ONLINE MINING OF HUMAN DEEP INTENTION BY PROACTIVE ENVIRONMENT CHANGES USING DEEP NEURAL NETWORKS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

NUR BAKİ ER

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

DECEMBER 2015

Approval of the thesis:

ONLINE MINING OF HUMAN DEEP INTENTION BY PROACTIVE ENVIRONMENT CHANGES USING DEEP NEURAL NETWORKS

submitted by **NUR BAKİ ER** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Gülbin Dural Ünver _____
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Gönül Turhan Sayan _____
Head of Department, **Electrical and Electronics Engineering**

Prof. Dr. Aydan M. Erkmen _____
Supervisor, **Electrical and Electronics Engineering Dept, METU**

Examining Committee Members:

Assoc. Prof. Dr. Afşar Saranlı _____
Electrical and Electronics Engineering Dept., METU

Prof. Dr. Aydan M. Erkmen _____
Electrical and Electronics Engineering Dept., METU

Prof. Dr. Erhan I. Konukseven _____
Mechanical Engineering Dept., METU

Prof. Dr. Uğur Halıcı _____
Electrical and Electronics Engineering Dept., METU

Assist. Prof. Dr. Kutluk B. Arıkan _____
Mechatronics Engineering Dept, Atilim U.

Date: 11.12.2015

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: NUR BAKİ ER

Signature:

ABSTRACT

ONLINE MINING OF HUMAN DEEP INTENTION BY PROACTIVE ENVIRONMENT CHANGES USING DEEP NEURAL NETWORKS

Er, Nur Baki

M.Sc., Department of Electrical and Electronics Engineering

Supervisor: Prof. Dr. Aydan M. Erkmen

December 2015, 99 Pages

This thesis focuses on surfacing human deep intention, which is known or assumed, in a smart environment that consists of autonomous robotic systems which can interact with the human. Deep intentions are defined as kind of actions that humans would like to behave but pushed deeper in the stack of the intentions in a daily life. The purpose of the designed system is to observe the human in the smart room for a while and to analyze human's behaviors to offer the optimal set of system behavior to surface a desired deep intention. Deep neural networks classify people implicitly by trained deep learning architecture and outputs the set of system behaviors to trigger deep intention. The autoencoders implemented in the network generate better and compressed representation of input vectors by creating feature vectors without using any feature extraction method. In addition autoencoders also enable the system to have better initialized parameters. This thesis work introduces our novel approach of surfacing human deep intention by utilizing human robot interaction in a smart environment.

Keywords: Human-Robot Interactions, Deep Neural Networks, Autoencoders, Intention Surfacing, Deep Intention

ÖZ

DERİN NÖRAL AĞLARI KULLANARAK ORTAM DEĞİŞİKLİKLERİ İLE ÇEVRİMİÇİ İNSAN DERİN NİYETİNİN ORTAYA ÇIKARTILMASI

Er, Nur Baki

Yüksek Lisans, Elektrik ve Elektronik Mühendisliği

Tez Yöneticisi: Prof. Dr. Aydan M. Erkmen

Aralık 2015, 99 sayfa

Bu tezde insanla iletişime girebilen otonom robot sistemleri içeren akıllı bir senaryo ortamında, bilinen ya da olduğu varsayılan derin insan niyetlerinin ortaya çıkarılması üzerine odaklanılmıştır. Derin niyetler insanların gündelik hayatlarında sıklıkla yapmadıkları ya da yapamadıkları ama yapmak istedikleri niyetler olarak tanımlanmıştır. Tasarlanan sistemin amacı, senaryo ortamındaki insanı kamera aracılığıyla gözlemlemek ve analiz etmek, ardından da o insanın derin niyetini ortaya çıkarmak amacıyla robot etkileşimlerini uygulamaktır. Derin öğrenme mimarisi ile eğitilmiş derin sinir ağları insanları açık olmayan bir şekilde sınıflandırıp, sınıflandırmaya uygun robot etkileşimlerini çıktı olarak vermek üzere tasarlanmıştır. Otomatik kodlayıcılar kullanılan girdi bilgisinden sıkıştırılmış özellik vektörleri oluşturmak amacıyla uygulanmıştır. Ayrıca otomatik kodlayıcılar sistemin parametrelerinin daha doğru bir şekilde ilklendirmesini sağlayacaktır. Bu tez çalışması insan-robot etkileşimini içeren akıllı bir ortamda insanın derin niyetinin ortaya çıkarılmasıyla ilgili özgün bir çalışmayı içermektedir.

Anahtar Kelimeler: İnsan-Robot Etkileşimleri, Derin Sinir Ağları, Otomatik Kodlayıcılar, Niyet Ortaya Çıkarma, Derin Niyet

To my precious family

ACKNOWLEDGEMENTS

I would like to enounce my deep gratitude to my supervisor Prof. Dr. Aydan M. Erkmen for her valuable supervision, advice, useful critics and discussions throughout this study.

I am also grateful to my thesis committee members Assoc. Prof. Dr. Afşar Saranlı, Prof. Dr. Uğur Halıcı, Prof. Dr. Erhan I. Konukseven and Assist. Prof. Dr. Kutluk B. Arıkan for their criticism and advices.

I wish to express my endless thanks to every member of my family, especially to my mother Gülcan Er and to my father Mahmut Er for their unconditional love. Without their encouragements and advices, that journey would be endless for me.

I am thankful to Can Görür for his previous contributions on the topic, his shares and help for some parts of this study.

My brother, Muhammet Er, deserves one of the biggest thanks for being involvement of the study actively and supporting me.

Special thanks is deserved by Burak Çetinkaya for being supportive and helpful companion through this journey. In addition, Gökmen Cengiz and Nehir Utku were always there for me whenever I gave way to despair. Without support of my colleagues Veysel Yücesoy, Ali Aydoğan and Volkan Özdemir, I would have failed while I was trying to overcome difficulties.

I am deeply grateful to Kübra Mutlu, Gizem Akar, Fatih Demir, Cihan Emre Kement, Mehmet Aydoğdu, Anıl Öztürk, Yasin Çevik, Ali Mancar, Kemal Arı, Esra Özdemir, Doğaç Çakır, Mehmet Tekin, Erkan Kılıç, Zeynep Gerem, Aybegüm Demir, Ege Talu, Egemen Yıldırım for their friendships, encouragement and support.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVATIONS

HMM	(Hidden Markov Model)
HRI	(Human Robot Interactions)
OOM	(Observable Operator Model)
NN	(Neural Network)
GLUT	(Greedy Layer-Wise Unsupervised Training)
RBM	(Restricted Boltzmann Machine)
ML	(Machine Learning)
CNN	(Convolutional Neural Network)
MLDNN	(Multi-Layer Deep Neural Network)
MLNN	(Multi-Layer Neural Network)

CHAPTER 1

INTRODUCTION

1.1. Motivation

Latest progresses in the field of artificial intelligence and hardware technologies accelerated the developments in the area of Human-Robot Interaction. In these days both in the daily life and industrial area, usage of the social robots to help people take places (Erden & Tomiyama, 2010). Since human thoughts and behaviors may vary in a large scale and not be predictable, the robots should analyze its interaction with human to assist the human in daily life (Yokoyama & Omori, 2010). In order to increase the social intelligence of the robots, the ability of understanding human intentions and behaviors should be developed.

In social cognition, reshaping intention is a growing study area in human-robot interactions. Previous studies, accomplished in our laboratory (MRC lab) opened new era to autonomous cognitive robots in a real life scenario. The seminal concept human intention reshaping by the contextual moves of remotely controlled robots (Durdu, Erkmen, & Erkmen, 2012). Human confidence was then introduced in the generation of intention transient by Gorur et. al (2014) who developed a planning strategy to break the obstinance of the person and attracting human intention towards the desired one. The work provided a new perspective of intention trajectories based on sequences of intention transients. With this thesis the novelty of the concept was geared towards the generation of the “intention mining” concept where robot interactions are used to make human’s deep intention surface with a real life and smart environment using deep learning architecture. By utilizing advantages of deep learning, without requirement of image processing human behavior is analyzed with stacked autoencoders in deep neural network.

This thesis is built on a real life scenario to show that robots can be utilized in human's daily life in smart homes where robots can help human achieve their goals by surfacing their deep intentions about a hobby like book reading, painting or playing guitar. Surfacing deep intentions that the human may not motivate himself/herself to execute without use of robots can also be used for the elderly to recover forgotten behavioral expression of the intentions

1.2. Problem Definition and Our Approach

In our novel approach of deep intention surfacing, the smart environment is designed to carry social cognitions from the visual observation which consist of human trajectories, hand gestures and postures. The system is able to analyze human behavior and human robot interaction in the smart environment. The main purpose of the system is to surface the deep intention of the human which is assumed or known from previous experiences. In order to trigger the human to act the system tries to establish a correlation between human behavior and system behaviors which are defined as the set of inputs to manipulate human to surface deep intention. Then a chosen set of system behaviors is activated to make autonomous changes in the smart real life environment in a defined logical order. Deep neural network architecture is built up to form a correlation analysis between human behaviors and system behaviors. In order to benefit from deep architecture low level feature extraction methods are applied to the visual data obtained from the camera input. The behavior analysis of the human user are implemented by stacked autoencoders network with layer-wise greedy training methods. According to the results obtained from the neural network architecture, the smart autonomous real life environment applies the appropriate input set for the human user.

1.3. Contribution

The contributions of this thesis are given as:

1. Making deep intention of the human surface by fully autonomous smart environment
2. Developing a smart real life environment with fully autonomous robots which are designed to guide human to surface deep intentions
3. Adaptation of deep neural network architecture to generate a correlation between human behaviors and deep intentions

1.4. Outline of Thesis

In the present section, outline of the thesis is explained. In this chapter, aim of the thesis, motivation and contributions are discussed.

Chapter 2 explains survey of the literature about the milestones of the thesis, which are human robot interactions, intention estimation and reshaping, deep learning architectures, neural networks and autoencoders.

Chapter 3 details the methodologies in our study that are human behavior analysis with deep neural networks, initialization problem of neural networks and layer wise greedy training, low level feature extraction and input dimension reduction by autoencoders. Design of the smart environment and conducted experiment to obtain the training data is discussed in this chapter.

Chapter 4 gives steps of the experiments conducted to analyze performance of the overall system. In addition, sensitivity analysis is presented in terms of system performance of mining deep intention of reading a book by regarding parameters like the captivity time and the captivity level.

Finally, in Chapter 5 the thesis work is concluded by overviewing the purpose and the results of the study. Possible future works that are appropriate in order to extend the study are also discussed.

CHAPTER 2

LITERATURE SURVEY

Our main intention in this thesis is to trigger deep intention of a human in order to surface it in human behaviors by robot proactive moves in a specific scenario. Thus the milestones of this thesis work are: 1) human robot interaction; 2) deep learning algorithms and applications. Consequently, the present chapter of the thesis will span the survey of relevant literature, grouped according to the milestones.

2.1. Human-Robot Interaction

Since robots are becoming present in human's daily life, it is necessary for robots to gain intuition about human robot interaction. For example, in the industrial area robot arms are used to assist people to handle force needed task easily. In order to make this interaction more effective and intuitive, it is necessary that the robot anticipates the human intention (Awais & Henrich, 2013). In other words, in order to decrease any limitations to interaction and to make the human robot symbiosis more sustainable, it is important that robots read human intention for natural and optional interaction with human beings. Since human understand the meaning of someone's gesture or non-verbal cues, he/she can read the opponent's mind including intention and goals and even predict his/her future ones. Simulation Theory is one of the mind related theories, which suggests that humans use their own mental mechanism to read other's mind like simulation (Han & Kim, 2010). However, due to the fact that human behaviors and emotions are unpredictable when interacting with the robots, the development of the robots that will assist human is become a difficult problem (Koo & Kwon, 2009; Yokoyama & Omori, 2010).

The robots need to behave like a human by understanding human behaviors and deciding how to act against those behaviors in order to established a rational interaction with him/her (Jenkins, Serrano, & Loper, 2007). Hence, understanding of human intention is necessary to increase the quality of the human robot interaction.

It is shown in the research that the difference in culture have a great impact on the human-robot interactions since the human behaviors or responses to particular actions can vary according to which culture he/she is belonging to. In addition, the environment where the interaction is realized, affects human behavior and interaction as well. To illustrate, people are more willing to interaction with a robot or another human in a place like museum than office buildings where the environment is thought to be places for human/human interactions and social grouping of human (Rivera-Bautista, Ramirez-Hernandez, Garcia-Vega, & Marin-Hernandez, 2010). Hence it is important to analyze human's behavior according to the environment and proposing the best way to interact robots with humans to maximize collaboration.

It is shown in the studies that a communication does not consist of only the verbal part. Verbal and nonverbal part of the communication of the human are important for a robot to understand his/her intention, which is necessary for such a interaction between a robot and human (Sato, Yamaguchi, & Harashima, 2007). Body movements, mimics, facial expressions are example of the nonverbal communications which are used to make prediction about human's intention in the studies in the literature. Therefore, when analyzing these features considering their contextual information leads to rational predictions of the one's intentions.

One of the best way to investigate the relation between user's actions and intentions is to classification based on the user modeling via artificial intelligence. According to Kelley (2008) for a robot system to function correctly in human environments, it must recognize and predict human actions quickly and accurately. One of the machine learning method which used to recognize human actions is Hierarchical Hidden Markov Model (HMMM). Zhu et al. focused the classification of the hand gestures by using primitive sensors. In addition to hand gestures, eyeball movement of a human

can be related with the thoughts and behaviors of a human (Schwarz, 2002). In other words, it is possible to detect the intention of a human by considering gaze data in a period of observation (Hwang, Jang, Mallipeddi, & Lee, 2012). Usage of gaze and gestures to control a PC is one of the examples in the literature shows requirement of the estimation of the intention of the human. User actions which are body and gaze movements are mapped to keyboard and mouse control in order to clarify the intention of the person (Ali, Khan, & Imran, 2007).

2.2. Deep Architectures Approach

2.2.1. Deep Multi-Layer Neural Networks

Until 2006 when positive results are obtained in the attempts to train neural networks with at most 2 hidden layer, researcher's study to train multi-layer neural networks were not considered as successful (Bengio & {LeCun}, 2007). However, the increase in the number of hidden layers generally results in unsuccessful performance for the training. After the introduction of the greedy layer wise training in which each layer of the network is trained individually with unsupervised manner (Hinton, Osindero, & Teh, 2006). Then autoencoders are introduced to perform a successful training in the hidden layer of the networks, which can be applied locally to each layer (Bengio & {LeCun}, 2007). In the same approach, more algorithms, which have different proposal than above ones, for this deep networks were introduced (Weston, 2008).

For a standard one hidden layer-neural networks, the training procedure consists of three main steps: namely 1) to define a cost function to optimize the problem for a specific output on the output layer, 2) to apply a gradient based optimization in order to optimize the parameters of the hidden layers which are weights and biases, 3) training the networks by using a set of samples which is defined as training data. However, implementation of these three steps on the deep networks gives not as much successful results as on neural networks with 1 or 2 hidden layers.

The success for a training algorithm for deep architectures can be measured by that how meaningful and complex is the representation of the the input data in the hidden layers but this is considered as difficult to achieve. Hinton (2006), came up with a possible solution for this problem by using binary variables to express the data in a complex and non-linear form with RBM. The reason for a successful initialization for the parameters in the hidden layers of the network can be explained by that this method provides with an abstract representation of the data for each individual layers. Feature vector is obtained by implicitly in order to be used represent the data at the previous layers. By this way, all the layers in the networks comprise a meaningful part of the input data. Autoencoders, which can be seen as a neural network that can extract feature representation from given data, are considered as an example for such method (Figure 2-1).

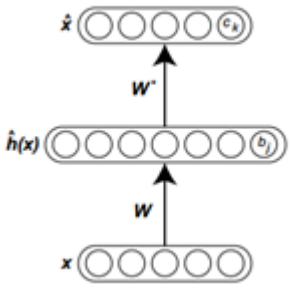

Figure 2-1 Architectural view of the Autoencoder network (Larochelle et al., 2009)

By noticing the similarity between "greedy layer-wise training" and autoencoders, one can implement autoencoders as initialization of the network parameters in the greedy layer-wise initialization part of deep architecture.

Convolutional Neural Network is the one of the alternatives, which are used by Valle & Starostenko (2013) to recognize human walking/running actions from the human silhouette. It is clear that obtaining a meaningful part from any data is not easy and in

neural networks, this process must be completed in the training phase of the networks. Hence, the learning tools in the networks should be able to extract feature for a successful classification. This can be achieved by the CNN, which is able to extract feature vector, which is used for classification process, in a medium level without the handcrafted data processing tools. Valle et al. claim that in order to recognize actions of the human successfully, silhouette of the human can be utilized and analyzed. After obtaining silhouette of the human by extracting the feature vector, which segments BG and FG of a scene by considering the moving objects, using the CNN classification of wide range of different silhouettes can be done.

Generally, in the training phase of the deep architectures stochastic gradient descent is implemented in order to calculate the cost function and to update network parameters like weights and biases. Yu, Wang and Xu (2013) claims that it is difficult to use stochastic gradient descent algorithm in neural networks. Since more than ten passes might be necessary for successful training in neural networks to have optimized parameters, as data set gets larger this process takes more and more time. In order to solve this problem a new solution is proposed namely “average SGD” for deep architecture with large data set.

CHAPTER 3

METHODOLOGY

The aim of this thesis work is to develop a smart environment that can mine the deep intention of human by a set of proactive environmental behaviors which include human-robot interactions and audio/visual focus manipulators. The goal of the entire system is to select appropriate environmental proactive interaction offer to the human by observing movement of the human and analyze it in terms of location and current intention of the human. In the observation phase, human motion is captured and stored as a 320x240 image. The current intention of the human inherent to that motion and in which part of the room the human observed or explored more can be revealed by this data. Studies show that human is more likely to interact with a robot or an environment if he/she is familiar with it. Familiarity makes a human more confident and enables his/her responses to changes in robot or environment (Görür & Erkmen, 2014). These motion data are therefore used as input the Multi-Layer Neural Network which is designed to decide the appropriate environmental interaction offers to direct the human from his/her current steady intention to the desired deep intention by generating transient intentions. The more current intention is migrated into full concentration the more difficult it is to change the current intention of the human. If changes occur despite a fully concentrated current behavior of the human, we postulate to have mined a deep intention. Our intention mining system is trained with the data set collected from experiments conducted with 15 different people. The training set contains the trajectories of these people together with their reactions of the environmental interaction offers. The network is composed of stacked autoencoders so that it can learn feature hierarchies without the requirement of extracting feature vectors by human crafted methods to generate a mapping function of the input and output. In addition, it is difficult to train such a deep architecture with traditional neural network learning procedures like random initialization and stochastic gradient

backpropagation, greedy layer-wise training method is applied (Larochelle et al., 2009).

In this chapter, firstly the experimental setup, which is designed smart environment, will be introduced, together with the labeling of the environmental interaction offer. Then, trajectory extraction and how it is used in the thesis work process will be explained in detail, followed by the introduction of the Multi-Layer Deep Neural Network architecture defined in terms of layer structure, stacked autoencoders and training phase of the network.

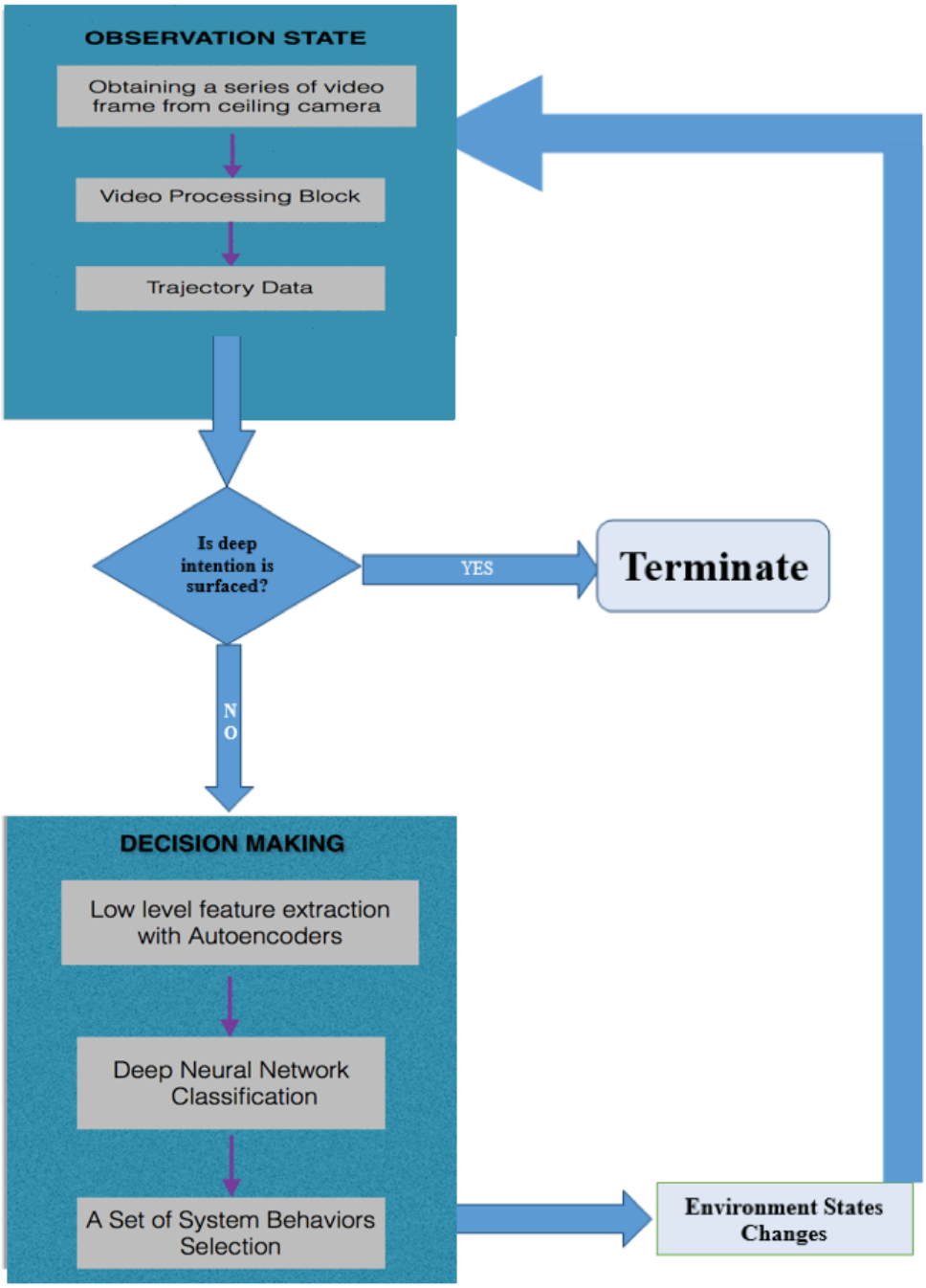

Figure 3-1 Flow chart of the proposed methodology

3.1. Experimental Setup

This room is designed by the inspiration gained of previous work of Gorur et al. (2014) and is equipped with two robots, audio/visual interaction tools that offer tasks that a human can do. The purpose is creating a home-like environment where people can act freely and confidently. Hence the activities which can be done in the room like reading book, playing computer etc. are chosen carefully in order to fit every person's routine behaviors. The general overview of the room can be seen from Figure 3-2 and Figure 3-3 together with labeled activities that they offer.

