a session

- **% of attempted steps**: percentage of prompt sections out of all the prompt sections that child actually attempts a step after prompt in a session

- **avg # of prompts**: average number of prompt sections in each step out of all the steps in a session

- **avg time**: average time durations in seconds in each step out of all the steps in a session

- **% of attempted steps**: percentage of steps that child attempts out of all the steps in a session

- **avg # of presses**: average number of soap presses in each soap step out of all the soap steps in a session

4.8   Data Analysis and Results

4.8.1   Participants Recruited

Due to limitation of time, we were only able to recruit one subject. - report subject demographics: age, gender, ethnicity, hand-washing ability, other inclusion criteria fit, SRS score. General impression (verbal?, communicative?, interactions with parents, behavior trend)

4.8.2   Experiment Design Change

Because we only had one subject, we planned to carry out the baseline phase A (only parent prompts) first, then the intervention phase C (only robot prompts), and lastly the intervention phase B (both parent and robot prompts). This is because we wished to see how well could the child wash hands with help from robot alone first, giving us an initial idea of whether a joint prompting phase (phase B) was even needed. However, it turned out that the results were far below expectation for phase C, so continuing the second part of phase C before starting phase B will give a weaker result in the end. Also, since we only had a single subject, a case study format with a more exploratory style for experiment design is more appropriate here. So, instead of a strict experiment design to compare the effects of phase B vs. phase C and phase B vs. phase A, we employed a looser mix of interventions, blending phase B and C as a single continuum of how much the parent intervenes. The freedom of choosing how much to intervene is left to the parent as he/she saw fit, keeping in mind that our ultimate goal is getting the child to be assisted by the robot alone by the end of the intervention phases. Another thing that happened was, after conducting the first half of phase C, we found that action controls needed to be granted to the operator at a lower level, so that the robot behavior can be adapted to the child's actions more easily and quickly. This was mainly due to our inexperience with working with children with ASD, being overconfident in the child's compliance to the robot prompts and underestimating how fast the reaction time the child had. After this change in control scheme, the robot was able to keep up with the child and its prompts started becoming useful. Because of the above reasons, we ended with the following general order of phases: A-A-C-B-B-C. The last three trials had more parent involvement (i.e.

more on the phase B side), but the parent employed prompt fading, and gradually decreased his/her involvement in the prompts, resulting in the last trial to be more on the phase C side.

- show figure of parent's involvement vs. time - show table of parent involvement (categorized into classes) across sessions, grouped in trials

Despite the experiment design change, the hypotheses raised were still valid for testing under the new design. The specific protocols for each phase still remained the same, and the experiment setup and data collection procedures remained unchanged.

4.8.3   Video and Survey Data Collected

There are 6 trials conducted as planned. However, not all trials had 8 sessions, [TABLE] shows the number of sessions for each trial. We had only 7 sessions for trial 3, but more than 8 sessions for trial 4, 5, and 6.

All three surveys and the SRS report were administered as planned.

4.8.4   Video Data Analysis

- method - result

4.8.5   Annotation Inter-rater Agreement Analysis

- method - result

4.8.6   Survey Data Analysis

- method - result

4.9   Discussion

Chapter 5

Visual Focus of Attention Estimation

To track the VFOA of the child, we are using the Microsoft Kinect to collect RGB and depth images of the childs head. Gaze directions can be tracked more accurately when head is viewed near frontal. Since we cared more about the childs attention to prompting agents than to sink objects, we setup the Kinect near the robot, and have sink objects further from the Kinect (i.e. soap and towel) be spread apart from each other to be easily distinguished.

The problem of estimating the object under childs attentional focus is broken into three parts: head pose estimation, eye pose estimation, object identification.

