

University of Arkansas, Fayetteville
ScholarWorks@UARK

Computer Science and Computer Engineering
Undergraduate Honors Theses

Computer Science and Computer Engineering


12-2011

Object recognition based on shape and function

Akihiro Eguchi

University of Arkansas, Fayetteville

Follow this and additional works at: <http://scholarworks.uark.edu/csceuh>

 Part of the [Computer Sciences Commons](#), and the [Developmental Psychology Commons](#)

Recommended Citation

Eguchi, Akihiro, "Object recognition based on shape and function" (2011). *Computer Science and Computer Engineering Undergraduate Honors Theses*. 1.

<http://scholarworks.uark.edu/csceuh/1>

This Thesis is brought to you for free and open access by the Computer Science and Computer Engineering at ScholarWorks@UARK. It has been accepted for inclusion in Computer Science and Computer Engineering Undergraduate Honors Theses by an authorized administrator of ScholarWorks@UARK. For more information, please contact scholar@uark.edu.

OBJECT RECOGNITION BASED ON SHAPE AND FUNCTION

OBJECT RECOGNITION BASED ON SHAPE AND FUNCTION

A thesis submitted in partial
fulfillment of the requirements for the degree of
Bachelor of Science

By

Akihiro Eguchi

December 2011

ABSTRACT

This thesis explores a new approach to computational object recognition by borrowing an idea from child language acquisition studies in developmental psychology. Whereas previous image recognition research used shape to recognize and label a target object, the model proposed in this thesis also uses the function of the object resulting in a more accurate recognition. This thesis makes use of new gaming technology, Microsoft's Kinect, in implementing the proposed new object recognition model. A demonstration of the model developed in this project properly infers different names for similarly shaped objects and the same name for differently shaped objects.

This thesis is approved for recommendation
to the Graduate Council.

Thesis Director:

Craig Thompson, Ph.D.

Thesis Committee:

Russell Deaton, Ph.D.

John Gauch, Ph.D.

THESIS DUPLICATION RELEASE

I hereby authorize the University of Arkansas Libraries to duplicate this thesis when needed for research and/or scholarship.

Agreed _____

Akihiro Eguchi

Refused _____

ACKNOWLEDGEMENTS

I thank Dr. Craig Thompson for his kindness and support in all our interactions. Without his guidance and encouragement, my time as an undergraduate student would have been far less exciting. In addition, I thank Dr. Russell Deaton and Dr. John Gauch for serving on my committee.

I am grateful to my parents, Koichi Eguchi and Yoshiko Eguchi, for always caring about me from overseas and for making possible my study in the United States.

TABLE OF CONTENTS

1. Introduction.....	1
1.1 Problem.....	1
1.2 Objective.....	2
1.3 Approach.....	2
1.4 Organization of this Thesis	3
2. Background	4
2.1 Key Concepts – Machine Learning and the Kinect.....	4
2.1.1 Machine Learning	4
2.1.2 Microsoft Kinect for the Xbox 360.....	5
2.2 Related Work	6
2.2.1 Study in Developmental Psychology	7
2.2.2 Object Recognition in Computer Science.....	7
2.2.3 Activity Recognition.....	8
3. Approach and Architecture	10
3.1 Machine Learning Techniques and the Kinect SDK	10
3.1.1 Selecting a Learning Technique.....	10
3.1.2 Learning to Use the Kinect SDK	12
3.2 Architecture of the Object Recognition Model.....	16
3.2.1 Plain Surface Removal with RANSAC Algorithm.....	17
3.2.2 K-Nearest Neighbor Algorithm for Shape Learning	19
3.2.3 Activity Recognition using the Kinect.....	19

3.2.4 Object Recognition Model with Shape Bias and Function Bias.....	21
4. Methodology, Results and Analysis.....	23
4.1 Methodology.....	23
4.2 Results.....	23
4.1.1 Object Recognition based on Shape	23
4.2.2 Object Recognition based on Function	25
4.2.3 Object Recognition based on both Shape and Function	28
4.3 Analysis	28
5. Conclusions.....	30
5.1 Summary.....	30
5.2 Potential Impact	30
5.3 Future Work.....	31
References.....	32