Figure 3-2 Defined activities namely Drinking Coffee, Playing with computer and reading book.

Drinking Coffee: User can drink coffee in this area marked as 1.

Playing with Computer: User can work with the computer by sitting on the chair robot in front of the table.

Reading Book: User can take a book from the library both by himself or using the chair robot for higher shelves.

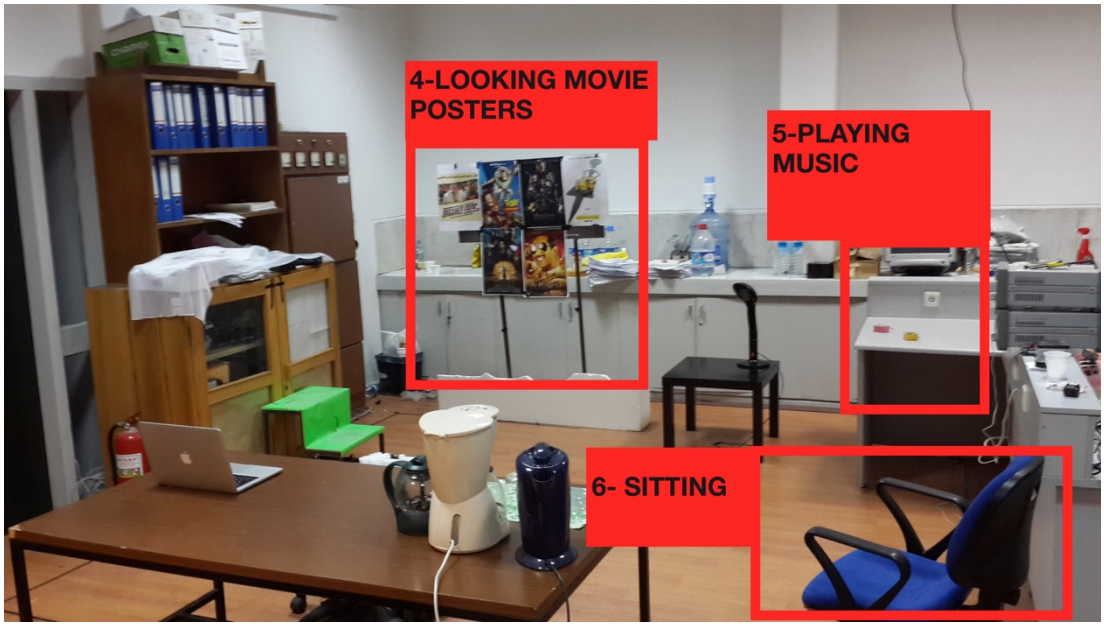

Figure 3-3 Defined activities namely looking at the film posters, playing music with an Ipad application or music boxes and sitting.

Looking at the Movie Posters: User can look and read the information on the different movie posters.

Playing Music: User can both play with the music box placed on the table and use the Ipad application to make music by learning to play piano.

Sitting: Users can simply sit on the chair and observe the environment.

In addition to the defined activities described above, environmental interaction offering tools generate environmental changes to interact with the human and direct him to the desired deep intention. Detailed information about the environmental interaction offering tools are given below the Figure 3-4 and Figure 3-5.

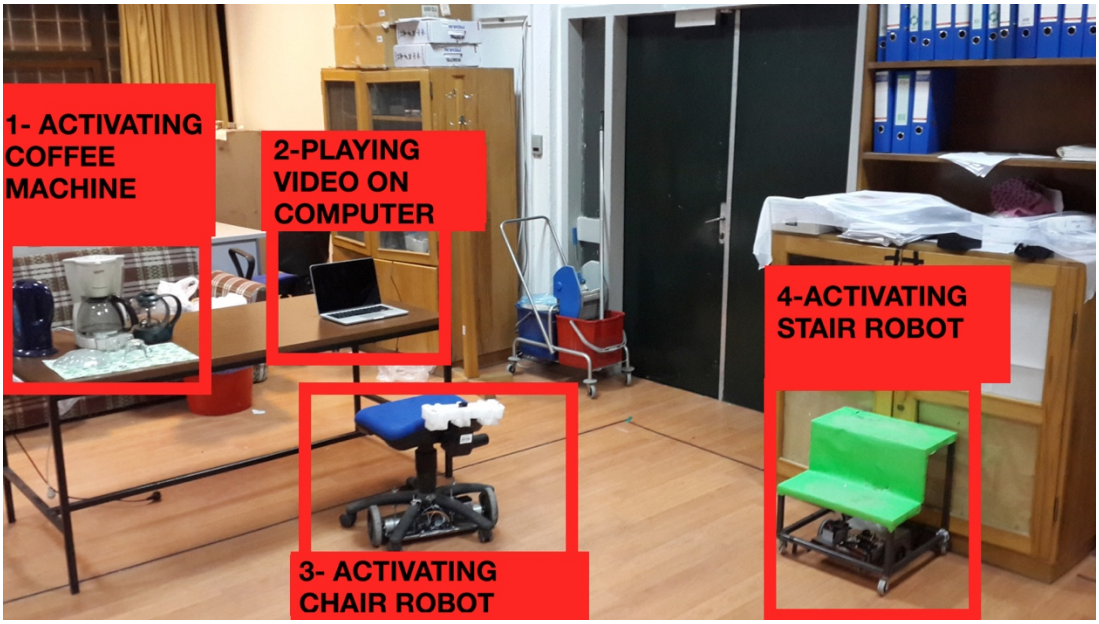

Figure 3-4 Defined environmental interaction offering tools (I)

Activate coffee machine or kettle: The purpose of this behavior is to capture the attention of the human by sound coming from kettle or coffee machine.

Playing a video on computer: The purpose is to captivate the attention of the human by audio and visual context on the computer.

Moving the Chair robot: Aim of this behavior is both to focus attention of the human and give him a message to sit on the chair.

Moving the Stair robot: As results of moving this robot, humans are given the message of stepping on it and reaching up.

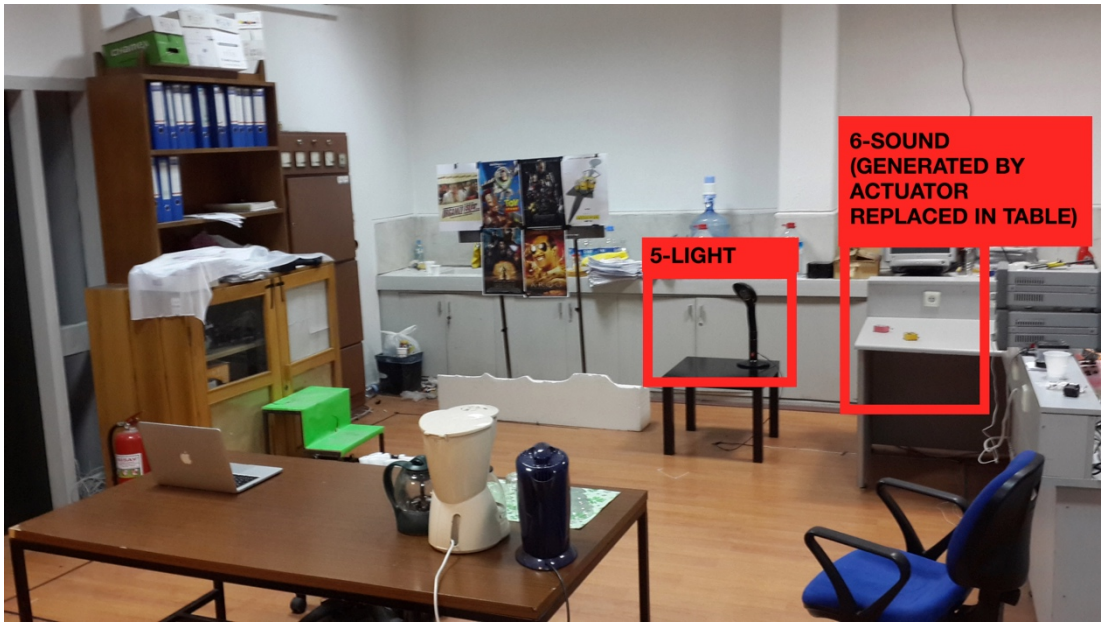

Figure 3-5 Defined environmental interaction offering tools (II)

Turning the light ON/OFF: The purpose is to capture the attention of the user to toys at the table or to different direction than the one used in the current intention.

Making a sound by the actuators placed on the table: The aim is to attract the attention of the user by audio tools to the table.

3.1.1. Chair and Stair Robots

As explained in previous subsections, the main aim of the whole system is to guide people to interact with the smart environment in order to mine the deep intention of reading a book despite any current intention strongly concentrated upon. Among interaction suggestion tools, the ones having large impact of mining human's deep intentions are stair and chair robots. Since they have both the ability to move, they can capture easily human attention and human robot interaction is likely to be established. Both of these robots can be used in changing human localization by making him follow the robot's movement. Hence if human's current intention is away from the desired intention, moving robots can be used to drag the human behind the robot by moving it to the place where the desired intention can be mined by a suitable interaction suggestion tools. Moreover, both stair and chair robot have specific contextual meaning suggesting the human to sit and climb. This situation helps to improve the human robot interaction by iterative intention changes by introducing transient intention states. By moving the robots, the human is directed to a desired intention by transits such as sitting or climbing.

As environmental suggestion tool, stair and chair robots are assigned to different task groups. For example, chair is grouped with computer and coffee machine because people tend to sit down while working on the computer or drinking coffee. Hence at the neural network level, these different components of the system are correlated to be used together. On the other hand, stair is used to direct human to the library to take one of the books. As explained above with the usage of the stair both "follow the robot" and "climb on the robot" messages are given to the human.

Figure 3-6 Stair and chair robots

Both robots are designed to be controlled remotely. Electronic design of the robots can be divided three main components namely: Controller and motor driver circuits, power units, RF receivers and antennas.

Controller and motor driver circuits: Arduino UNO is used as controller boards of both robots and remote controller. This controller is programmed in order to receive commands from RF transmitters and to evaluate this commands by deciding upon necessary moves to be taken. Since these robots are heavy mechanical structures, the motors should be strong enough to easily move them. The average current drained by motors is about 1.5-2 Amps. In order to drive these motors L298 motor driver circuit is chosen which is capable of driving 2 different motor with rated current 4 Amps.

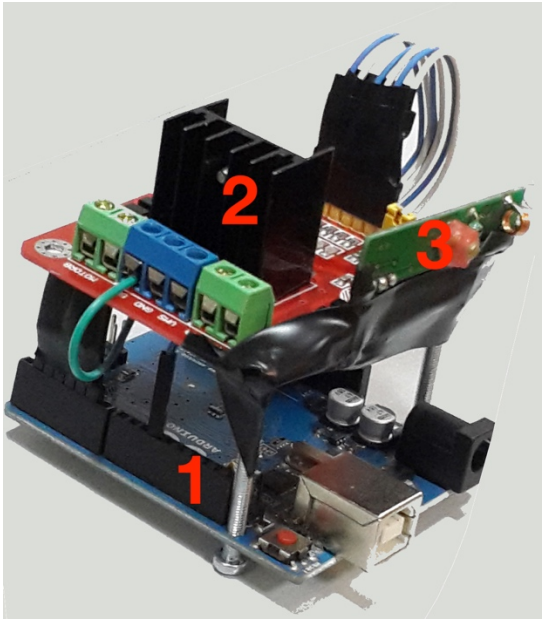

Figure 3-7 Arduino Uno controller board (1), L298 motor driver board (2), RF receiver and antenna(3)

RF Communication Circuits: The robots are supposed to interact with the human so they need to be controlled remotely by the decision mechanism of the intelligent intention mining system. To achieve this, an RF communication protocol with 433 MHz frequency is established between the robot's controller and remote controller. As illustrated in Figure 3-7 with mark 3, a receiver with an antenna is placed to each robot in order to receive the commands about how robots should proceed in moving.

Power Units: Both robots are powered by 12V/2A storage batteries to power Arduino boards, motor controller circuit and RF circuits.

Remote Controller: In order to move the robots according to the output of the neural network, a remote controller is designed. As controller board, Arduino Uno is used again and to transmit data to the robots an RF transmitter is used. According to the selected proactive behaviors set, which is decided by the neural network, robots are

controlled manually by the remote controlled in order to realize the changes suggested by the network.

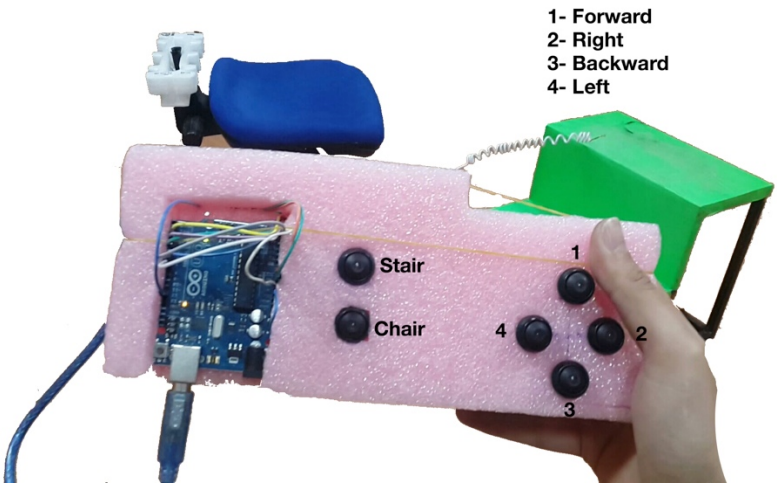

Figure 3-8 Remote controller with Arduino Uno controller board and command switches

3.2. Human Localization and Tracking

The observation phase is accomplished by recording human behaviors and movements in a video with a ceiling camera and processing them. As the output of the processing, human trajectories in the room are extracted and represented by 240*320 image frames which represent real localization of the human with respect to the intention suggestion tools in the smart environment. By extracting human trajectories, it is aimed to reveal which part of the room the human is likely to enter into interaction more and what is the current intention of the human. This information enables the decision making from proposed system to select the appropriate interaction suggesting tool for mining the deep intention of reading book in a human with totally different current intention.

The localization phase consists of five main steps namely; background recording, BG extraction from video, morphological operations, blob analysis and merging. In this

section each of these steps will be explained on real data collected from the experiments.

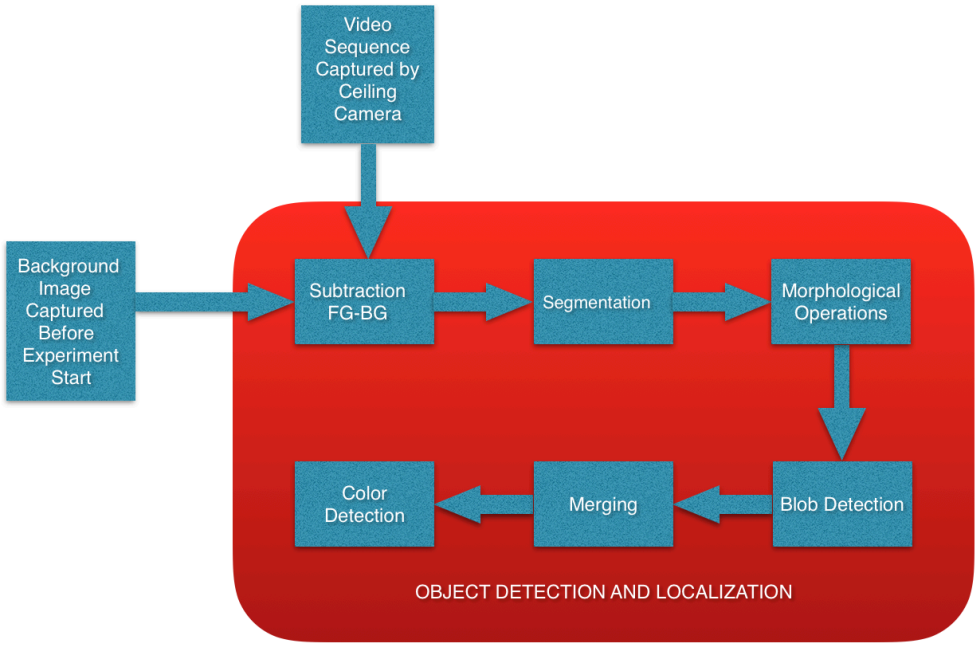

Figure 3-9 Object detection and localization steps in video processing

3.2.1. Object Detection and Localization

The background recording and extraction step is the initial phase of the video processing implementation. The purpose of this phase is detection of the changed pixels and their location by subtracting the current frame, which is obtained from video sequence while human is moving in the room, from the recorded background image.

As shown in the previous study of Gorur et al. (2014), due to illumination changes it might be difficult to obtain a stationary background image. Hence in order to eliminate

this noisy factor, Gaussian Mixture Model is implemented on the previously recorded background frames. The purpose is to estimate a mean value for the BG pixel values.

After this step, a segmentation process takes places to evaluate the pixels of the moving objects in the scene separately from the BG pixels. Otsu Autothresholding method is applied on the image for this purpose. Otsu's method is widely used in computer vision and image processing areas for automatically performing clustering-based image thresholding. Algorithm evaluates images with assumption of containing two classes of pixels following bi-modal histogram namely FG and BG pixels. In the light of this assumption, the aim is to decide upon an optimum threshold in order to separate image pixels into those two different classes. Threshold is decided by minimizing intra-class variance and maximizing inter-class variance.

Figure 3-10 Background (left) and processed video frame (right)

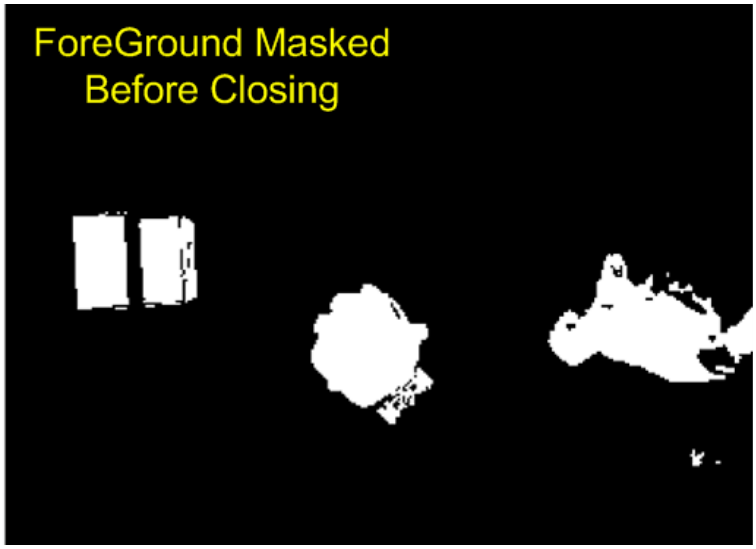

Figure 3-11 Foreground masked before closing operation

After obtaining the image shown in Figure 3-10, it can be clearly seen that there are some holes in the white areas which are representing moving objects. In order to fill these holes, “Closing” is used as a morphological operation. The purpose is to create connected blobs without holes to detect in blob detection part. The structural element is chosen as a rectangle with size of 7x1 pixels. After closing the holes like shown in Figure 3-11, blob detection part is initialized.

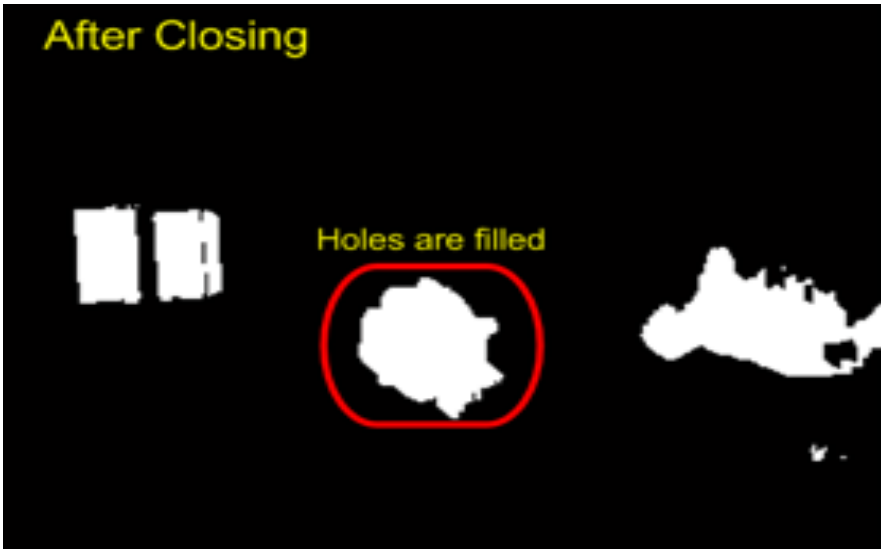

Figure 3-12 Processed video frame after closing operation

As Chen et al. (2007) proposed in their research, their blob detection algorithm is implemented and adapted to this problem. After detecting blobs, it is important to decide which one is human and which ones are robots. Since our focus is on detected human position and tracking him, robots needs to be identified. This is where color identification takes place. It is known that stair robot's color is green and chair robot's color is blue. To distinguish these robots from each other and human, blobs center pixel is calculated and analyzed in terms of RGB value. Since green and blue are main colors, it is easy to distinguish these colors from other objects in the room unless human wore something in these color.