5.1 Head Pose Estimation

5.1.1 2D Video Camera Approach

5.1.2 3D Kinect Camera Approach

Approach Overview

A more accurate way for gaze tracking is to use the Kinect depth sensing camera so that we track the head pose using 3D data. The idea is to first build a 3D model of the specific user's head using several frames of the depth images. Then we can fit the model through rigid transformation onto new frames of depth video stream to estimate the head pose in real-time. After the head pose is obtained for a depth frame, we transform the corresponding frame of color image to the frontal head pose, and crop out the eye regions for gaze prediction. This method is more accurate than the previous 2D method for the following reasons. First, the color image is transformed into a frontal pose, and this transformation introduces less distortion to the image compared to the stretch based transformation used previously. Second, when cropping the eye regions from the color image, we can specify the eye cropping regions ahead of time. Since all color images are now in frontal pose, the cropping regions remain the same even when the user is rotating his/her head. This gives a more accurate and more stable cropping compared to the corner detection based cropping used previously. Of course, the two advantages for this method are both dependent on that the head pose tracking is accurate.

KinFu Head Modeling

We would like to choose a method for building a user specific 3D model of the head that requires minimal user interactions. This is because we are focusing on the children with ASD population. Asking a child with ASD to sit still and keep head straight for a period of time for a 3D laser scanner to scan his/her head is less feasible. Instead, we aimed to obtain the head model through a short video stream of the Kinect's depth frames captured as the child came into the washroom and stood in front of the sink, without trying to restrict the child's movements.

To achieve this, two pieces of softwares are essential. One is Point Cloud Library (PCL)'s KinFu, which enables us to integrate the frames from Kinect's depth video stream into a single 3D model of the scene. The second software is Microsoft Kinect for Windows (K4W) SDK's Skeleton Tracking, which enables us to track where the user's head is, and so that only the head is reconstructed by KinFu.

How KinFu Works   PCL is an open source stand alone library that processes 3D data in the form of point clouds (collections of 3D points). It provides functionalities such as filtering, normal estimation, feature extraction, transformations, segmentations, surface reconstructions, etc [REF]. KinFu is PCL's open source implementation of Kinect Fusion [REF], an algorithm first proposed by Microsoft and demonstrated in its KinectFusion API in K4W SDK [REF]. The idea of Kinect Fusion is related to the traditional Simultaneous Localization And Mapping (SLAM) algorithm [REF], where feature points in the scene are matched across frames of a 2D video stream, so that the camera's orientation within the scene is tracked across frames. At the same time, the frames are patched together to build a sparse 3D map of the scene. Kinect Fusion extends SLAM into a dense version where every point in the point cloud now becomes a feature point and the full 3D scene is reconstructed. It also makes usage of the General Purposed GPU (GPGPU)'s parallel computation to speed up the algorithm to real-time. The PCL's KinFu and Microsoft's KinectFusion implementations have similar processing pipeline – they only differ in some specific low level algorithms used. Below is KinFu's processing pipeline [REF]:

Preprocessing   First, the depth frame from the Kinect camera is filtered through bilateral filtering to remove noise – it selectively smooths the surfaces while preserving edges [REF] [FIGURE]. Then, multiple resolutions of the depth frame is generated through sub-sampling, we call them the multi-resolution pyramid [FIGURE]. Lastly, each layer of the pyramid generates a 3D point cloud through back projection using the camera's calibration matrix. In each point cloud, the normal for each point is estimated by an eigenvector of Principal Component Analysis (PCA) of its neighborhood points [REF].

Alignment   The preprocessed point clouds now need to be aligned to the current scene model. If this is the very first depth frame, then its point cloud is used as the current model – alignment starts at the second frame. Alignment is done through the Iterative Closest Point (ICP) algorithm (with some modification of its procedures), starting from the point cloud of the coarsest layer in the pyramid. ICP is performed in loops until one of the exit criteria is met, after which the same is done for the point cloud of the next level in the pyramid, and the next, until all levels are traversed.

The ICP loop (the modified version) has the following procedures: Both points from the new frame's point cloud and the model's point cloud are projected to the model point cloud's camera image frame, and any pair of points from the two clouds falling onto the same pixel is noted as a match. Then the rigid transform that globally minimizes the sum of squared errors between matched points is calculated,

the error being the distance between the point of new cloud to the tangent plane of point of the model cloud. The exit criteria for the ICP loop are either 1) the maximum number of iterations are reached, 2) the changes of the transformation matrix falls below threshold, or 3) the sum of squared errors fall below threshold.

Surface Reconstruction   Finally, the aligned new frame needs to be integrated into the model and to form a new model point cloud so the pipeline can loop from the top as frames arrive. A new model point cloud is formed by first converting the two clouds into Truncated Signed Distance Functions (TSDF). A TSDF is basically a representation of surfaces of objects in a scene, with negative values assigned to voxels inside the surface or voxels that are not measured yet, positive values assigned to voxels outside the surface, increasing in value as we move further away from the surface, and voxels on the surface of objects are assigned the value of zero [REF] [FIGURE]. Then, volume integration of the two clouds are done through a weighted running average of the model cloud's TSDF with the new frame cloud's TSDF. Here the raw value of the new frame with no filtering is used for calculating its TSDF to avoid losing details. Lastly, surface reconstruction is done through the marching cube algorithm, which converts the new model's TSDF into a point cloud.

Using KinFu with Kinect2 Camera   To use KinFu with Kinect2 Camera, the open source PCL Kinect KinFu SDK is used [REF]. It acts like a driver for Kinect2 using the PCL point cloud framework. This SDK currently only supports generating 3D point cloud from the depth frames of Kinect2. We had to implement the generation of color point cloud using the color frames ourselves. However, the advantage of using this over using Microsoft Kinect2 for Windows SDK is that: first, Kinect Fusion was only in an unstable beta version in the Windows SDK during the time this thesis was conducted; second, the open source nature of PCL's KinFu enables us to have much more control in using the algorithm for our application. The modifications to the KinFu algorithm are outlined in the sections following.

Using KinFu for Head Modeling   Using KinFu for head modeling is very similar to the original purpose of KinFu – scene modeling. The difference lies that, for object (e.g. a head) modeling, we need to filter everything except the object of interest in the depth frame, so that KinFu only builds the 3D model over data on the object, and ignores everything. One thing that's neat about object modeling is, instead of rotating the Kinect camera around the object, we can rotate the object while fixing the camera. Also, specifically for head modeling, we can utilize Kinect's Skeleton Tracking algorithm to filter out everything except the head.

Not all objects can be tracked using the ICP algorithm in KinFu. In KinFu's paper, the author reports that whenever the object is moving too quickly or when not enough 3D features (e.g. edges and corners) are present on the object surface, ICP often fails [REF]. This is mainly due to the assumption in ICP that between frame movements are small (note that this problem may be potentially solved by higher frame rate camera with higher computational power so no frames are skipped). However, for head tracking, KinFu turns out to work just fine as long as the person isn't moving his/her head too fast [FIGURE]. Also, facial expressions deformat the face, making it deviate from the learned head model, and tracking accuracy may suffer due to the ICP's rigid nature. Thus we see that the robustness of ICP head tracking depends on two factors:

- the head's movement speed (both translational and rotational) relative to the ICP loop process

speed or the Kinect camera's frame rate (which ever is the bottleneck)

- the dynamics of facial expressions on the head

A validation of KinFu's ability for head modeling can be done by finding the maximum movement speed of the head (both translational and rotational) under expressionless vs. moving jaw faces conditions before KinFu's alignment step fails.

KinFu Head Pose Tracking

After obtaining the user's 3D head model, the 3D position and pose (pitch, yaw, and roll) of the head can be tracked in the depth video stream using KinFu. Skeleton Tracking is again used to filter out everything except the head in the depth video stream. In addition, it is used to give initial position of head for the ICP loop, so that new frames to be aligned are within the capability of ICP. Also, the KinFu algorithm needs to be modified to skip the surface reconstruction step, since we already have a model of the head.

A similar validation test as the one for KinFu head modeling can be done to see KinFu's robustness in head tracking.

Point Cloud Based Inverse Pose Transformation

Having accurate head model and head pose tracking enable us to restore the color image frames of the user's head into frontal pose. This inverse pose transformation is done by first forming a colored point cloud from the color frame, then 3D rigid transforming the point cloud inversely to the head pose so that the head represented by the point cloud is in frontal pose, finally projecting the transformed point cloud onto the camera image plane to obtain the 2D color image of the frontal posed head.

Colored Point Cloud   To form a colored point cloud, we need to calculate the 3D coordinates of every pixel in the color frame.