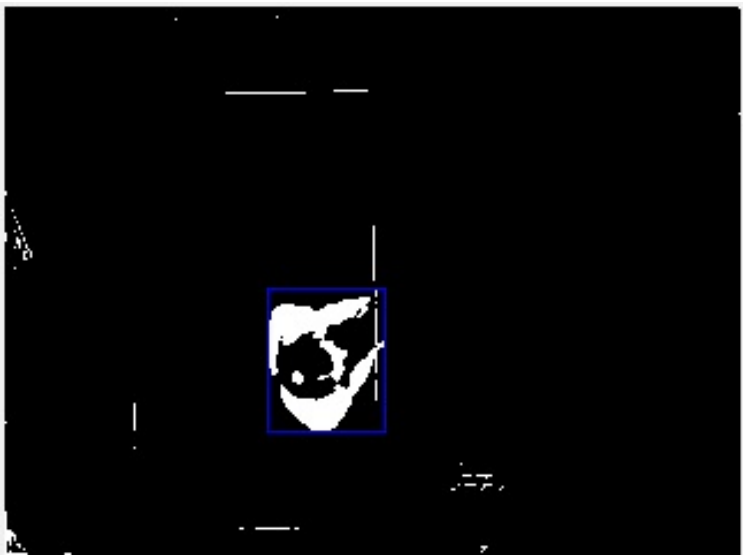

Figure 3-13 – After blob detection human body is extracted and marked in blue rectangle

Center points of the blobs are extracted and form the actual output of the system to track trajectory of the desired object which are human, stair robot or chair robot.

3.2.2. Human Tracking

After detection and localization steps, the center position of the blobs belonging to human and robots are identified. For solving the tracking problem, Kalman filter is implemented by considering its performance on estimating past, present and future states. In tracking algorithm, each object location is compared with previous location in order to test correctness of the detection algorithm. It is assumed that the distance between current and previous location of the robots and human should be smaller than predefined thresholds.

3.2.3. Analysis of Output of the Human Trajectories

The output of the object detection and tracking part of the system (observation state) is the vector representation of the center position of the human throughout his/her entire movement. Before this vector representation is directed to the decision making subsystem based on deep neural networks, this vector is converted to a 240*320-pixel image data. In our study, the purpose is to analyze and extract feature from image data which represents real localization during human or robot trajectories. Feature extractions like human current intention, interest in different parts of the room are handled by deep neural network without the need of handcrafted feature extraction methods. Vector representation of the trajectories is converted to image data as shown in Figure 3-14 and Figure 3-15.

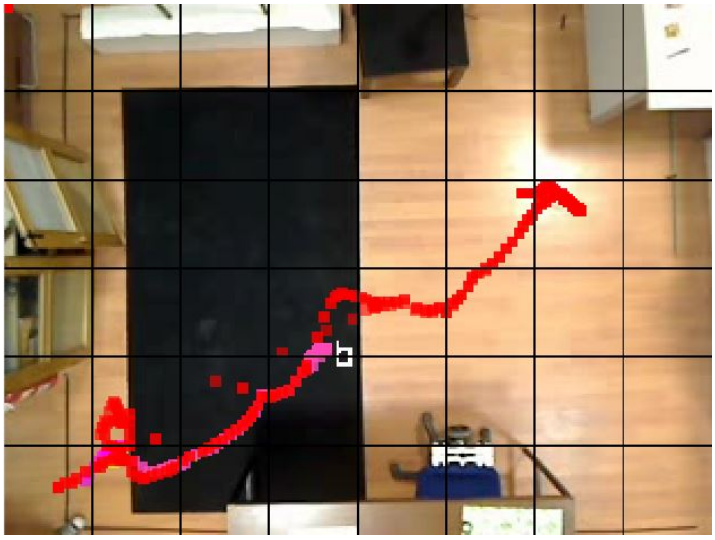

Figure 3-14 Extracted trajectory information of the human moving in the room

Figure 3-15 Trajectory data represented as image

3.3. Deep Network Architecture Implementation

3.3.1. Theoretical Background

3.3.1.1. Artificial Neural Networks

A neural network is defined as two sets N, V and a function w where N is the set of neurons and V is the $\{(i,j)|i,j \in \aleph\}$ whose elements are called connections between neuron i and neuron j. The function w defines the weights, where $w(i,j)$ the weight of the connection between neuron i and neuron j is shortened as $w_{i,j}$.

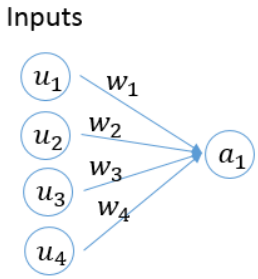

Figure 3-16 - One layer representation of Neural Network

The neuron a_1 is a composition of the inputs u_1, u_2, u_3 and u_4 in terms of weights w_1, w_2, w_3 and w_4 and bias term b. The activation formula for neuron a_1 is

$$a = \left(\sum_{j=1}^{N} w_j u_j \right) + b$$

The inputs and the weight are real values. The b term referred as bias element.

The output value of the neuron is a function of its activation

$$x = f(a)$$

Furthermore, the vector notation

$$a = w^T u + b$$

is useful for expressing the activation for a neuron.

The activation function "f" can be defined as switching status of a neuron. The activation state of a neuron results in the reactions. There are different types of activation functions used in neural networks like;

Linear

$$f(a) = Ka$$

Threshold (Step)

$$f(a) = \begin{cases} 0 & a \leq 0 \\ 1 & a > 0 \end{cases}$$

Sigmoid

$$f(a) = \frac{1}{1 + e^{-Ka}}$$

The activation function selection is very important because of the fact that the small changes in any component of the network should not result in great changes at the output. Hence it affects the robustness of the system. In terms of robustness sigmoid function is more stable compared the two other functions given above.

By using the basics explained above a neural network architecture can be created with the chosen number of neurons and layers. Generally, NN's consist of three layer namely input, hidden and output layer. However, it is possible for a NN to have more than 1 hidden layer. In this case the network is called Multi-Layer Network.

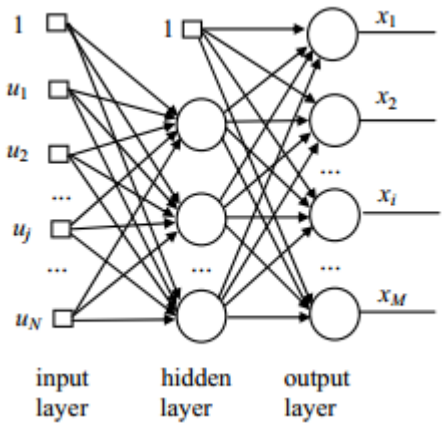

Figure 3-17 – One hidden layer NN with neurons and connections between neurons

According to direction of the connection between different neurons in different layers, NN's are called with different names. In feed forward neural networks, the neurons are organized in the form of layers. The neurons in a layer get input from the previous layer and feed their output to the next layer. In this kind of networks connections to the neurons in the same or previous layer are not permitted. The structure, in which connections to the neurons of the same layer or to the previous layers are allowed, are called recurrent networks.

3.3.2. Training Procedures for NN

During the training process, the weights and bias parameters are optimized in order to create a function between input and output layers. This process is actually an optimization and minimization problem. The parameters which are optimized is weights and bias, the parameter which is minimized is cost function, which is generally described as the Euclidean Distance Function in NNs.

$$J(w) = \frac{1}{2}\sum_{i=1}^{L}(h_w(x^{(i)}) - y^{(i)})^2$$

where $h(w)$ is activation function mapping from X to Y.

3.3.2.1. Stochastic Gradient Descent Algorithm

It is wanted to optimize w so as to minimize $J(w)$. In order to do this, Stochastic Gradient Descent Algorithm (SGDA) can be used with initial guessing about the weight vector. By changing the values of w, the purpose is to converge a value to minimize $J(w)$. SGDA repeatedly performs the update:

$$w_j \coloneqq w_j - \alpha \frac{\partial}{\partial w_j} J(w)$$

where α is called learning rate. This algorithm repeatedly takes a step in the direction of the steepest decrease of J. Also the updates are simultaneously done for all j values from 1 to n.

In order to implement the update, the gradient of $J(w)$ must be calculated.

$$\frac{\partial}{\partial w_j} J(w) = \frac{\partial}{\partial w_j} \frac{1}{2} (h_w(x) - y)^2$$

$$\frac{\partial}{\partial w_j} J(w) = 2 * \frac{1}{2} (h_w(x) - y) \frac{\partial}{\partial w_j} (h_w(x) - y)$$

$$\frac{\partial}{\partial w_j} J(w) = (h_w(x) - y) \frac{\partial}{\partial w_j} \left(\sum_{i=0}^{n} w_i x_i - y \right)$$

$$\frac{\partial}{\partial w_j} J(w) = (h_w(x) - y) x_i$$

Hence the update rule becomes

$$w_j \coloneqq w_j - \alpha \left(h_w^i(x) - y^i \right) x_j^i$$

In this algorithm, this method is repeatedly run through the training set each time taking a single training sample.

3.3.2.2. Backpropagation Algorithm

In the process of derivation of SGD algorithm, it is assumed that NN does not contain many layer and many training samples. If we consider a general NN with one hidden layer, the gradient taking process become difficult. In order to handle this problem, the Backpropagation Algorithm is introduced for training NNs.

Firstly, how backpropagation can be used to compute $\frac{\partial}{\partial W_{ij}^l} J(W, b; x, y)$ and $\frac{\partial}{\partial b_i^l} J(W, b; x, y)$ is described below. The overall cost function $J(W, b)$ can be computed as

$$\frac{\partial}{\partial W_{ij}^{l}} J(W,b) = \frac{1}{m}\sum_{i=1}^{m} \frac{\partial}{\partial W_{ij}^{l}} J\left(W,b;x^{i},y^{i}\right)$$

$$\frac{\partial}{\partial b_{i}^{l}} J(W,b) = \frac{1}{m}\sum_{i=1}^{m} \frac{\partial}{\partial b_{i}^{l}} J\left(W,b;x^{i},y^{i}\right)$$

The intuition behind the backpropagation algorithm is as follows. Given a training example (x, y), first "forward pass" is applied to compute all the activations throughout the network. Then, for each node i in layer l, an error term δ_i^l that measures how much that node was related to any errors in the output. For an output node, it can be directly measured the difference between the network's activation and the true target value and definition of δ_i^{nl} can be done where nl is the output layer. For hidden units, δ_i^l is computed based on weighted average of the error terms of the nodes that uses a_i^l as an input.

In detail the backpropagation algorithm:

1. Perform a feed forward pass, computing the activations for all layers
2. For each output unit i in layer n_l (the output layer), set

$$\delta_i^{nl} = \frac{\partial y}{\partial z_i^{nl}} \frac{1}{2} \left|\left|y - h_{W,b}(x)\right|\right|^2 = -\left(y_i - a_i^{nl}\right).f'z_i^{nl}$$

3. For $l = nl-1, nl-2, \ldots, 2$

   For each node i in the layer l, set

$$\delta_i^l = \left(\sum_{j=1}^{l+1} W_{ji}^l \delta_i^{l+1}\right) f'(z_i^l)$$

4. Compute the desired partial derivatives, which are given as:

$$\frac{\partial y}{\partial W_{ij}^{l}} J(W,b;x,y) = a_j^l \delta_i^{l+1}$$

$$\frac{\partial y}{\partial b_i^l} J(W, b; x, y) = \delta_i^{l+1}$$

3.3.2.3. Training with Backpropagation and SGD

The overall training steps can be written like below,

1. Initialize W and b randomly
2. Set $\Delta W^l := 0, \Delta b^l := 0$ for all l
3. For $i = 0$ to m,
    a. Use backpropagation to compute $\nabla_{W^l} J(W, b; x, y)$ and $\nabla_{b^l} J(W, b; x, y)$.
    b. Set $\Delta W^l := \Delta W^l + \nabla_{W^l} J(W, b; x, y)$
    c. Set $\Delta b^l := \Delta b^l + \nabla_b J(W, b; x, y)$
4. Update the parameters:

$$W^l := W^l - \alpha[\left(\frac{1}{m}\Delta W^l\right) + \varphi W^l]$$

$$b^l := b^l - \alpha\left[\frac{1}{m}\Delta b^l\right]$$

3.3.3. Deep Network Architecture Approach

"Shallow" architectures like one hidden layer NN, support vector machines are base for many existing machine learning algorithms. It is shown in the research that this type of architectures is incapable of extracting some types of complex structure from rich sensory input and they have simple internal representation (Bengio & {LeCun}, 2007). It is also difficult to have large labeled data set to train this system. However, studies show that many layered nonlinear processing are done by visual cortex while recognizing objects with small labeled data set. (Lee et al., 1998). For example, to create a parity function for n-bit inputs it is needed to have a feed-forward NN with

O(log n) hidden layers and O(n) neurons. For the same task if one chooses a NN with only one hidden layer, it is needed to have an exponential number of the same neurons to achieve same goal (Bengio & {LeCun}, 2007).

Backpropagation (Rumelhart, Hinton, & Williams, 1986) was one of first the approaches for multi-layer networks. Unfortunately, this approach does not handle well the increase in the number of layers. Since deep architectures consist of many layers with nonlinear modules, most of the case the problem is a non-convex optimization problem. Because of non-convexity there exist lots of local minima or plateaus, which makes the problem more difficult to optimize. The gradient based optimization like backpropagation with random initialization mostly get stucked at these local minima if the network has more than two or three layers.

By using the standard initialization and training approach for the deep architecture results in poorer results because of the reasons discussed above. Hinton (2006) have proposed a new training approach for training many layered networks which is known as Greedy Layer-Wise Training. The purpose of this algorithm is to pre-train each layer in an unsupervised manner, which solves the problem of random initialization. It provides the hidden layer units, an effective hint about what they should learn. After greedily training each layer, the neural network is fine-tuned with supervised methods.

One of the special NN type which one can use greedily layer-wise training algorithm is called autoencoders or autoassociators. Autoencoders are kind of neural network to have a compact representation of the input vectors from which input should be obtained with very small loss. The obtained representation of the input vector is actually a feature vector of the raw input data.

3.3.4. Greedy Layer-Wise Training

Training multi layered network is a challenging task with backpropagation with gradient based algorithms. Generally, initialization of this process is done randomly, which results in getting stuck at local minima or plateaus because of the non-convexity

of the problem. Each layer in the deep neural networks incorporates of some nonlinearities, which makes the optimization problem more difficult.

Hinton (2006) introduced a new approach to train deep neural networks called greedy layer-wise training. The purpose of this is to give an efficient hint to the hidden layer parameters about what they should learn. The pseudo code for the algorithm is as follows (Erhan, Manzagol, Bengio, Bengio, & Vincent, 2009);

Input: Training set $D = \{(x_t, y_t)\}_{t=1}^{T}$, pre-training learning rate $\varepsilon_{pre-train}$ and fine-tuning learning rate $\varepsilon_{fine-tune}$.

Initialize weights $W_{j,k}^{i} \sim U(-a^{-0.5}, -a^{-0.5})$ with $a = \max(\widehat{h^{i-1}}, \widehat{h^{i}})$ and set biases b^i to 0.

1. Pre-Training Phase

For $i \in \{1, \dots, l\}$ **do**

While Pre-training stopping criteria is not met **do**

Pick input example from x_t from training set

$\hat{h}^0(x_t) \leftarrow x_t$

for $j \in \{1, \dots, i-1\}$ **do**

$a^j(x^t) = b^j + W^j \hat{h}^{j-1}(x_t)$

$\hat{h}^j(x_t) = sigm(a^j(x^t))$

end for

Using $\hat{h}^{j-1}(x_t)$ as input example, update W^i and biases b^{i-1}, b^i with learning rate $\varepsilon_{pre-train}$ according to a layer-wise criterion

end while

end for

2. Fine-Tuning Phase

   In this phase one of the global neural network training algorithms can be implemented like Backpropagation with Stochastic Gradient Descent Algorithm.

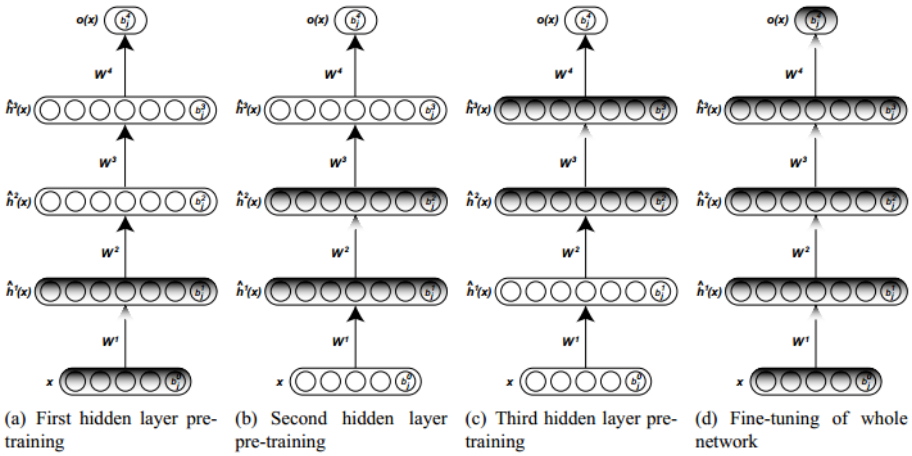

Figure 3-18 - Steps of Greedy Layer-Wise Unsupervised Training Algorithm (Larochelle et al., 2009)

It can be seen that since all the parameters are initialized by the first pre-training phase, the whole networks with fine-tuning phase will be effected. The aim of pre-training phase is to create system parameters which will not be far from the solution found by the fine-tuning phase (Bengio, 2009).

3.3.5. Autoencoders

Autoencoders can be defined as learning circuits aiming to represent input data with more compact way and the least possible amount of loss (Baldi, 2012). Despite of their simple architecture, they play an important role in deep networks. Autoencoders is a way to extract low level feature from the input data without computationally expensive feature extraction techniques. Data representation and feature extraction methods are very important steps for machine learning methods. In order to achieve well represented feature vectors, most of the efforts goes to deploying some machine learning algorithms. This feature extraction process is the main weakness of the current approaches in machine learning because an AI is expected to understand the world around it without clear identification about the sensory data it has (Baldi, 2012).

Autoencoders are used at the center of most of the deep architecture like NN classifiers or Restricted Boltzman Machines. In this network structures, autoencoders can be in the form of stacked and trained by the greedily layer-wise unsupervised training algorithm. These type of networks are shown to be state-of-the-art for the different and challenging classification and regression problems.

Typical autoencoders consist of two operations namely encoding and decoding. The main purpose is to encode the input data to create compact, feature extracted version of the input. Generally encoding part can be implemented to reduce the dimensionality by choosing a hidden layer size smaller than the input size as shown in Figure 3-19 (bottleneck). Decoding part is implemented to preserve the input characteristic while reducing the dimensionality. By decoding hidden layer representation of the input, the purpose is to reconstruct the input vector with small loss. Autoencoders can be considered as a neural network with one hidden layer which have same vector at input and output.

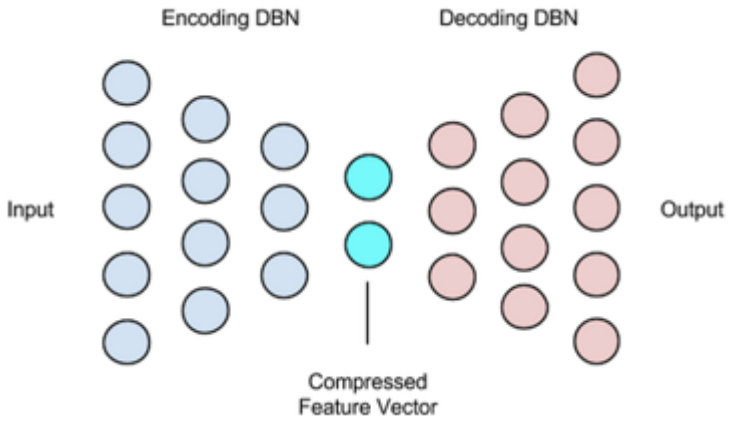

Figure 3-19 General overview of an Autoencoder (retrieved from http://deeplearning4j.org/deepautoencoder.html)

Given an input vector x , representation of x :

$$\widehat{h_j}(x) = f(a_j) \text{ where } a_j(x) = b_j + \sum_k W_{jk} x_k$$

Reconstruction of the input is controlled by decoding function;

$$\widehat{x_k} = g(\hat{a}_k) \text{ where } \hat{a}_k = c_k + \sum_j {W^T}_{jk} \widehat{h_j}(x)$$

Here, the activation functions f(.) and g(.) are selected to be sigmoid activation functions.

3.4. Implementation and Training of Stacked Autoencoders as Multi-Layer Deep Neural Network

Aim of this work is to mine deep intention of a human and more specifically that of reading book in smart social environment. The system input is trajectory information which is extracted from the visual data obtained from the camera at the ceiling of the experiment room. The visual data consists of human trajectories and his/her interactions with the environment. One of the sub motivation of this study is to be able to use the trajectory information as image data as inputs without applying any feature extraction methods. The usage of the deep network architecture is able to realize this motivation with specific type of training algorithm called Greedy Layer-wise Unsupervised Training and specific type of neural network structure called Stacked Autoencoder. GLUT provides the system with training without being stucked at local minima or plateaus by avoiding random initialization of the parameters in the network. Autoencoder are implemented for creating a deep neural network architecture and obtaining feature extraction without applying ML algorithms. The system is firstly trained by GLUT then fine tuning processing is applied with supervised labeled data.