The function "MapColorFrameToCameraSpace" provided by K4W SDK is for this purpose, and is used by the PCL Kinect KinFu SDK. However, MapColorFrameToCameraSpace provides the 3D coordinates of color pixels by associating each pixel in the color frame with a pixel in the depth frame, and then calculating the 3D coordinates of every pixel in the depth frame. This approach is convenient to code and fast in execution, but is limited by the depth frame resolution. For the typical resolution of the Kinect2 camera, the color frame resolution is 1920 X 1080 (2073600 pixels), the depth frame resolution is 512 X 424 (217088 pixels). Using the above method, 2073600 pixels are available in color frame, but can only form 217088 unique points in the point cloud. This is an order of magnitude reduction, wasting the HD color frame provided by the Kinect2 camera. Even for Kinect1 camera, with color frame resolution BLAH and depth resolution BLAH, this is a BLAH reduction in resolution. This problem of using the depth frame resolution for point cloud greatly reduces the resolution of the resulting frontal pose color image, as seen in figure BLAH, making the next step, gaze prediction, harder.

To avoid resolution reduction, we use the 3D head model instead of the depth frame for calculating the coordinates of each color pixel:

Head Model Mesh Projection   To do this, we first generate a triangular mesh version of the 3D head model. Then we transform the head model mesh to match the pose in the current depth frame

(given by the previous step). Next, we project the mesh onto the color camera image plane, keeping track of which pixel each vertex of the mesh lands on. This enables us to mark which mesh surface each pixel in the projected image belongs to. More specifically, for each mesh surface in the model, we do a depth first search traverse on the projected image starting with the pixel that one of the surface vertices projects to. During traverse, we go to the pixel's neighbors one by one (there are eight adjacent neighbors to each pixel) if the pixel itself lies within the projected surface (i.e. inside the triangle formed by the surface's three projected vertices), and stop traversing if the pixel is outside of the projected surface, outside of the image boundary , or is already visited by the recursion. There are times where two surfaces overlap in their projections – this happens when one surface obstructs the view of the other (from the camera's point of view). We handle this by assigning pixels to belong to the nearest mesh surface to the camera's focal point. The result of this head model mesh projection is shown in [FIGURE].

3D Coordinate Calculation   After assigning surfaces to every pixel, the pixels' 3D coordinates can be calculated by linearly interpolating from the surface vertices. To do this, we take advantage of the fact that Barycentric coordinates are preserved during projection of a planar object in 3D onto another plane. Thus, a point on the 3D mesh surface preserves its Barycentric coordinate after projecting onto the color image plane. Note that expressing a point in Barycentric coordinates w.r.t. a triangle it is on is basically expressing the point as a linear combination of two of the triangle's edges. Mathematically, this means the following: Given triangular mesh surface vertices with coordinates P1, P2 and P3 in 3D, projected vertices with coordinates p1, p2 and p3 in 2D, and a point on the mesh surface with coordinate P in 3D, and projected coordinate p in 2D. The Barycentric coorindate for P w.r.t. P1, P2, P3 is $(\lambda1, \lambda2, \lambda3)$, with $P = \lambda1 \times P1 + \lambda2 \times P2 + \lambda3 \times P3$, $\lambda1 + \lambda2 + \lambda3 = 1$, and $\lambda1, \lambda2, \lambda3 > 0$. Then, the Barycentric coorindate for the projected point, p, w.r.t. p1, p2, p3 is also $(\lambda1, \lambda2, \lambda3)$, with $p = \lambda1 \times p1 + \lambda2 \times p2 + \lambda3 \times p3$. Using this fact, we can calculate the Barycentric coordinate for each pixel w.r.t. the mesh surface it belongs to, and then calculate the pixel's back projected 3D coordinate using the surface's vertices' 3D coordinates. An example of a colored point cloud formed using this method is shown in [FIGURE], note the comparison of this much denser cloud with a cloud formed using the previous depth frame based method.

Inverse Pose Transformation   With the color point cloud created, we are ready to form a frontal pose color image. First, we rigid transform the cloud so that the center of the head is at coordinate (0, 0, 2 × focal length) and its pose facing the origin. The reason for 2 × focal length is that the image plane is at 1 × focal length, and we want the cloud to be a little distance away from the image plane so that the projection looks good inside the image boundaries. Note that focal length is a programmer defined value used in the next step for perspective projection, and is the distance from image plane to the camera's focal point.