3.5. Training of the Suggestion Generation Module

In order to obtain a data set to train neural network, experiments are conducted on 15 different people by analyzing their behaviors in the designed room. As explained in Section 3.1, each person is instructed to be interactive in the room. They are told that the smart environment waits for them to do one of the 6 different tasks namely, sitting, drinking coffee, looking at movie posters, reading a book, playing music on Ipad application or music boxes and playing with computer. After people enter the room and choose one of the tasks, it is waited that their intention becomes steady. This is measured by the level of focus the person The environment suggestion decision making selects system behavior in order to generate a transient intention towards mining the deep intention of reading a book. System behaviors are being selected among :

a) Activate coffee machine or kettle

b) Playing a video on computer

c) Moving the chair robot

d) Moving the Stair robot

e) Turning the light ON/OFF

f) Making a sound by the actuators replaced in the table

The aim of the smart environment is to apply system behavior for the emergence of transient intention gearing the person towards the mining of a desired deep intention of the human, namely reading book in particular scenario. According to the resulting of human action after the occurrence of the system behaviors, the suggestion of the smart environment is marked as successful or unsuccessful.

99 different experimental data in total is collected from 15 different people. At each time, they are observed to do one of the task available in the room according to their current intentions. By considering corresponded trajectories and interactions with the smart room, different set of system behavior is activated. In these 99 experiment data, people's current intention distribution shown in Figure 3-20:

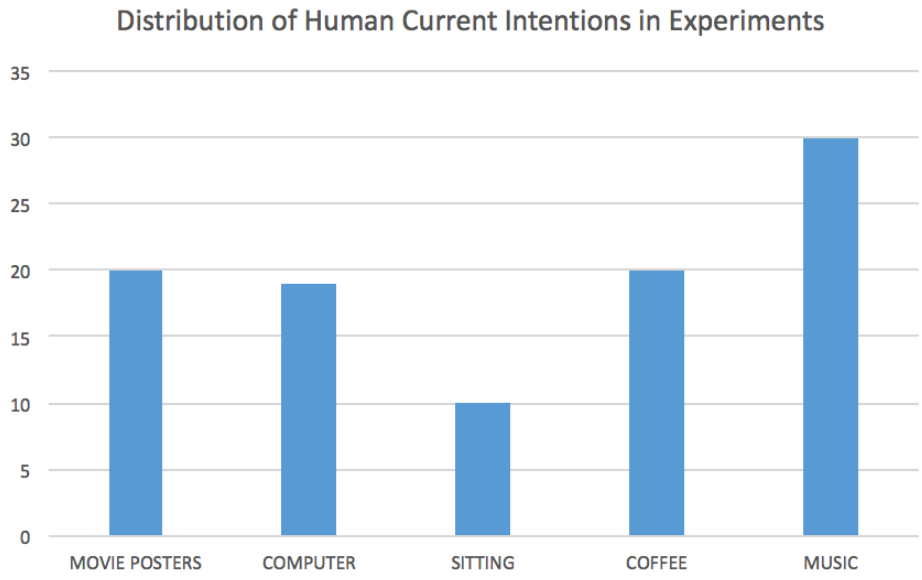

Figure 3-20 Distribution of human current intentions in experiments

Since the prominent aim is to make them read a book, mostly used system behavior after generation of the transient intention is the moving stair robot to the library. Information about activation frequency of system behaviors can be seen in Figure 3-21.

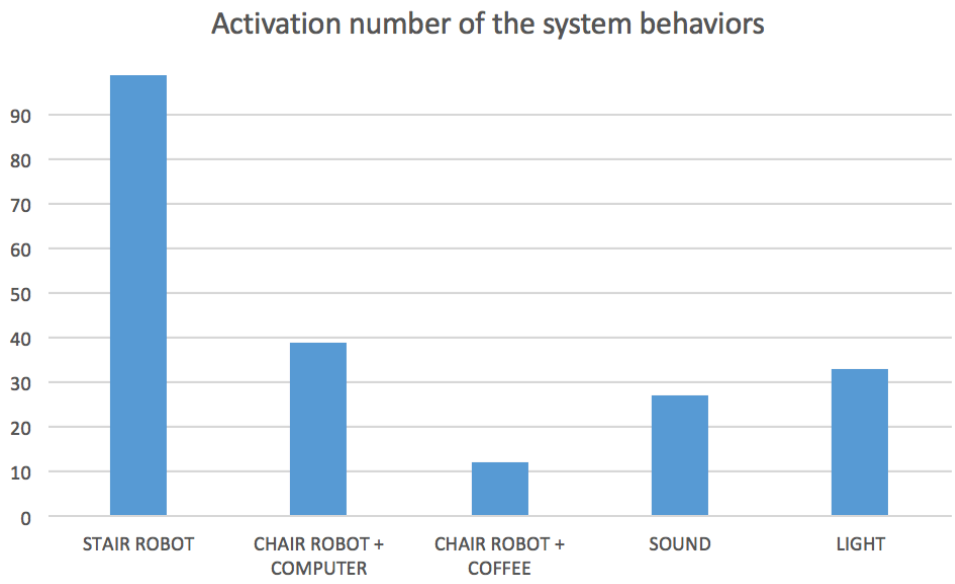

Figure 3-21 Number of activation of the different system behaviors during the experiments to collect training data

As explained in previous sections, video data gathered from ceiling camera are processed in order to extract trajectory information, which are subsequently analyzed by the system behavior suggestion module. One of the training data processing is shown systematically in the figures given below.

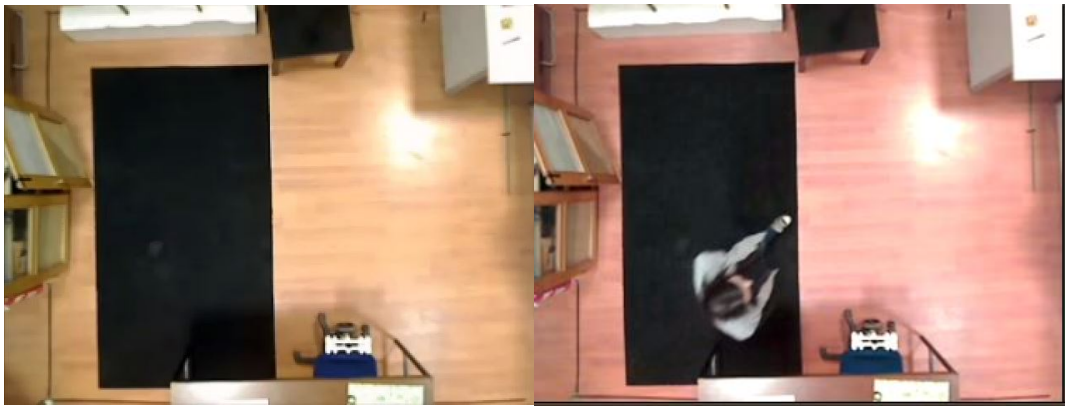

Figure 3-22 Background frame recorded before starting the experiment(left), human is moving in the room (right)

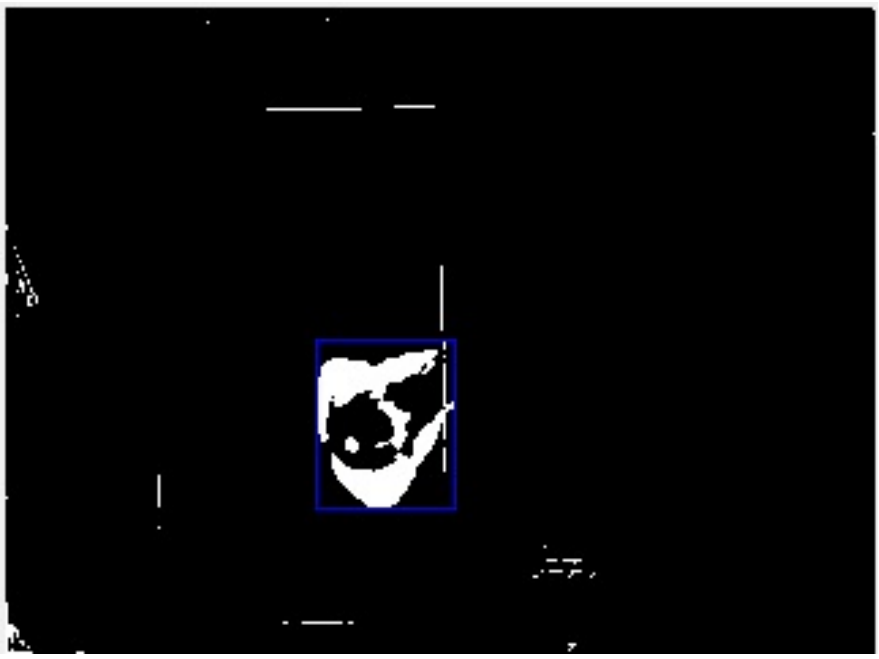

Figure 3-23 Background subtraction from input video followed by segmentation and blob detection

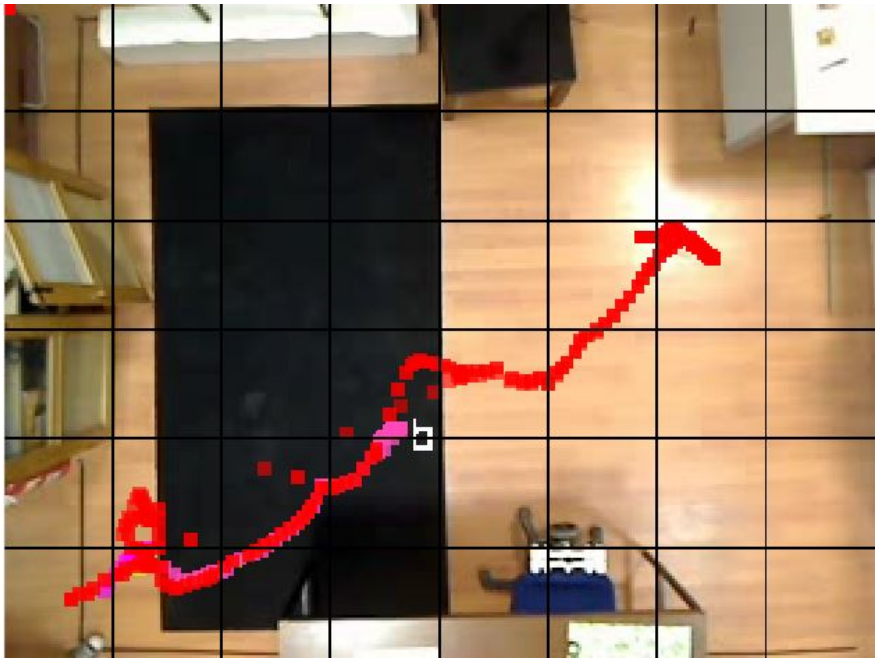

Figure 3-24 Extracted trajectories of the human from beginning to point when current intention becomes steady

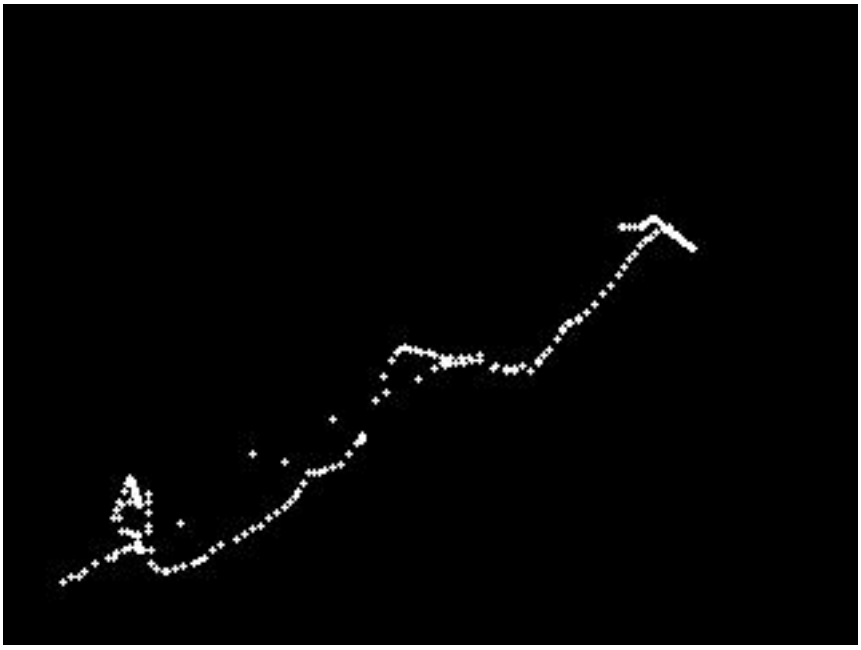

Figure 3-25 Trajectory data is converted to image data representation by preserving the real location room component and human location

After obtaining image representation of the human trajectory in Figure 3-25, a set of system behaviors is applied to the system represented in vector form as:

STAIR ROBOT	1,00
CHAIR ROBOT + COMPUTER	1,00
CHAIR ROBOT + COFFEE	0,00
SOUND	0,00
LIGHT	0,00

The meaning of the suggestion vector is that firstly by using chair robot and computer, a transient intention should be generated. This followed by the activation of the stair robot to reach the ultimate goal of mining deep intention of reading a book. After applying these system behaviors, the human subject started to move the towards library and took a book from it. Hence her deep intention of reading book was quickly surfaced in this example. This behavior set and trajectory information are stored for training phase of the decision making module that will suggest system behaviors in the smart environment after training.

All 99 different training data are analyzed and processed with the steps shown above and used for training of the deep neural network.

3.6. Multi-Layer Deep Neural Network Implementation

In this study according to the information given in previous sections, a multi-layer deep neural network is constructed with stacked autoencoders. The network consists of one input layer, two hidden layer and one output layer. Training of each layer is handled by GLUT algorithm. Each layer's training and construction of the overall network are explained below systematically.

3.6.1. Implementation of the First Autoencoder

As a first hidden layer, an Autoencoder is defined and trained according to GLUT. The data set provides 240*320 pixel images with size of 76800x1 input vector for each training sample. Hence first autoencoders input layer consist of 76800 neurons, in which single neuron corresponds with single pixel in the input image. Sigmoid activation function is used for both encoder and decoder part of the network. The network is trained according to scaled conjugate gradient function. The hidden layer in the network is chosen as 100 because the purpose is both reducing the dimensionality of the input and extracting feature representation from the data set.

The generated Autoencoder is in Figure 3-26:

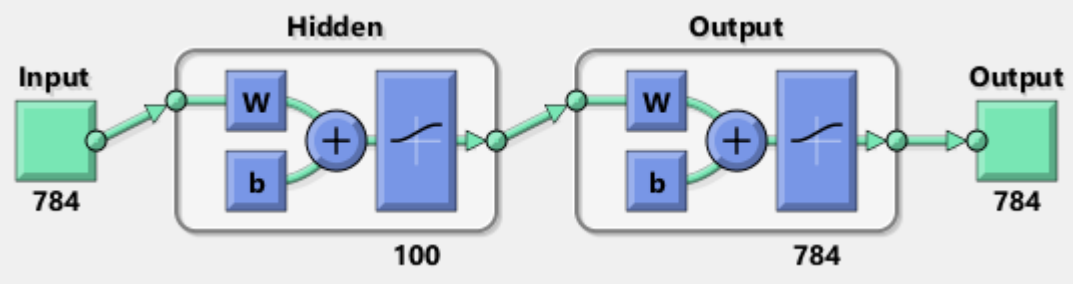

Figure 3-26 - First Autoencoder Network

The input and output of this network are defined to be identical. The purpose is to generate an intermediate representation at the hidden layer with reduced dimensionality. This representation is the feature extraction results from the input.

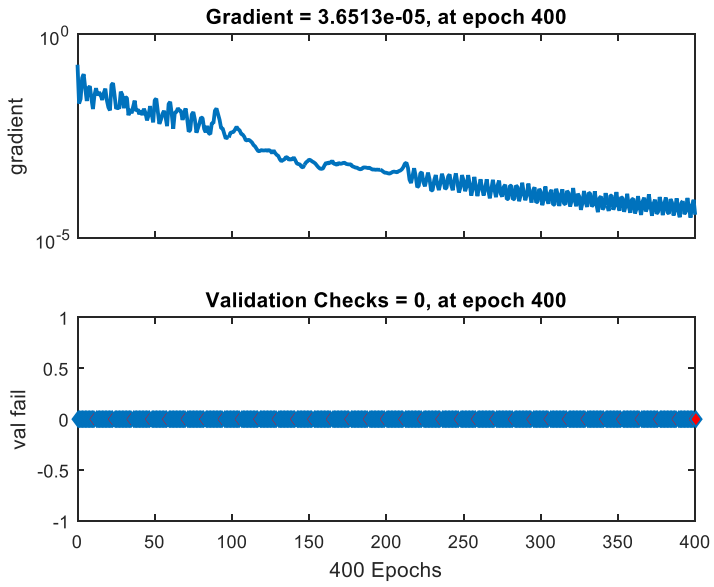

Figure 3-27 - Gradient and Validation Check for Training Process of the First Autoencoder which is First Hidden Layer

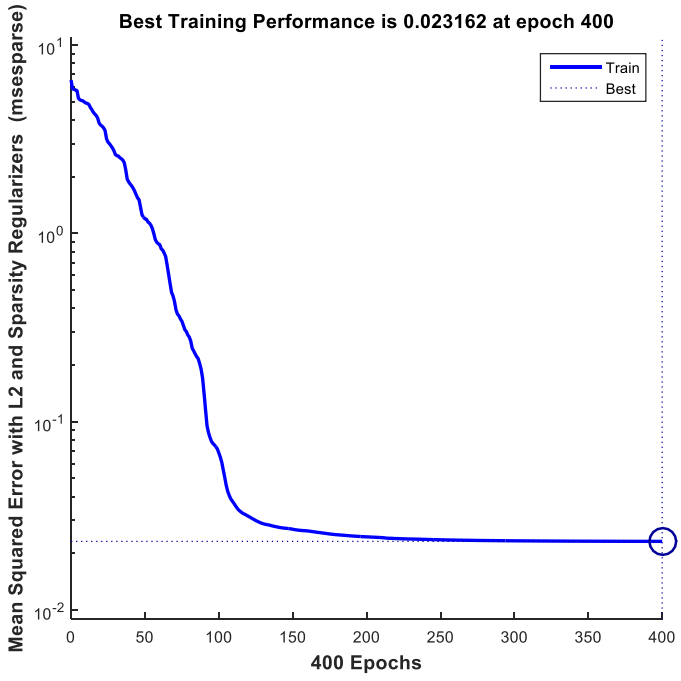

Figure 3-28 - Mean Squared Error obtained Training Process of the First Autoencoder

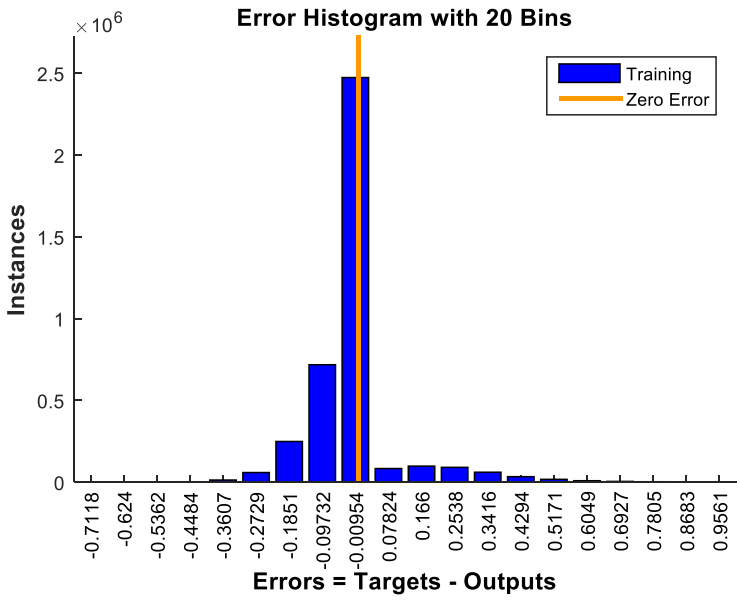

Figure 3-29 - Error Histogram obtained during training

3.6.2. Implementation of Second Autoencoder

After completing the training of the first autoencoder, second autoencoder is trained in same manner. However, the input layer is fed with the feature vector generated at first autoencoder. Hence the constructed autoencoder is in Figure 3-30;

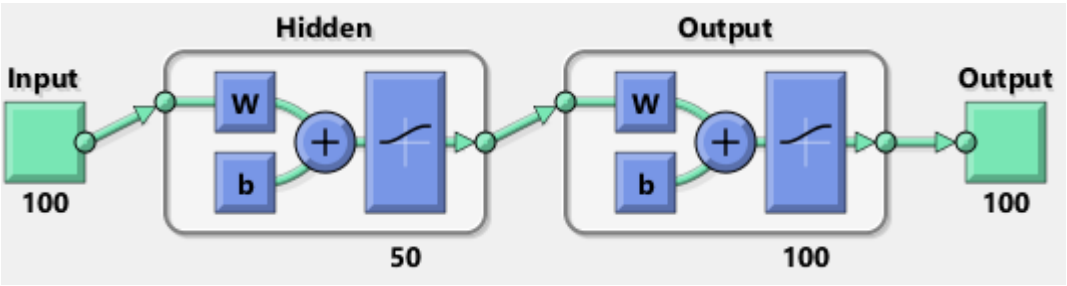

Figure 3-30 - Second Autoencoder Network

The training performance of this network is shown in the following figures;

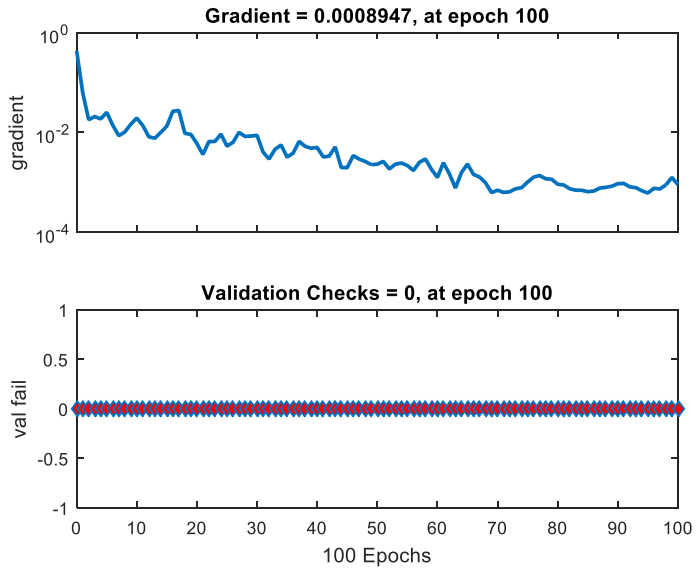

Figure 3-31 - Gradient and Validation Check for Training Process of the Second Autoencoder which is Second Hidden Layer

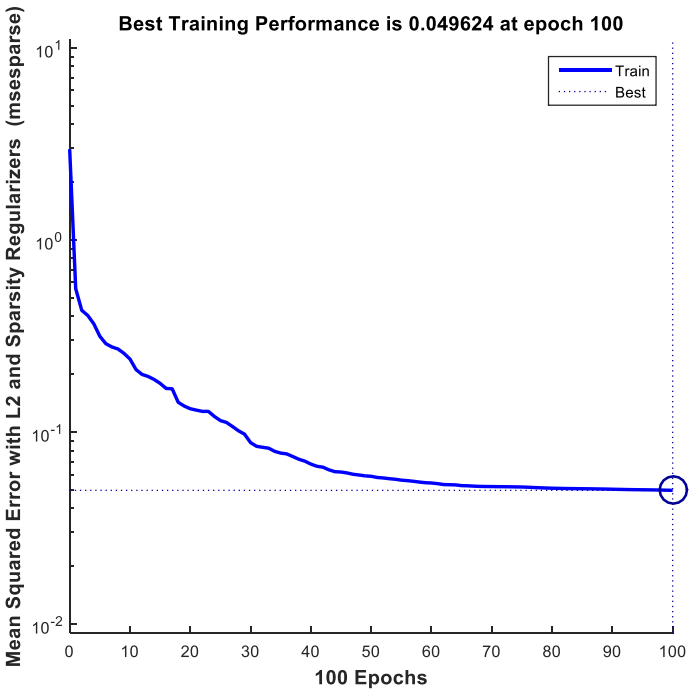

Figure 3-32 - Mean Squared Error obtained Training Process of the Second Autoencoder

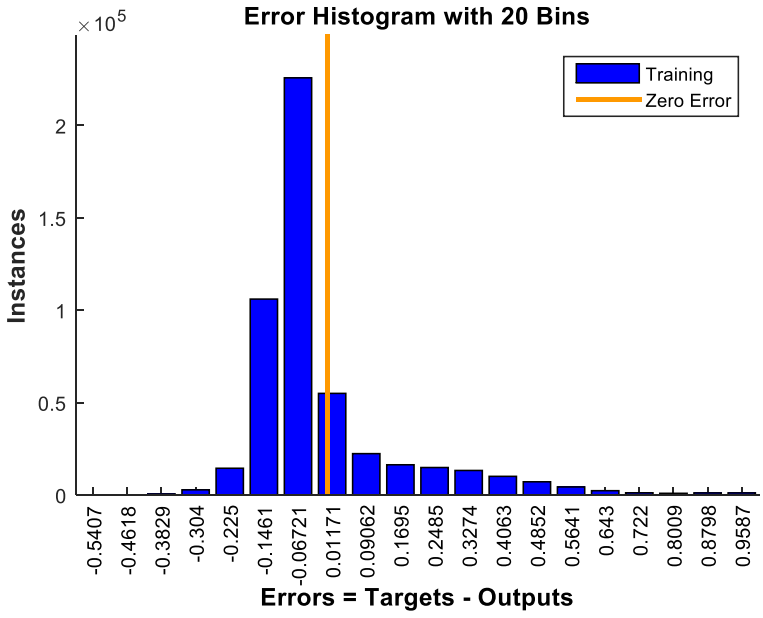

Figure 3-33 - Error Histogram obtained during training of second Autoencoder

3.6.3. Implementation of the Softmax Layer

Softmax layer is defined as the last layer. Unlike the previous hidden layers, it is trained by supervised method using the labels of the training data. However, input layer of this network is still the output vector of the previous autoencoder network. The output of softmax layer consists of 10 neurons.

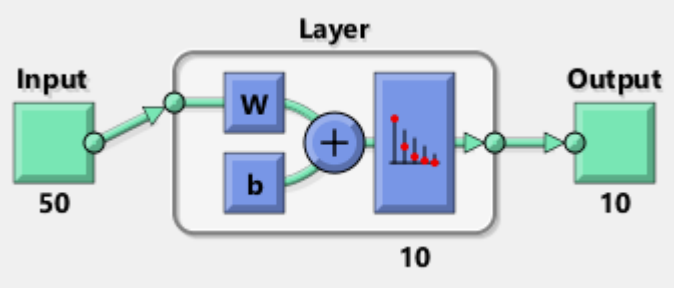

Figure 3-34 - Third Autoencoder Network

The dimension reduction from first Autoencoder input layer to softmax layer can be seen in Figure 3-35;

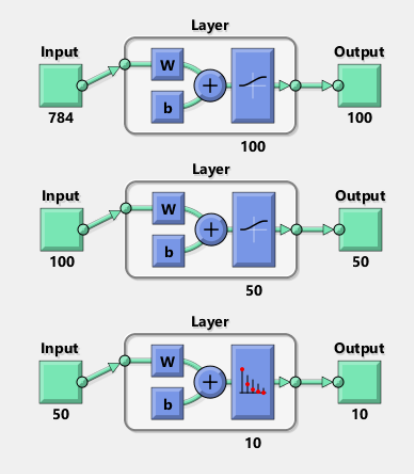

Figure 3-35 - Dimensionality Reduction from First Autoencoder to Softmax Layer

3.6.4. Implementation of Multi-Layer Deep Neural Network Consisting Stacked Autoencoders

After generating and creating autoencoders explained above, those are used to construct Multi-Layer Deep Neural Network with two hidden layer. The output layer of the one Autoencoder is connected to input of the next Autoencoder. The resulted architecture is shown below;

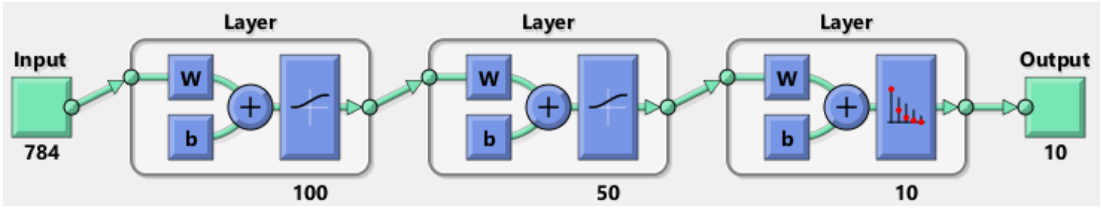

Figure 3-36 - Multi-Layer Neural Network consisting of two hidden layer

CHAPTER 4

RESULTS AND DISCUSSIONS

In this chapter, real world experiments results will be explained and discussed in terms of success rate of the overall system to mine the human deep intentions of book reading that has lost priority and pushed into oblivion. Then sensitivity analysis will be conducted by observing changes in system outcomes when system parameters do change.

4.1. Surfacing Deep Intention of Book Reading Experiments in Smart Environment

In this section, steps of the experiments conducted in the designed smart environment will be discussed. As defined previously, the test experiments are conducted on 12 different people. The experiments aim to mine the deep intention of reading books in all experiments, which are conducted by 12 different people. The system behaviors are tries that disconnect people's attention from their current intentions and directs them to the book-reading zone gradually. The system behaviors are applied by considering the experimenter's current location and that of the library.

4.1.1. Information About Experiment Procedures

As mentioned above, experiments are conducted on 12 different people for test purpose, which is apart from experiments conducted on 15 different people to collect training data under the assumption of that they read book in their daily life but that this intention has lost priority and pushed deeper in human intention stacks losing priorities. The smart environment with pre-defined system behaviors mentioned in

Section 3.1 is used to surface people's deep intention of reading books. Experiments consists of the following main steps:

1. People are informed about smart environment as described in the following sections.
2. After they enter the room, all their movements and behaviors are recorded by the ceiling camera until they select one of the task defined in section 3.1.
3. Recorded 240*320-pixel video data are processed by the observation state components of the system and trajectory information is extracted as a 2D vector consisting of (x, y) coordinates.
4. Trajectory information is converted into 240*320-pixel image data to represent real locations.
5. Image data are fed as input to the trained deep neural network and system behavior set is determined by that network to surface experimenters' deep intention of reading book.
6. According to the results of the previous steps, system behaviors are applied by considering the current intentions and the desired intention.
7. After applying each system behavior, a sufficient time period is let to pass in order for the transient intention to die off.
8. After applying the system behavior decided upon by the neural network, the new current intention of the experimenter is observed and how different the current intention compared to the desired one is evaluated.

The most important part of the experiment procedures is selection of the experimenters and given information about the environment and study. All of the experimenter are not instructed at all about the purpose of the experiment and the existence of the system behaviors in the experiment environment in order to satisfy the objectivity condition. Here is exactly what has experimenters have been told:

1. Experimenters are told to consider the environment as the smart environment, which is one of the room belonging to their own home. They have been introduced the task that they can do in the room namely:
    a. Playing with computer
    b. Drinking coffee
    c. Siting
    d. Playing music box and Ipad piano teaching application
    e. Looking at the movie poster
    f. Playing with the minion toys

    Intentionally they have not been informed anything about the library and reading book action in order to avoid any biases.

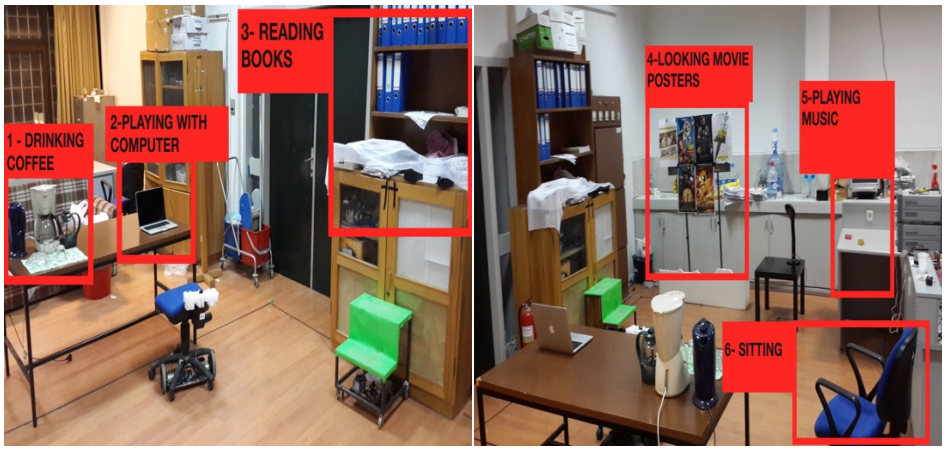

Figure 4-1 Task that can be done in the smart environment. Left side: drinking coffee, playing with computer, reading books and right side: looking movie posters, playing music, sitting.

2. They have been asked to behave like in their home assuming that they have come from the work or school to home. They are told that they can freely interact with anything in the room.
3. The purpose of the information session about the environment is to create a confident relationship between experimenter and the room. It is essential for experimenters to behave comfortable and interact with anything they would like to. As Gorur et al. (2014) stated in his thesis study, the confidence between robots, environment and human is very important to establish human-environment interactions and human-robot interactions. Since the scope of this study does not include the establishment of the confidence between human and robot, we make sure that confidence between human/environment and human/robot is established at the initial state of the experiments.
4. They asked to choose one the task defined in Figure 4-1 after they enter the room and completing the observing room.
5. Experimenters are also asked to stay in the rooms boundaries shown with the black lines on the floor. This is to make sure they stay within the region that ceiling camera covers.
6. Experimenters are also not informed about the system behaviors in the smart environment. They are just told that the environment is a smart home which can interact with them in some ways which are not specified further to the experimenters.

The experiments are conducted on 12 different people whose knowledge about the experiments are limited to information provided by us and mentioned above. However, experiments are divided into 3 cases in order to analyze how the environment changes affect experimenters' behaviors. The main purpose is to have for each case an increase the captivity of the activities explained in Figure 4-1. Three cases are:

a. Simple environment without Ipad and toys
b. Ipad with piano learning applications are with high captiviy level added next to the music boxes
c. Toys are added on the table where light stands.

4.2. Results and Discussions

In this section, the experimental results obtained by 2 different class which are chosen by regarding the captivity level of the environment at that moment of the environment will be presented and discussed.

4.2.1. A simple environment without captivating tasks

In this scenario, every task defined in previous sections exist like playing with computer, sitting, drinking coffee etc. but Ipad with piano learning application and toys are not present in the environment. This scenario will be named for remaining part of the thesis as “simple environment” in the following sections.

In Figure 4-3 one of the experimenters’ video scene is presented while he is heading to “Sitting” task of the room. From the scene analysis, experimenter is avoiding the interaction with the other task in the room but he is observing the parts while moving. In Figure 4-3(f) it is seen that experimenters current intention is “Sitting”. According to training data, collected from different people, Figure 4-2 shows distribution of the selected behaviors to mine the deep intention of reading book when the current intention of the experimenter is “Sitting” for 10 different people. From Figure 4-2 it can be inferred that except from the stair robot, “chair robot + computer” is likely to be chosen by the network and “chair robot + coffee” is not likely to choose by the decision mechanism of the system. Thus the chair robot moves in order to induce the transient intention that will aid at surfacing the desired intention of reading a book that is most apparent in the current status of the experimenter.

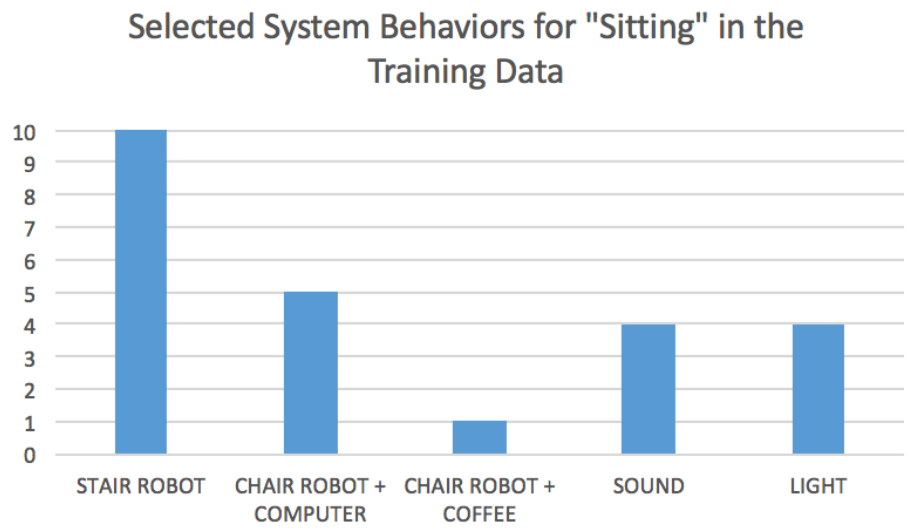

Figure 4-2 Distribution of the selected behaviors to mine deep intention of reading book when current intention of the experimenter is “Sitting”

Figure 4-3 One of the experimenters video scenes while heading to “Sitting” task which is shown in (f).

In the observation state of the study, video data of the experimenter is analyzed and trajectory information is extracted and converted to image data as shown in Figure 4-4.

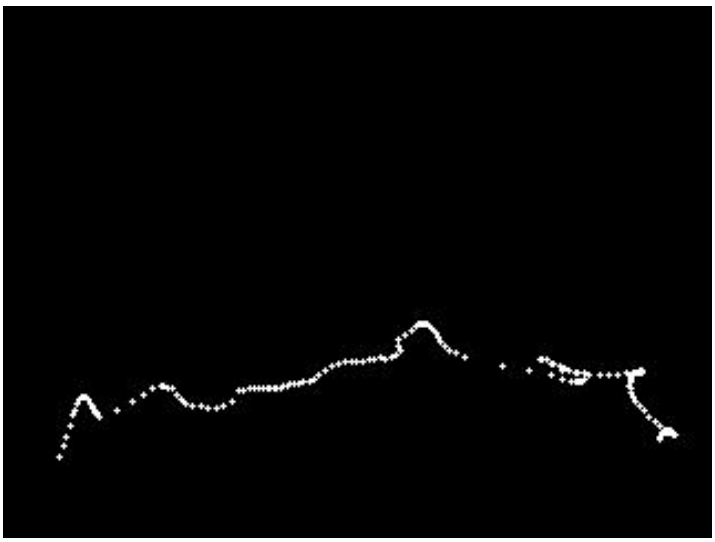

Figure 4-4 Trajectory information of the experimenter converted into 240x320 pixel image data.

This image data is fed to the previously trained Deep Neural Network to generate system behaviors decided automatically by the system to surface the deep intention of reading a book for experimenters whose current intention is sitting. Output of the network is a 1x6 vector whose rows are stair robot, chair robot, computer, light, sound and coffee machine respectively. The values show the weight of the selection that corresponds to the system behavior that will be chosen to generate transient intentions.

$$\begin{bmatrix} Stair\ Robot \\ Chair\ Robot \\ Computer \\ Sound \\ Light \\ Coffee\ Machine \end{bmatrix} = \begin{bmatrix} 0.5413 \\ 0.3552 \\ 0.0997 \\ 0 \\ 0.0056 \\ 0 \end{bmatrix}$$

According to the output vector for the input data of Figure 4-4, the system behaviors set which is selected by the neural network to be Stair Robot and Chair Robot + Computer. This means that firstly the chair robot will be activated to to capture experimenter's attention and move towards the computer so that experimenter will follow him to the computer and play with it. After experimenter intention of working computer becomes steady, the stair robot will be activated in order to take the experimenter's attention and make him leave the computer. The chair robot's move is towards the library to in order to ease the surfacing of the deep intention of the human.

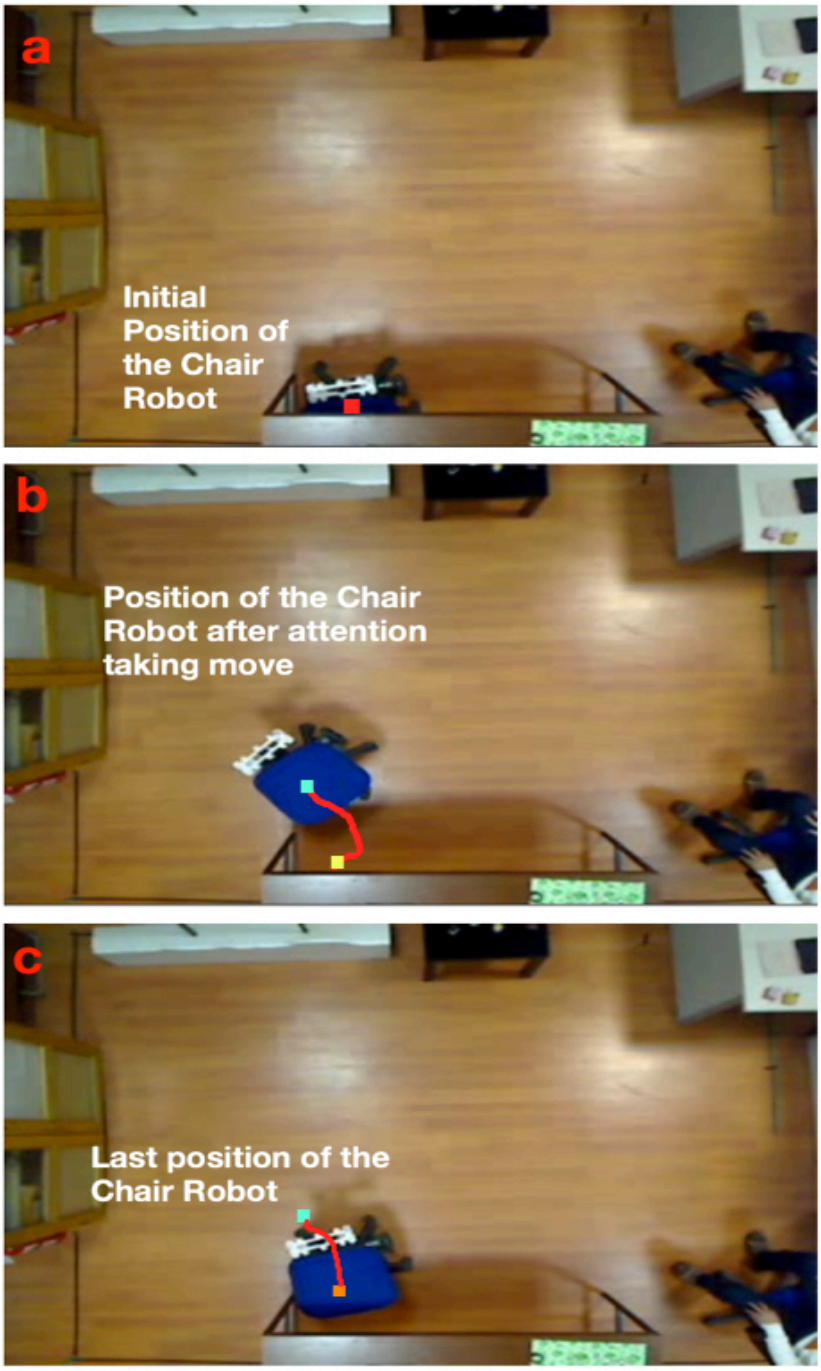

Figure 4-5 Movement of the chair robot. (a) Initial position of the chair robot just after experimenter revealed his current intention which is 'sitting'. (b) Chair robot is moving backward and show itself to the experimenter to gain his curiosity and to take attention of him. (c) After attention is taken chair robot move towards the computer and try to direct him to the computer.

After the chair robot has completed its movement and gave the message which is to sit on it and to work on computer, the experimenter is nevertheless waited to give a reaction to the chair movement. Figure 4-6 shows experimenter's movement to chair robot. Directing experimenter intention from 'sitting' to 'working on the computer' is the first successful step of the experiment which is generating a transient new current intention. After the experimenter sits on the chair and gets interested with the computer (Figure 4-6(d)), enough time is given to experimenter for this transient new current intention to become steady before another transient intention offered by the activation of the stair robot.