Projection onto Camera Image Plane

The last step before obtaining the frontal pose color image is the perspective projection. For this, we go through every point in the color point cloud and calculate each point's 2D image coordinate. If two points land in the same pixel, the one closer to the camera's focal point is used. There are pixels in the image that are blank because no points in the color point cloud landed on them. If the cause of this is due to the point cloud being sparse, then the resulting projected image has scattered and small

blank spots. To this end, OpenCV's In-Painting algorithm is used to fill them, seen in [FIGURE] [REF]. However, if the blank spots are caused by occlusion due to head pose, the spots are larger and more concentrated, and the In-Painting algorithm may be doing a poor job [FIGURE]. This only happens at the eye regions at extreme head poses, where other sources of distortion errors (e.g. head pose tracking inaccuracy) also come in. Thus In-Painting being inaccurate in this case is tolerable, and treated as a limitation.

Using the EYEDIAP Dataset

Our ultimate goal is to predict the user's gaze given the depth and color video streams from Kinect. So far, we are able to extract a frontal pose corrected color images of the eye regions. The next step is to train a gaze predictor so the gaze direction can be predicted given a pair of eye region images. To this end, we obtained the EYEDIAP Dataset [REF].

The EYEDIAP Dataset was created by the IDIAP Research Institute for the purpose of training and evaluating gaze prediction algorithms using depth and color video streams. It consists of 16 participants, 12 males and 4 females. Participants are asked to gaze follow visual targets while being recorded by a Kinect1 camera and a HD video camera. For each participant, three visual target conditions are recorded: target on a computer screen changing positions discretely, target on a computer screen changing positions continuously, and a small moving ball floated by a long pole moving continuously. For each condition, two sessions are recorded, with one requiring the participant's head to remain still while the other allowing free movement. The videos are annotated automatically for head pose and gaze direction. This is done by using the 3D Morphable Model (3DMM) algorithm for head tracking, knowing the location of the visual targets on screen, and extracting the location of the floating ball target from the depth data. Using the annotations, gaze predictors can be trained and evaluated.

To use the EYEDIAP Dataset in our framework, modifications to KinFu were made. First, instead of subscribing to the video sources of a Kinect camera, the depth and color videos of the dataset are read frame by frame using OpenCV. Note that we are not using the Kinect1 camera's color videos; instead, the higher resolution HD camera videos are used along with the Kinect1's depth videos. Depth frames are undistorted using the distortion calibration values provided, following the method used in the open source calibration toolbox by [Herrera C. et al. 2012]. Also, image buffers' sizes are changed to match the HD video camera's resolution (1920 X 1080) and the Kinect1 depth camera's resolution (640 X 480). Next, to form color point clouds, we need to map from depth frame's image coordinate to the world coordinate for forming the face model, and map from the world coordinate to the color frame's image coordinate for assigning mesh surfaces to color pixels. These mappings are done by transforming the coordinates using the cameras' extrinsics and intrinsics provided [FORMULA]. During processing, the ICP alignment loop along with the color point cloud formation and projection reside in one thread, while the video frames grabbing reside in a separate thread. Thus, synchronization is needed between threads to ensure no frame skipping when the processing thread is slow. Lastly, Skeleton Tracking is only available if a dataset is recorded using Kinect Studio, thus it is not available in the EYEDIAP dataset. Instead, we used in its place the 3DMM head pose tracking annotations by frame provided in the dataset. We only used the translation part of the head tracking, and only needed it for initialization of ICP – we still rely on ICP for orientation alignment in the first frame as well as full head pose tracking in new frames after that.

Evaluation On Child With ASD Videos

5.1.3 Eye Pose Estimation

Eye Image Cropping and Stabilization

The appearance-based method we will use follows after Mora et al. (Mora & Odobez, 2013), and requires cropped eye images from frontal head pose. As explained in previous section [REFsection], the cropped color eye images are easily obtained as long as we have stable tracking of the head – the relative position of the eyes on the head is unchanged although the head is moving. Given the tracked head pose, we can reverse the viewing angle and project back the RGB image to obtain frontal head pose eye images.