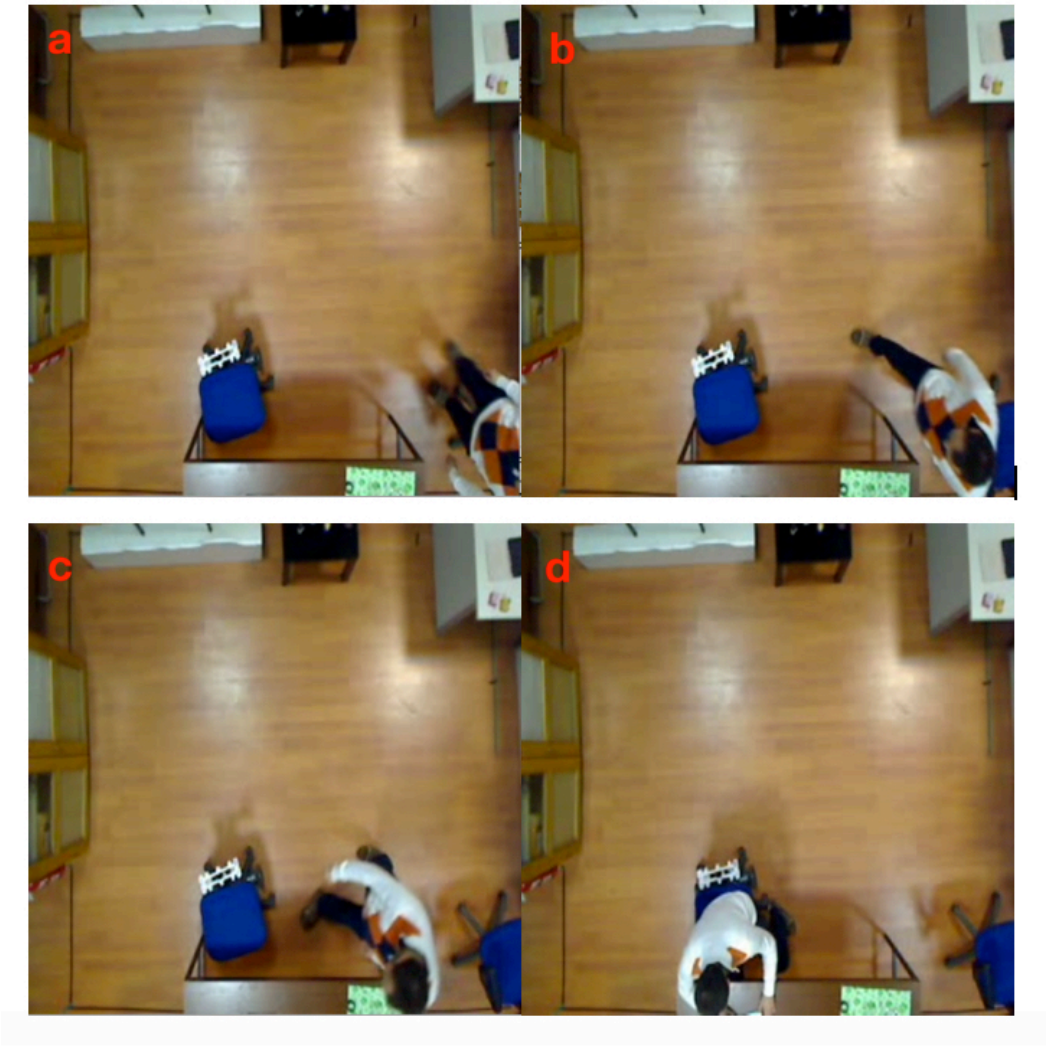

Figure 4-6 Experimenter movement steps after chair robot completes its movement. In (d) it is seen that experimenter intention is successfully changed to play with computer from sitting.

The initial position of the stair robot is out of the field of view of the ceiling camera. As chair robot is activated and moved towards the experimenter, the person's attention is deviated. The aim is to captivate the attention of the experimenter by the stair robot move towards the library to direct him to the library where his deep intention of reading a book can surface. In Figure 4-7, the stair robot movements are illustrated in (a) and (b). On the left side, stair robot position after it is moved from its initial position is shown. In (a) the experimenter curiosity can be seen from the fact that he stopped

working on the computer and starts to observe the movements of the stair robot. After the curiosity is gained, the stair robot is moved along the shown trajectory in (b) in order to focus the attention of the experimenter to the library. By moving towards the library, the stair robot induce the deep intention of experimenter towards taking a book from the library by stepping on it.

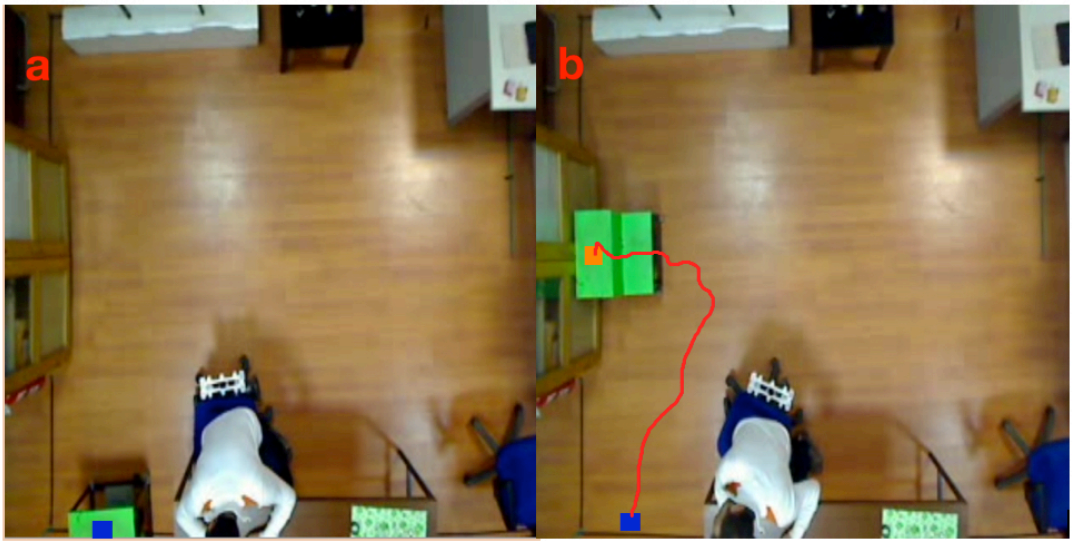

Figure 4-7 Stair robot movements. (a) Blue dot shows the position of the stair robot after it activated and moved inside the ROI of the ceiling camera to take attention of the experimenter. (b) Orange dot shows the position of the stair camera after it moves towards to library to surface the deep intention of reading book of experimenters. Red line shows robot trajectory towards the library.

In the Figure 4-8 the movements of the experimenter after the stair robot finishes its movement towards the library are illustrated. In (d) it is seen that experimenter is successfully directed towards the library in order to take a book. Hence his deep intention of reading book is surfaced by the robot moves in this smart environment.

Figure 4-8 Experimenters movement towards library and stair robot. In (d) experimenter is taking a book from the library.

4.2.2. Modified Environment with Captivating Tasks

In the previous section, one of the successful experiments in the simple environment is explained in detail. In order to analyze the system performance and conduct a sensitivity analysis, a captivity factor should be clearly incorporated in the tasks to be accomplished by the experimenters in the environment. To do so, an Ipad with piano learning application is placed on the table where music boxes are on and further more for case 3 some toys are placed next to the light on the small table. In the following sections one of the successful experiments and one failure are explained in detail.

Firstly, the failed experiment will be explained by analyzing experimenter's behavior and system behaviors selected by the neural network according to trajectory information of the experimenter. This case is selected to be included in the thesis since it motivated us to create a case 3 change in the attires of the smart environment. In Figure 4-9 from (a) to (d), experimenter's moves can be seen. After entering the smart environment, the person is directly moving to the music area which consists of music boxes and Ipad with a piano learning application. Observation of the consequent frames of (d) shows that his current intention is to play music on the Ipad.

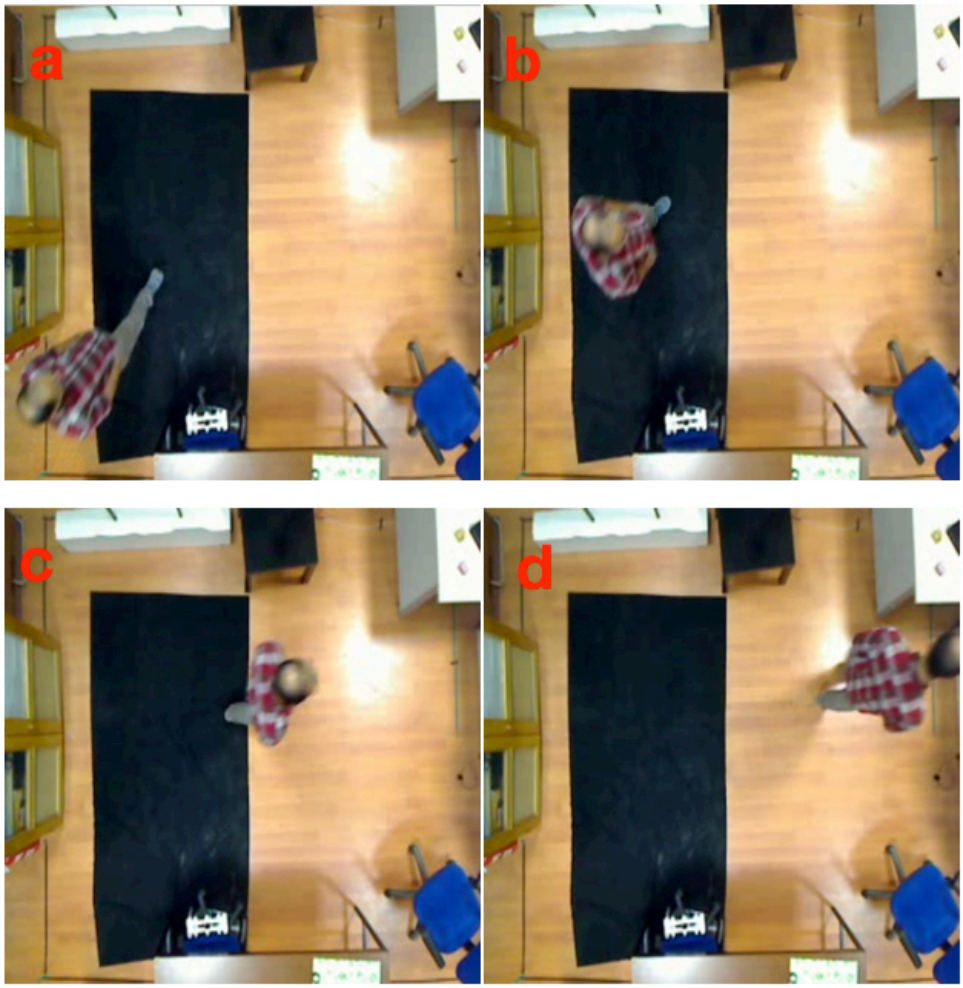

Figure 4-9 Experimenter's movement in the room. After entering the room, he is moving directly to the music area. His current intention is playing music on Ipad (d).

The experimenter's movement video is processed and his trajectory information is obtained and converted to the image data shown in Figure 4-10.

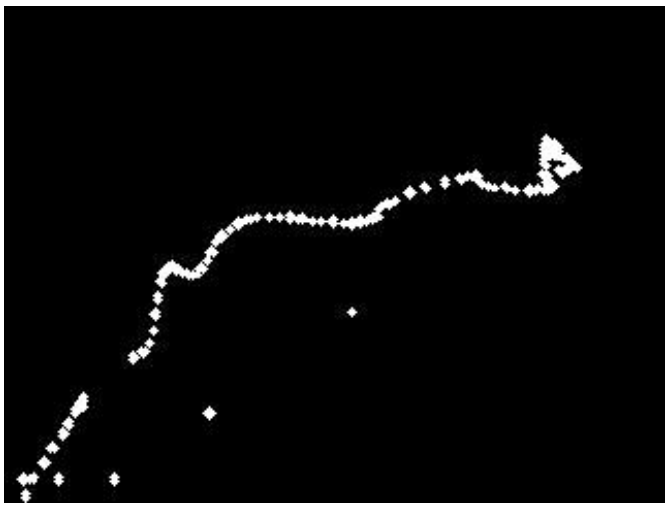

Figure 4-10 Trajectory information of the experimenters is converted to image data.

This extracted trajectory information is fed as input to the neural network that decides which system behaviors are appropriate for this experimenter in order to surface his deep intention of book reading. The output vector of the neural network for this particular case is:

$$\begin{bmatrix} Stair\ Robot \\ Chair\ Robot \\ Computer \\ Sound \\ Light \\ Coffee\ Machine \end{bmatrix} = \begin{bmatrix} 0.6669 \\ 0.1034 \\ 0.2251 \\ 0 \\ 0.0046 \\ 0 \end{bmatrix}$$

This output vector demonstrates of the weight values for the system behaviors candidates that will generate the next transient intention for the human. From above vector, the system concludes that the stair robot and chair robot + computer will be

activated in order to generation of the new current intentions with the priority of higher weight first.

Firstly, chair robot will be activated to direct experimenter to computer and make him be close to the library for the next step. In Figure 4-11 chair robot attention taking and directing to the computer movements are illustrated. In (a) chair robot's initial position is shown with yellow point. After experimenter current intention become steady, chair robot is activated to move to orange point shown in (b) with the trajectory in red line. Even if experimenter attention is taken to the chair robot, in the following frames experimenter continue to play Ipad while ignoring chair robot. Then as shown in (c), chair robot is repeated attention taking movement towards experimenter by closing him with trajectory in red line. Again experimenter ignore the movement of the chair robot and didn't interact with it to play computer on the table. In the next frames, chair robot returned its initial position shown in (d).

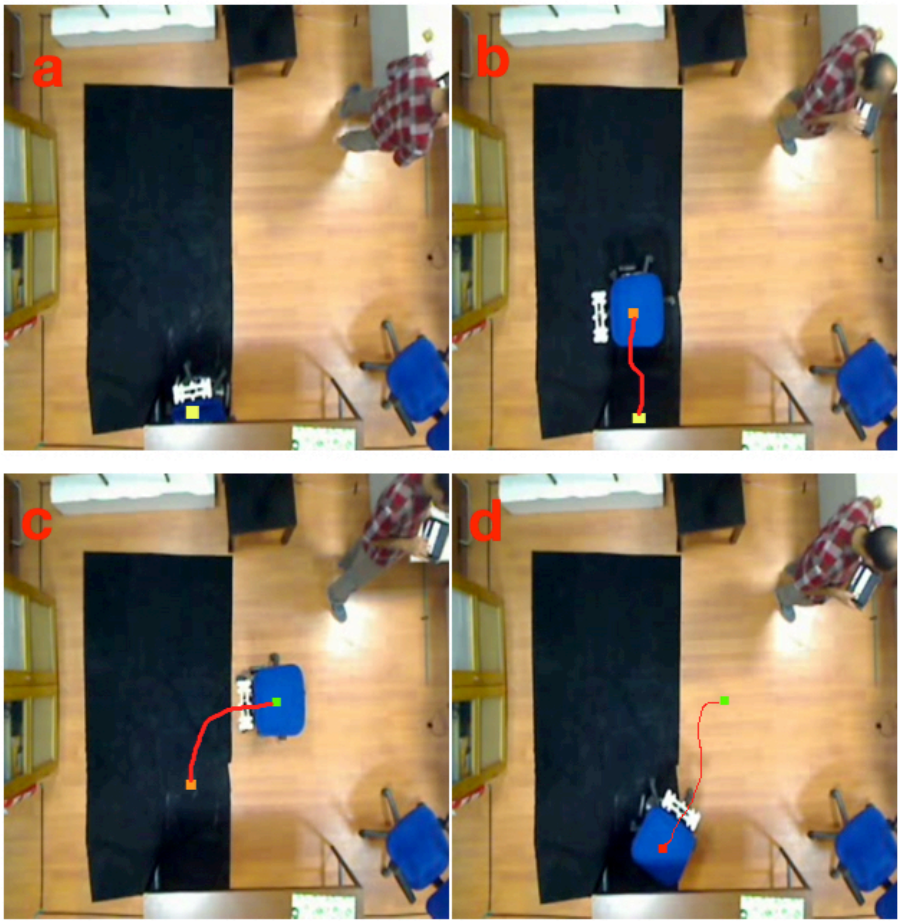

Figure 4-11 Chair robot movements. (a) Initial position of the chair robot. (b) Attention taking movement of the chair robot towards experimenter. (c) Because of the ignorance of the experimenter, attention taking movement is repeated by getting closer to experimenter. (d) Chair robot returns initial position.

After the chair robot completed its movement and fails to direct the experimenter to the table where computer is on, by interacting with him, the next system behavior which is stair robot is activated. The stair robot movements and experimenter reaction against these movements are shown in Figure 4-12. In (a) the stair robot position after it enters the field of the view of the ceiling camera is shown as a yellow point. The attention capturing move of the stair robot takes place from its initial position shown by a yellow dot to the position shown by an orange dot (a). The stair robot's attention capturing movement is ignored by the experimenter since he is strongly captivated by

the Ipad. Then the stair robot starts to move towards library to direct the experimenter to focus to library. As illustrated in (b), stair robot moved from the position shown by orange dot to position shown by blue dot along trajectory shown by red line. After this movement, the experimenter's attention is taken and he starts to move towards stair robot. However, he is still captivated by the Ipad and holds and plays with it while moving, as seen in (c) and (d). At the end of experimenter's movement, he is still playing with the Ipad and refuses to interact with the stair robot. Hence his deep intention of reading book could not be surfaced by the smart environment.

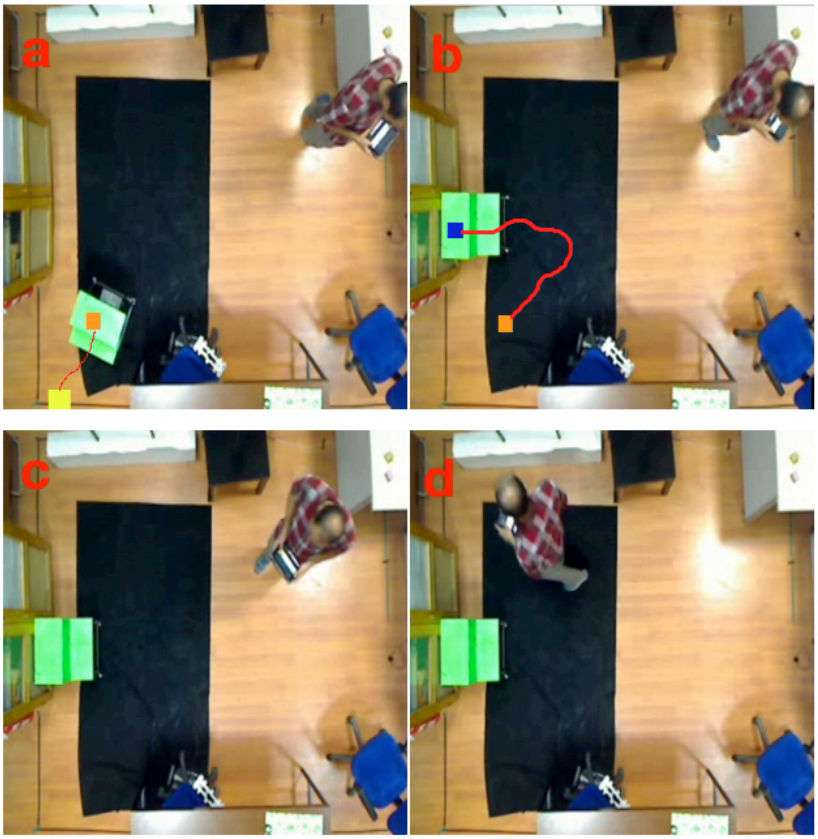

Figure 4-12 Stair robot movements. (a) and (b) Stair robot enters ROI and moves to take attention of the experimenter. (c) Experimenter becomes curious about stair robot however he is still captivated by Ipad. (d) Experimenter ignores stair robot and continues to play with Ipad.

This experiment shows that increasing the captivity of the environment may result in failure which is not being able to surface the deep intention of reading a book in the experiment. Hence in order to compensate this captivity factor in the environment, another change needs to be made by an attire of lower captivity than Ipad. In order to be able to break experimenter captivation, another captivating and interesting task of a lesser degree is introduced to the system which is adding different interactive toys to different place in the room. By that way the environmental smart behaviors will gradually break the constant attention of the experimenters on the captivating tasks. The purpose of this therefore to create an intermediate step in breaking obstinance of experimenter. Due to this changes, the training data and trained network is modified to increase the weight of the "light" for the cases where experimenter's current intention is "playing music". Investigating training data, reveals a fact that experimenters are most likely interacts with the system behavior which they pass near to or which stay in their sight area. Hence the modification on the learning system is applied for the cases experimenter trajectory data met this requirement. Some of the training data whose output vector is modified is shown in Figure 4-13. By using new training data, multi layer neural network designed as explained in previous chapter is trained again. For the remaining experiments, this modified network is used.

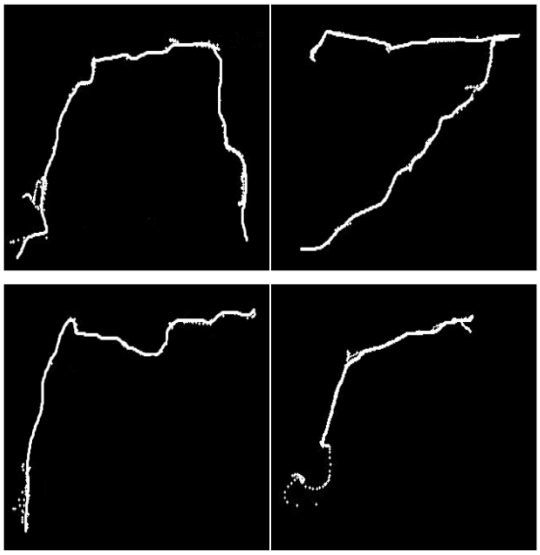

Figure 4-13 Example of training data whose output vector is modified to increase system behavior of “light”

With trained new network, following experiment is conducted for case 3 and results are explained in detail. In Figure 4-14(a,b,c,d,e) experimenter’s movement illustrated from “entering the room” to “reaching current intention of playing music” . As can be seen from Figure 4-14(c) and Figure 4-14(d), while experimenter is moving towards the table where music boxes are on, he is passing very close to the light where toys are placed . As explained above, it is expected by logic to activate the light to break the experimenter’s captivation to the music application on Ipad.

Figure 4-14 Movements of the experimenters while he is moving towards music area

The extracted trajectory of the experimenter is shown in Figure 4-15.

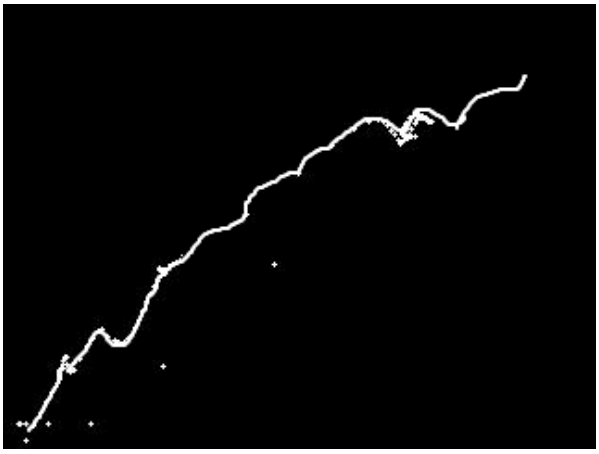

Figure 4-15 Extracted trajectory of the experimenter

This extracted trajectory information is fed as input to the neural network to get a decision on the system behaviors that will be appropriate in this case to generate the transient intentions for this experimenter that will lead to surfacing his deep intention of reading a book. The output vector of the neural network is:

$$\begin{bmatrix} Stair\ Robot \\ Chair\ Robot \\ Computer \\ Sound \\ Light \\ Coffee\ Machine \end{bmatrix} = \begin{bmatrix} 0.6068 \\ 0.1258 \\ 0.0170 \\ 0 \\ 0.2504 \\ 0 \end{bmatrix}$$

The output vector states that in order to surface experimenter's deep intention, system behaviors that should be activated are "light", "chair robot + computer" and "stair robot". "Light" is selected by the neural network because it is in the sight area of the experimenter's current position and the experimenter has passed near to it while he was walking towards the music area. In Figure 4-16 and Figure 4-17 illustrate activation of the system behaviors and experimenters interactions with them. As can

be seen in Figure 4-16 (a) and Figure 4-16(b), change in light's ON/OFF status takes attention of the experimenter. This fact is used to release experimenter from captivate effect of the Ipad application and direct him to another task which is playing with toys in this case. Figure 4-16(c) shows that experimenter's intention is successfully changed to playing with toys. After releasing him from the captivations of the music area, the chair robot is activated to capture his current attention and make him to get closer to the library. Chair robot's attention taking and message giving movements can be seen in Figure 4-16(d) and Figure 4-16(e). In Figure 4-16(d), the chair robot takes the attention of the experimenter, then directs him to the computer on the table. Experimenter accepts to interact with the chair robot and moves towards the computer and as shown in Figure 4-16(f) starts to work on the computer. This shows that generation of the first transient intention is done.

After new transient intention of working on the computer on the table is settled, the last system behavior which is stair robot is activated. In Figure 4-17 the movement of the stair robot and the interaction between the robot and experimenter are illustrated step by step. While experimenter is working on the computer, the stair robot tries to capture his attention by its movement in Figure 4-17(a) and (b). As expected, experimenter starts to investigate the stair robot and tries to understand why it is moving. After the experimenter's attention is taken, the stair robot starts to move towards the library as shown in Figure 4-17(c) and (d). By this movements robot induces in the experimenter's intention pushed deep into. After robot completes its movement, the experimenter climbs on stair robot and takes a book from the library as shown in Figure 4-17(e). This results in a successful experiment in which experimenter's deep intention of reading book is surfaced by the smart social environment.

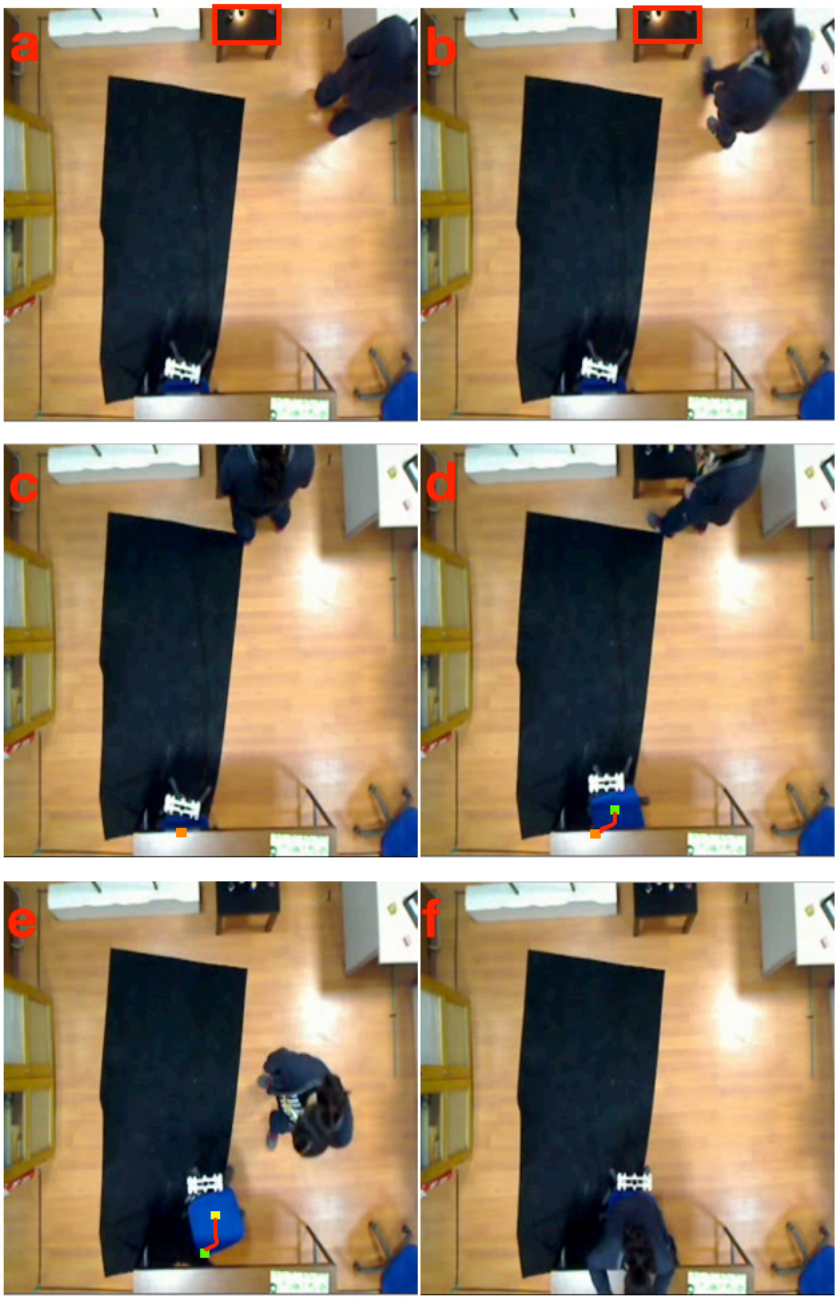

Figure 4-16 Activation of the system behaviors which are “light” and “chair robot” and interactions between this behaviors and experimenters

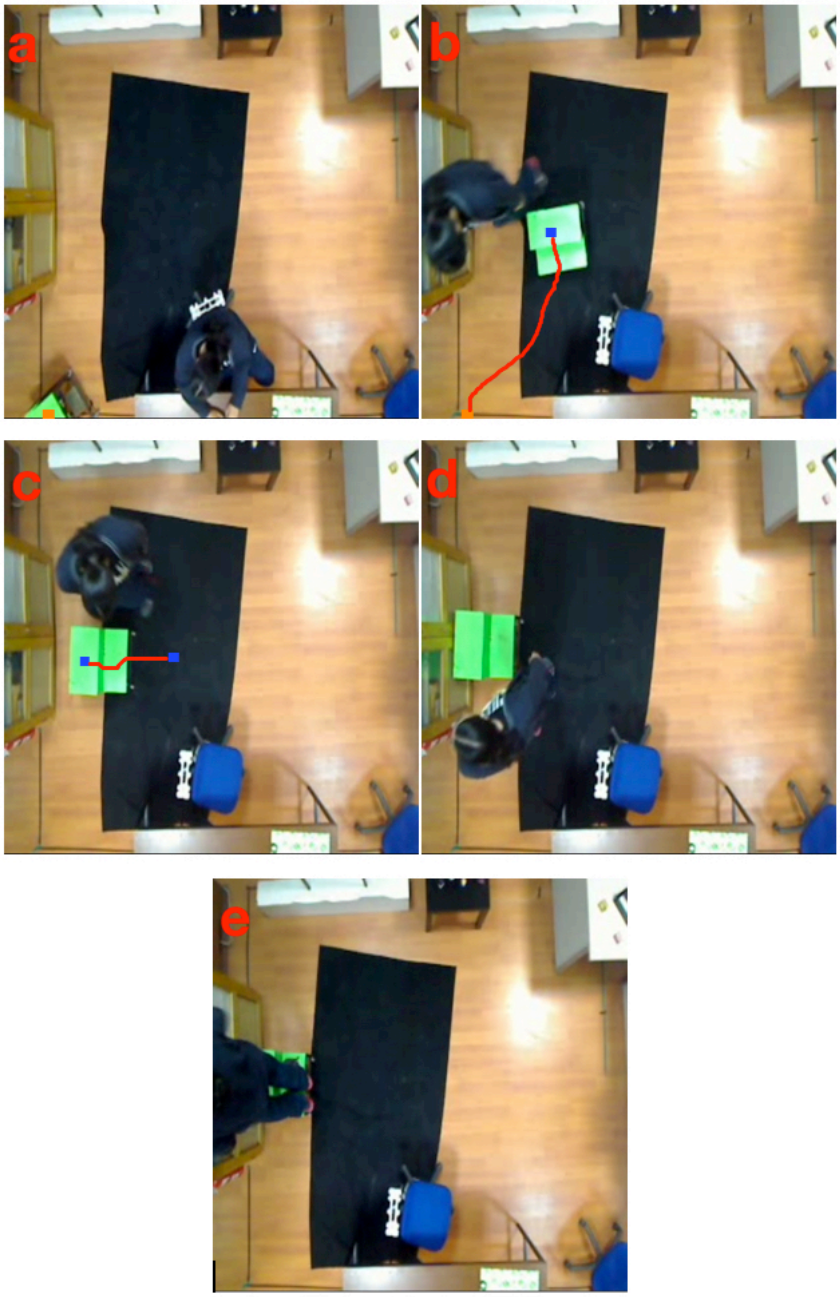

Figure 4-17 Movement of the stair robot to surface deep intention of the experimenter by directing him to library to take a book.

4.3. Sensitivity Analysis of the System According to System Parameters

After designing the test scenario of the study, the experiments are conducted for collecting both training and test data as explained in the previous sections. In order to able to analyze the system performance according to parameters, the relevant parameters are defined and changed during the experiments their effects on system performance are discussed. These parameters are : 1) the captivity level of the tasks and 2) the captivity time. The captivity level is defined as how much the tasks are interesting and addictive for an average type experimenter. Since increasing addictivity level of the task that can be done in the experiment room may results in difficulty to change current intention of the experimenter, changing captivity level parameter affect the experiments. The captivity level is measured by considering difficulty level of breaking the obstinance of experimenters while they are focused on particular task as a current intention. The captivity time is another important parameter which affect the difficulty of changing current intention of the experimenter as well. It is defined as how much time is given to experimenter to be spent within his current intention. Increasing the captivity level makes difficult to change one's intention since experimenter is connecting with the task more in time and becoming closer to any change.

4.3.1. Effects of the Captivity Level of the Tasks in the Environment

The captivity levels of the defined tasks to be executed in a simple environment are adjusted almost equal to each other. Then captivity levels of some task are changed to observe the effect to the experiments in terms of human-robot and human-environment interactions and performance. As a first step, an Ipad with a piano learning application, which has higher addictivity level than other tasks in the smart environment, is added to system. This is selected since it involves addictivity of following instruction to play a song from beginning to end. Hence, experimenter can play with the piano learning in an interactive manner instead of playing with music boxes, which may become a boring task after a while. After making this change, an experiment is conducted and

illustrated in Figure 4-18. As can be seen in Figure 4-18(c) and (d) experimenter is still under captivation of his current intention. This results in failure to surface his deep intention of book reading.

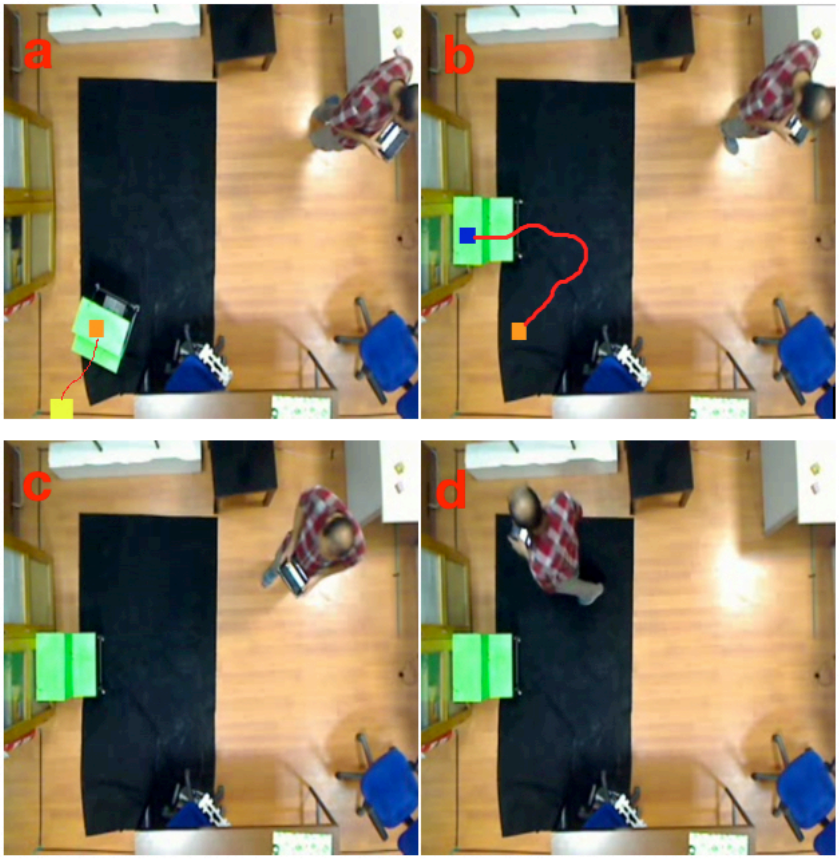

Figure 4-18 Results of increasing captivity level of the tasks in the environment. Despite of the changes in the environment, experimenter could not be released from captivation of the piano learning application.

In order to increase the success rate of the system in an environment that involve a task with higher captivity level, another task with a high captivity level is added. The motivation for making this change is that if the experimenter's current intention is one of this captivating task then another captivating task is used to break obstinance of the

experimenter within the deep focus of current intention. After directing experimenter from one captivating task to another, his intention status changes from steady to transient. This transient state increases the chance to change the experimenter's current intention to another task with lower captivity level.

The addition of toys is this new tool addition creating another task with high captivity level. The training data set is thus modified to include these additions:

1. For the cases where the experimenter's current intention is to play with piano learning application, "light" is added as a required system behavior to break the experimenter captivation with the current intention.
2. In the same manner, "sound" is added as a required system behavior to break the experimenter captivation from his current intention of playing with toys.

After training the neural network under modified data set from the modified environment, the changes in system behavior reflected by the newly trained network's decision can be seen in Figure 4-19. That is instead of directing the experimenter with chair and stair robot, the system behavior output uses the "light" to break the focus of the experimenters in playing piano application.

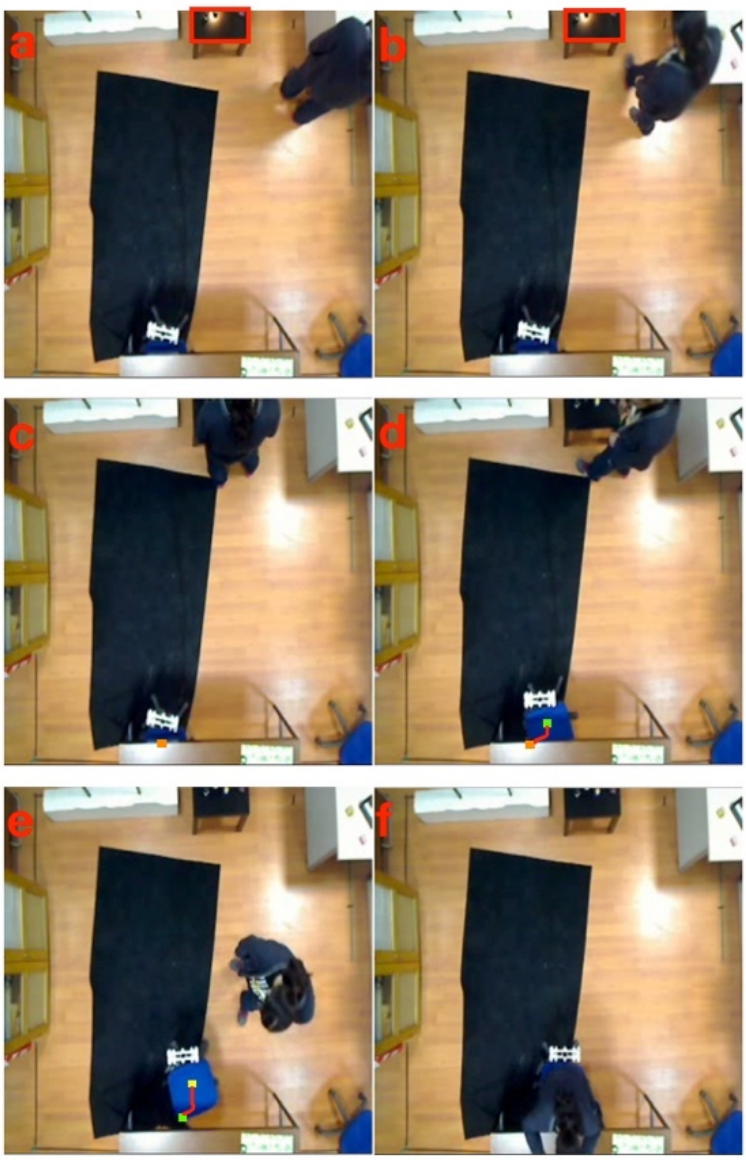

Figure 4-19 Usage of “light” as a middle step while directing to computer on the table

4.3.2. Effects of the Captivity Time

The captivity time is one of the most important parameter which directly affects the performance of the overall system. The captivity time is defined as the time given to experimenters at each step to be spent on their current intentions before activation of any system behavior. This directly affects level of experimenter's commitments on the current intention. If the experimenter's commitments to the task being actively executed is deep, the experimenter becomes resilient to changes in intentions or to interact with robots or the environment. Since our system tries to change experimenter's current intention step by step by getting him closer to the library, the increase in captivity time affects the success rate of the system. In order to analyze this situations, during experiments captivity time is increased systematically from 0.5 minute to 5 minute. Table 4-1 shows how captivity time is changed during the experiment procedures. Experimenter's initial intentions during the experiments are called current intentions and each captivation time is given to the experimenter, is showed in the corresponding column.

Table 4-1 Illustration of how captivation time is changed in the experiments

Experimenter	Current Intention	Captivation Time
Experimenter-1	Sitting on chair	30 sec
Experimenter-2	Sitting on chair	30 sec
Experimenter-3	Playing music	30 sec
Experimenter-4	Playing music	30 sec
Experimenter-5	Playing computer	1 minute
Experimenter-6	Sitting on chair	1 minute
Experimenter-7	Playing music	2 minute
Experimenter-8	Playing with toys	2 minute
Experimenter-9	Playing computer	3 minute
Experimenter-10	Playing computer	3 minute
Experimenter-11	Playing music	4 minute
Experimenter-12	Sitting on chair	5 minute

It is seen that as the captivity time increases, success rate of the overall system decreased. Especially if experimenter's current intention is "playing toys" or "playing music" and the captivity time is high, it is more likely to fail and end up with unchangeable intentions away from reading a book. This is because both captivity level and captivity time is inverse proportional with the success of the system which is surfacing experimenter's deep intention of reading a book. The change in the success rate of the experiments is shown in Figure 4-20. As can be seen from Figure 4-20 as the captivation time increases, the success of surfacing experimenter's deep intention decreases.