Eye Image Descriptor

We first convert the eye image to gray-scale, normalize intensity values by setting mean to 125 and standard deviation to 30 (given that original intensity range is $[0, 255]$). Then, we bin the image pixels into a grid of 3X5. We form the descriptor e as the concatenated vector of bin values, normalized such that elements of e sum to 1.

Adaptive Linear Regression (ALR)

For a single eye, the gaze estimation problem can be formulated as the following: given training examples (e_i, g_i), input e', we want to estimate the gaze g'. Let E be the matrix whose i^{th} column i is e_i, G be the matrix whose i^{th} column is g_i, ϵ be a tolerance parameter, we formulate our problem as a sparse reconstruction problem, finding the optimal w by minimizing the L_1 norm of w:

$$w' = argmin_w \|w\|_1 \quad s.t. \quad \|Ew - e'\|_2 < \epsilon$$

, then the estimated gaze

$$g' = Gw$$

Coupled Eyes Constraints

Now considering both eyes together, the ALR equation holds if we redefine the following:

$$e = \begin{bmatrix} e_l \\ e_r \end{bmatrix}$$

$$w = \begin{bmatrix} w_l \\ w_r \end{bmatrix}$$

$$E = \begin{bmatrix} E_l & 0 \\ 0 & E_r \end{bmatrix}$$

and

$$g = \begin{bmatrix} g_{\phi l} \\ g_{\theta l} \\ g_{\phi r} \\ g_{\theta r} \end{bmatrix}$$

the vector of (pan, tilt) angles of the (left, right) eye

Then, the coupled eyes constraints can be formulated as:

1. Left and right eyes tilt angles should be the same:

$$g_{\phi l}^T w_l - g_{\phi r}^T w_r = 0$$

2. Left and right eyes pan angles should not differ by more than a threshold, τ_ϕ, with left eye the bigger angle

$$\tau_\phi < g_{\phi r}^T w_r - g_{\phi l}^T w_l < 0$$

Solving ALR with Coupled Eyes Constraints

The ALR with coupled eyes constraints can be solved as a Second Order Cone Programming problem (Mora & Odobez, 2013; Ziemke, 2001).

Training Examples Collection and Model Selection

Since we are dealing with children with ASD, collecting person specific training examples would not be possible. Instead, generic training examples across multiple normal individuals will be collected.

We will ask each person to gaze follow a small target placed between the person and the Kinect. As the target moves to different locations, data is collected. Since the location of the eyes and the target are both known through Kinect data, the eye pose can be calculated. The number of eye poses collected per person, the number of people, and whether adult's eyes can be used here are to be determined through some preliminary testing.

After training examples are collected, eye pose estimation for a new person is done through ALR searching through the training examples. We keep track of which person's training examples are used more often, and only keep the top few. This is done by accumulating a running sum of wi for each person. This way, people that have eye appearances greatly differing from the new person are ignored, making the search more efficient and the estimation more accurate.

5.2   Object Identification

Given the gaze directions, in order to know which object is being looked at, we need to know the objects' locations. For the purpose of the pilot study, we will assume locations are fixed during the trial and calibrate their positions before the trial.

The calibration should be performed by a person 3D head model has been learned and whose eyes are part of the ALR training examples for best gaze estimation accuracy. The person is asked to look at each object when prompted while walking around, and gaze directions are recorded. The intersection of two gaze directions pinpoints an object's location. However, because the objects are larger than pinpoints, and gaze directions have errors, we describe each object location as a Gaussian ellipsoid, with mean and variance calculated from all possible intersections of recorded gaze directions [FIGURE].

We use a simple heuristic when identifying the object under gaze. If gaze direction lies within x variance away from an object location mean, then this object is considered to be gazed at (x is a sensitivity parameter chosen manually). If the gaze lies in two or more objects' vicinities, the object

whose distance from gaze normalized by variance is less is chosen as the object under gaze. If no object is close to the gaze direction, the person is considered not looking at any object, i.e. idling.

5.3 Discussion

Chapter 6

Conclusion

Bibliography