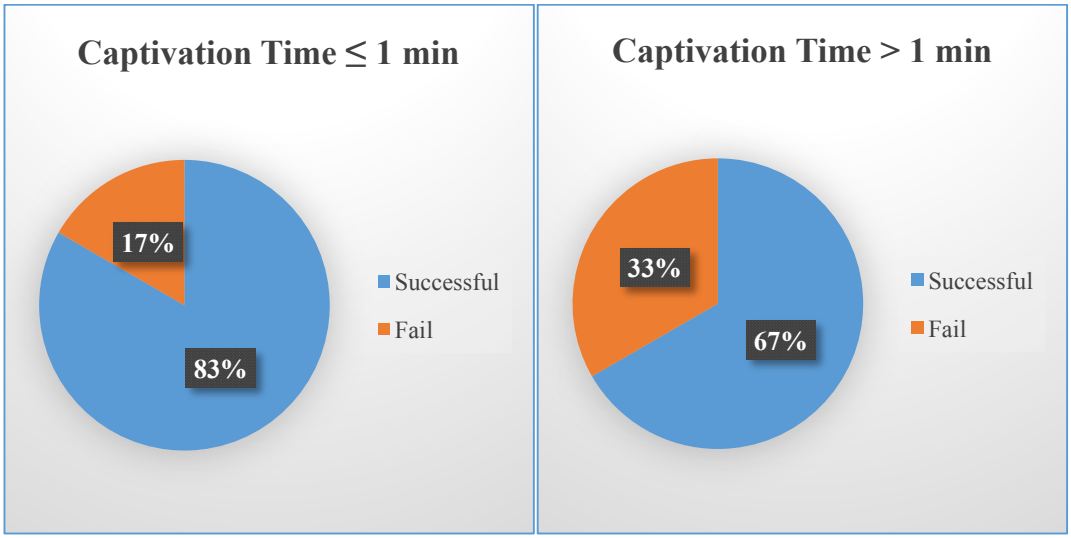

Figure 4-20 Change in success rate of the experiments with respect to change in captivation time

As mentioned above while captivity time increases, if experimenter's current intention is "playing music" it is more likely to fail changing this intention to the deep intention of reading a book. That is because that increase in both captivity level and captivity time. In order to illustrate this effect,
Figure 4-21 and Figure 4-22 can be seen. In

Figure 4-21, two experimenter's deep intentions of reading book are successfully surfaced while the captivity time is smaller or equal to 1 minute. However while captivity time increases, experimenter's commitment on playing music is getting so high that experimenter's intention could not be changed. (Figure 4-22)

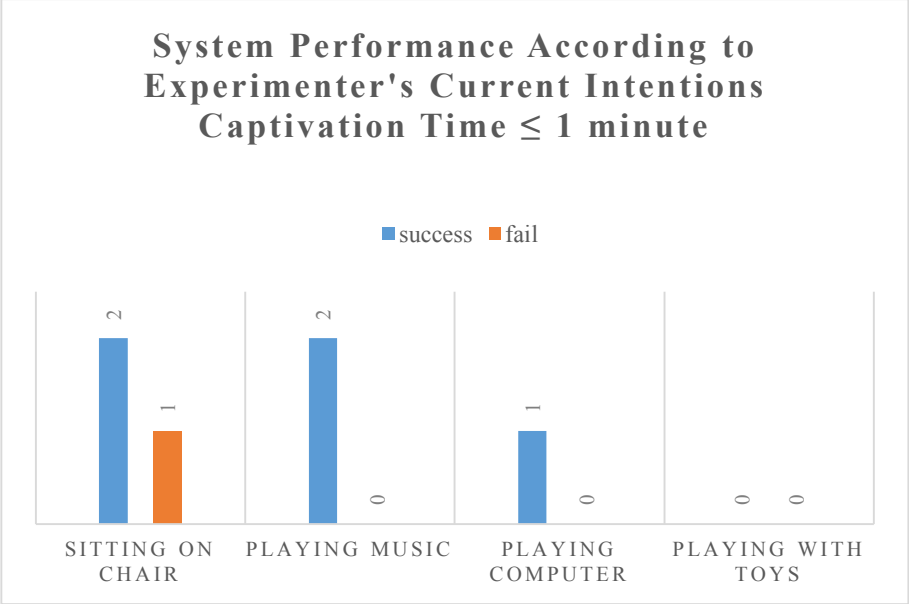

Figure 4-21 System performance analysis according to current intentions of experimenters when captivity time ≤ 1 minute.

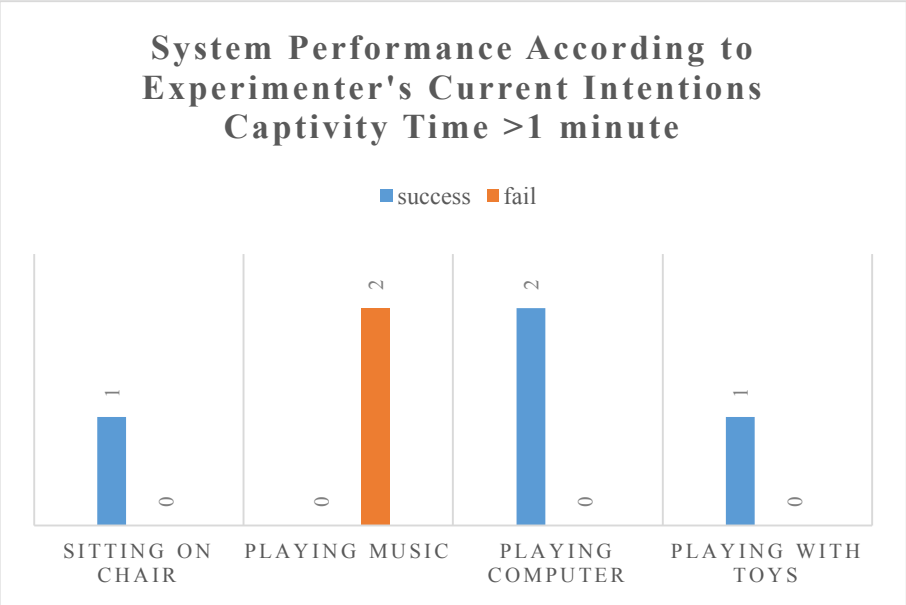

Figure 4-22 System performance analysis according to current intentions of experimenters when captivity time > 1 minute.

4.4. Performance Analysis

The main performance criterion of the overall system is the ability to the surface experimenter's deep intention of reading book in a specially designed smart environment with a skeletonized scenario in terms of the finite and limited number of specific tasks to be executed in the environment. The system analyzes the experimenter's trajectory information obtained by a ceiling camera and generates appropriate sets of system behaviors executed in order to change experimenter's intention from current to the desired one. In addition, the system needs to generate different sets of system behavior for different input trajectories. In order to test system performance, a series of experiments are conducted on 12 different people.

The distribution of the experimenter's current intentions is given in Figure 4-23. As can seen among 12 people, "sitting on chair" and "playing music" are selected the most frequent current intentions.

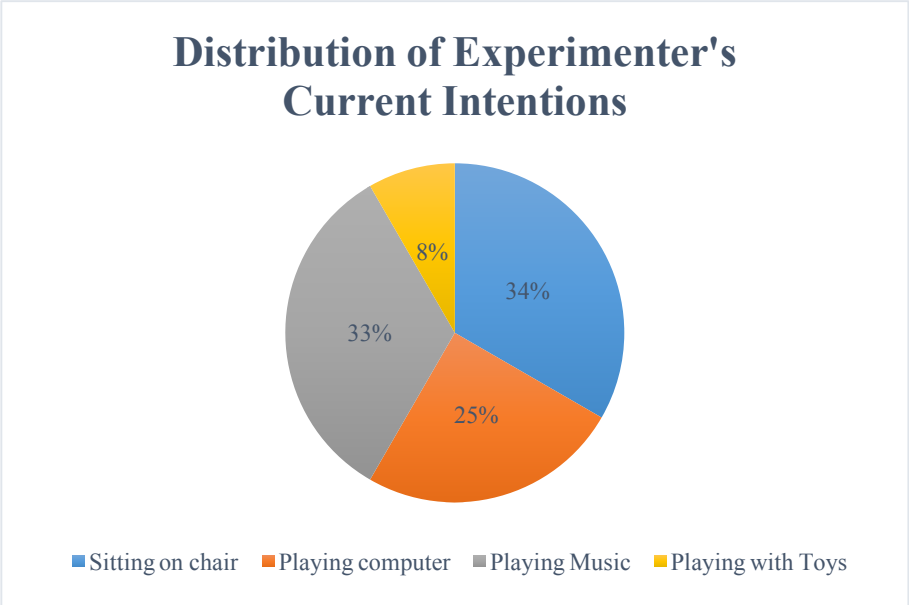

Figure 4-23 Distribution of experimenter's current intentions

According to these current intentions and the experimenters' trajectory information, the trained neural network selects appropriate system behaviors sets to surface the experimenters' deep intention of reading a book. The distribution of the selected system behaviors for different current intentions of the experimenters having different trajectories in the environment are illustrated in Figure 4-24.

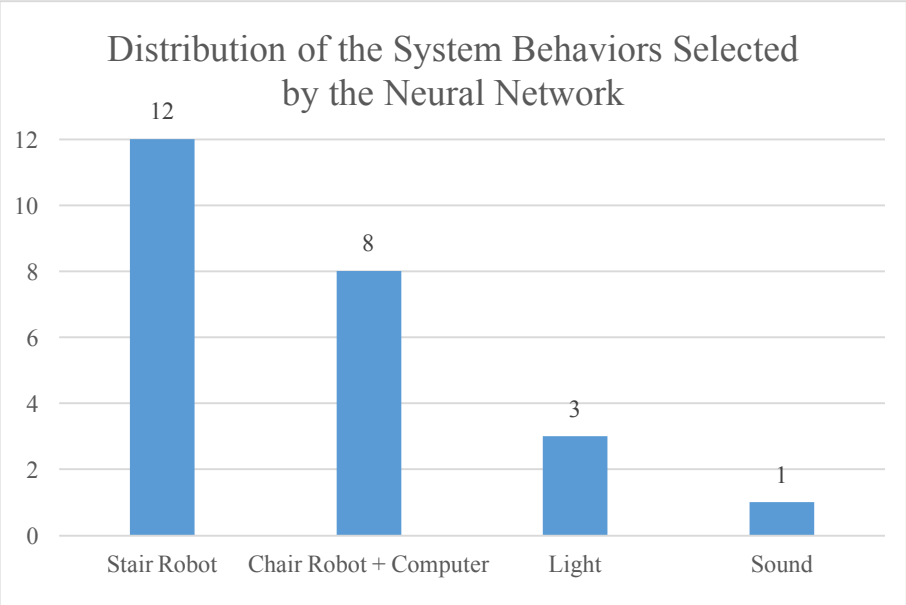

Figure 4-24 Distribution of the system behaviors selected by the neural network

The main performance criterion for this study is the percentage of the experimenters whom deep intention of reading a book is successfully surfaced. In Figure 4-25, this performance criterion is illustrated. As can be seen, system's success rate is %75.

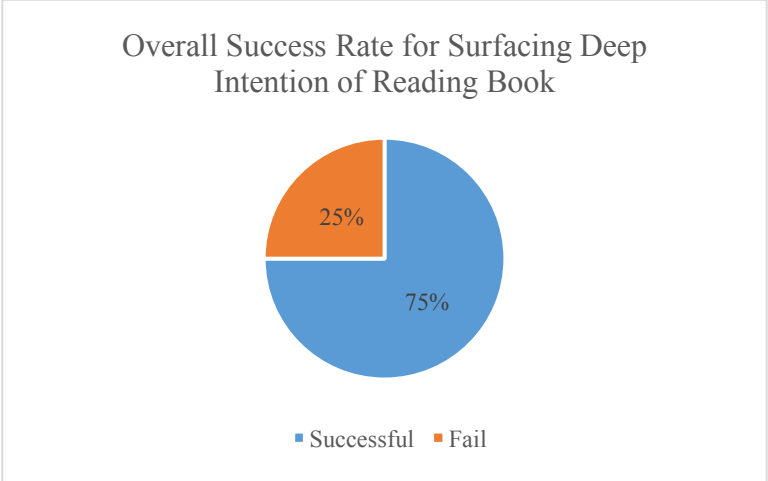

Figure 4-25 Overall Success Rate for Surfacing Deep Intention of Reading Book

Experimenter's current intention affect critically the performance of the system. Since all tasks have different level of captivity, which is defined as how much the task is interesting and addictive for an average type experimenter, the effects of the system behaviors on the experimenters vary. Also as Gorur et al. states in his study, according to the personality of the experimenter the human-robot or human-environment interaction can be easier or more difficult. Due to same cause experimenter's current intentions may vary during the experiment. Hence some people can easily interact with the environment while some prefers minimum interactions. This leads to the conclusion that the same system behaviors can affect differently the experimenter's according to their current intentions and personalities. In Figure 4-26, the relation between the experimenter's current intentions and system success is given. It is seen that failing to surface deep intention occurred when the experimenters' current intentions are "sitting on chair" and "playing music". As explained in sensitivity analysis part, "playing music" has greater captivity level among all tasks. Hence this level explains why the system fails with %50 percent when current intention is "playing music". Also as studied in Gorur's previous work, some people can behave suspiciously in the experiment room. This results in minimum interaction with both robots and environment. "Sitting on chair" requires minimum interaction among other task. Hence, failure in the experiment where the experimenter's current intention is "sitting on chair" becomes understandable when the experimenter's suspicion towards robots and environment is clearly present.

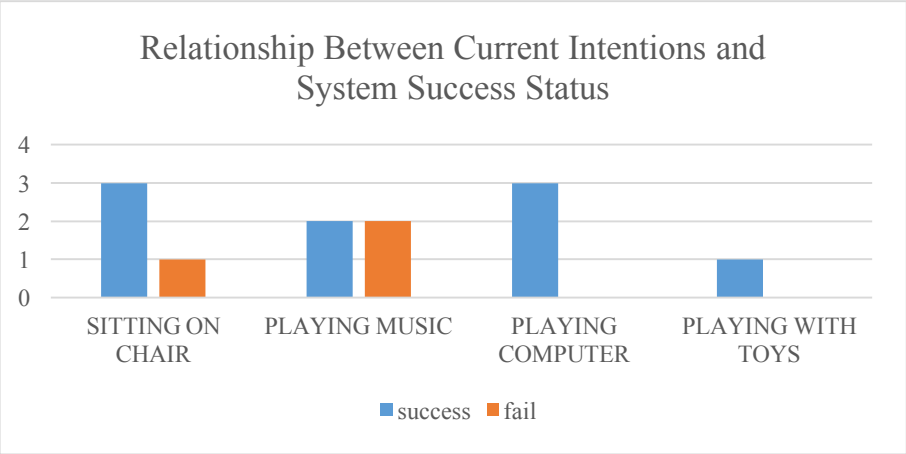

Figure 4-26 Relationship Between Current Intentions and System Success Status

CHAPTER 5

CONCLUSION AND FUTURE WORKS

In this thesis study, a smart social environment is designed to surface human's deep intention of reading book as a proof of concept study. The smart environment consists of robots (stair and chair) and manipulating components (light and sound) to direct the human towards the desired intention. These are called system behaviors and a subset of system behaviors is selected special to each experimenter according to their trajectory information. Movement of the experimenters in the smart experiment room is recorded by ceiling camera as video data and processed to extract trajectory information. This extracted trajectory information is converted into a 240*320-pixels image data to represent the real localization in the room. In order to find a correlation between trajectory information and system behaviors, a multi-layer neural network with stacked autoencoders is designed. Usage of stacked autoencoders is important for two main reasons which are: 1) reducing dimension by meaningful and 2) compressed representation of input and extracting feature vectors of input without handcrafting methods. GLUT is preferred to initialize network parameters in order to avoid to be get stuck at a local minima or plateaus during the training phase. As discussed in 2.2.1 both autoencoders and GLUT increase network performance in terms of training time and correct classification rate. In addition, since system inputs are 240x320-pixels image data which correspondence 1x76800-vector representation, it is crucial to reduce the dimension of the input with minimum loss. Otherwise, training time increases exponentially as input size increases. As explained in Chapter 3 and Chapter 4, this network is trained with training data set collected from the real time experiment conducted on 15 different people. Then, as explained in Chapter 4, real time tests are conducted on 12 different people to observe overall system performance with observation phase and decision-making phase. The results show that according to the trajectory information, appropriate set of system behavior can mine human's deep intention of reading book.

The motivation behind this study is to design a smart social environment which can track human behavior and learn their daily routines. This system can be used to increase the quality of the human daily life such as the environment can be programmed as that it can surface human deep intention of reading a book or doing sport every day. Furthermore, the system can be used as assistants for the people in need of help (such as elderly people) understanding their needs and guiding them accordingly. This system can be used to remind the activities or things to be held which can be forgotten or pushed deeper as intention by elderly people.

In addition to completed studies in this thesis work, the system can be improved in many ways. For example, in addition to trajectory information users' gestures, mimics and body movements can be investigated as well in the system and used for classification. These variables can inform the system for better understanding of human psychology. In addition, it is important to analyze confidence level between human and robots or human and environment. This affects human's response to robot activities and collaboration level between human and robots. In addition, larger training data set increases the quality of the selection process of the system. Hence, reliability of the neural network performance can be increased by conducting experiments with many more people.

REFERENCES

Ali, N., Khan, N. Y., & Imran, A. S. (2007). Controlling Mouse through Eyes, 179–183. doi:10.1109/ICET.2007.4516339

Awais, M., & Henrich, D. (2013). Human-robot interaction in an unknown human intention scenario. *Proceedings - 11th International Conference on Frontiers of Information Technology, FIT 2013*, 89–94. doi:10.1109/FIT.2013.24

Baldi, P. (2012). Autoencoders, Unsupervised Learning, and Deep Architectures. *ICML Unsupervised and Transfer Learning*, 37–50.

Bengio, Y. (2009). Learning Deep Architectures for AI. *Foundations and Trends® in Machine Learning*, *2*(1), 1–127. doi:10.1561/2200000006

Bengio, Y., & {LeCun}, Y. (2007). Scaling Learning Algorithms towards AI. *Large Scale Kernel Machines*, (1), 321–360.

Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(8), 1798–1828. doi:10.1109/TPAMI.2013.50

Conference, I. I., & Processing, S. (2014). Exploring One Pass Learning For Deep Neural Network Training With Averaged Stochastic Gradient Descent Zhao You , Xiaorui Wang , Bo Xu Institute of Automation , Chinese Academy of Sciences , 6904–6908.

Deng, L., Hinton, G., & Kingsbury, B. (2013). New types of deep neural network learning for speech recognition and related applications: an overview. *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 8599–8603. doi:10.1109/ICASSP.2013.6639344

Durdu, A., Erkmen, I., & Erkmen, A. M. (2012). Observable operator models for reshaping estimated human intention by robot moves in human-robot interactions. *Innovations in Intelligent Systems and Applications (INISTA), 2012 International Symposium on*, 1–5. doi:10.1109/INISTA.2012.6247009

Erden, M. S., & Tomiyama, T. (2010). Human-intent detection and physically interactive control of a robot without force sensors. *IEEE Transactions on Robotics*, *26*(2), 370–382. doi:10.1109/TRO.2010.2040202

Erhan, D., Manzagol, P.-A., Bengio, Y., Bengio, S., & Vincent, P. (2009). The difficulty of training deep architectures and the effect of unsupervised pre-training. *International Conference on Artificial Intelligence and Statistics*, *5*, 153–160. Retrieved from http://machinelearning.wustl.edu/mlpapers/paper_files/AISTATS09_ErhanMBBV.pdf

Görür, O. C., & Erkmen, A. M. (2014). Elastic networks in reshaping human intentions by proactive social robot moves. *RO-MAN'14 The 23rd IEEE International Symposium on Robot and Human Interactive Communication*, 1012–1017. doi:10.1109/ROMAN.2014.6926385

Han, J. H., & Kim, J. H. (2010). Human-robot interaction by reading human intention based on mirror-neuron system. *2010 IEEE International Conference on Robotics and Biomimetics, ROBIO 2010*, 561–566. doi:10.1109/ROBIO.2010.5723387

Hinton, G. E. (2007). Learning multiple layers of representation. *Trends in Cognitive Sciences*, *11*(10), 428–434. doi:10.1016/j.tics.2007.09.004

Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, *18*(7), 1527–54. doi:10.1162/neco.2006.18.7.1527

Holden, a J., Robbins, D. J., Stewart, W. J., Smith, D. R., Schultz, S., Wegener, M., … Moloney, J. V. (2006). Reducing the Dimensionality of Data with Neural Networks, *313*(July), 504–507. doi:10.1126/science.1127647

Hwang, B., Jang, Y. M., Mallipeddi, R., & Lee, M. (2012). Probabilistic human intention modeling for cognitive augmentation. *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, 2580–2584. doi:10.1109/ICSMC.2012.6378134

Jenkins, O. C., Serrano, G. G., & Loper, M. M. (2007). Interactive human pose and action recognition using dynamical motion primitives. *International Journal of Humanoid Robotics*, *04*(02), 365–385. doi:10.1142/S0219843607001060

Ji, S., Yang, M., Yu, K., & Xu, W. (2013). 3D convolutional neural networks for human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(1), 221–31. doi:10.1109/TPAMI.2012.59

Kelley, R. (2008). Understanding Human Intentions via Hidden Markov Models in Autonomous Mobile Robots, 367–374.

Koo, S., & Kwon, D. (2009). Recognizing Human Intentional Actions from the Relative Movements between Human and Robot. In *The 18th IEEE*

International Symposium on Robot and Human Interactive Communication, 2009. RO-MAN 2009. (pp. 939–944).

Larochelle, H., Larochelle, H., Bengio, Y., Bengio, Y., Lourador, J., Lourador, J., … Lamblin, P. (2009). Exploring Strategies for Training Deep Neural Networks. *Journal of Machine Learning Research*, *10*, 1–40.

Rivera-Bautista, J. A., Ramirez-Hernandez, A. C., Garcia-Vega, V. A., & Marin-Hernandez, A. (2010). Modular control for human motion analysis and classification in Human-Robot interaction. *Human-Robot Interaction (HRI), 2010 5th ACM/IEEE International Conference on*, 169–170. doi:10.1109/HRI.2010.5453210

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning internal representations by error propagation. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*.

Salakhutdinov, R., & Salakhutdinov, R. (2009). Learning Deep Generative Models. *Mit.Edu*, 1–84. Retrieved from http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Learning+deep+generative+models#0

Sato, E., Yamaguchi, T., & Harashima, F. (2007). Natural interface using pointing behavior for human-robot gestural interaction. *IEEE Transactions on Industrial Electronics*, *54*(2), 1105–1112. doi:10.1109/TIE.2007.892728

Schwarz, U. (2002). Cognitive eyes, 175–179.

Tracking Human Motion and Actions for Interactive Robots Odest Chadwicke Jenkins Matthew Maverick Loper X , Motor Primitives and Imitation Learning Ci L ¢ i (X , X , X) i = l. (n.d.), 365–372.

Valle, E. A., & Starostenko, O. (2013). Recognition of Human Walking / Running Actions Based on Neural Network. *2013 10Th International Conference on Electrical Engineering, Computing Science and Automatic Control (Cce)*, 239–244.

Weston, J. (2008). Deep Learning Via Semi Supervised Embedding. doi:10.1145/1390156.1390303

Yokoyama, A., & Omori, T. (2010). Modeling of human intention estimation process in social interaction scene. *International Conference on Fuzzy Systems*, 1–6. doi:10.1109/FUZZY.2010.5584042