LIST OF FIGURES

Figure 1: Grid structure for the hand-written number recognition	10
Figure 2: Neural network for the hand-written number recognition.....	11
Figure 3: Depth image retrieved from Kinect.....	13
Figure 4: Noise reduction by horizontal interpolation.....	14
Figure 5: Set a tolerance of the gap between points to draw a smoother image.....	15
Figure 6: 3D image using two Kinects	16
Figure 7: Room layout	16
Figure 8: Use case for the object recognition model	17
Figure 9: RANSAC for surface detection	18
Figure 10: Naming snack based on the shape	19
Figure 11: Skeleton on the 3D image and body joint tracking	20
Figure 12: Choosing name of object corresponding to the action	21
Figure 13: Shape learning of insecticide, chair, and antiperspirant	24
Figure 14: Different shape of a chair	25
Figure 15: Sitting on a chair.....	26
Figure 16: Killing bugs with insecticide.....	26
Figure 17: Deodorizing with an antiperspirant	27
Figure 18: Testing the action sitting on a different chair.....	27
Figure 19: Case where the object or action cannot be recognized properly	29

1. INTRODUCTION

1.1 Problem

Object recognition is a subfield of machine vision and artificial intelligence. Earlier object recognition research required significant knowledge of the mathematics of image processing and expensive equipment like LIDAR cameras or SICK sensors. The recent introduction by Microsoft of the Kinect for the Xbox 360 has changed the landscape and enabled many researchers to tackle a range of image recognition problems without such extensive knowledge or expensive equipment. Most research on developing object recognition models has been based only on the shape of the objects. Therefore, there has been a difficulty in recognizing objects like a uniquely designed chair or different objects that have similar shapes.

In order to solve this problem, this thesis focuses on a strategy that human children use to solve the problem. In the field of psychology, many studies have focused on language acquisition by children. Results indicate that there exist two strong biases children use when they try to learn names of objects: shape bias and function bias. Most past studies of object recognition in computer science have focused on shape bias; only a few studies have focused on the function of objects to recognize objects. In order to learn the name of object like a chair, if computer can learn the name not only based on its shape but also based on the functionality, which is a place to sit, then the program can perform more flexible object recognition in a manner more similar to humans.

1.2 Objective

The objective of this research is to combine the Kinect sensor with machine learning techniques to implement an object recognition model that uses both shape bias and function bias to learn the names of objects in a manner similar to how human children acquire names of objects.

1.3 Approach

The first step of this research includes becoming familiar with machine learning techniques and with the Microsoft Kinect SDK. To understand the basics of machine learning techniques and to select one for use in this project, the author wrote a program to recognize a hand-written number using a neural network program and a K-nearest neighbor algorithm. Based on comparing the results from those two techniques, the K-nearest neighbor algorithm was selected based on its speed and simplicity. To become familiar with Kinect SDK, the author drew a 3D model of the environment with OpenGL by drawing polygon based on the point clouds retrieved from the Kinect sensor. When it was determined that there was considerable noise in the raw data, a noise reduction by horizontal interpolation technique was applied. Also, since the Microsoft SDK supports only one Kinect sensor to be used in a program, the author developed a program to use two different Kinects as clients to a shared server.

The second step was to implement a *shape bias* capability. In order to implement shape bias, the first step is to remove the plain surface where the object will be placed in the depth map retrieved from the Kinect sensor. This allows the program to only have to

learn the shape of the target object. Then, the K-nearest neighbor algorithm is used to recognize the object based on its shape.

The third step was to implement a *function bias* capability. In order to recognize the functionality of a target object, a program was written to recognize human activity, in this case walking, skipping, and running in place. This was accomplished by first recording ten seconds of the movement as sets of xyz coordinates of twenty different human body joints as retrieved from the Kinect sensor. Then a K-nearest neighbor algorithm was executed to make the program learn the activity. Instead of using absolute distance from the Kinect sensor, by using relative distance of each joint from the head position, the program recognizes the correct activity even if the actor is facing a different direction. This activity recognition technique for the learning of the use of objects enabled implementing the function bias.

The final step was to combine the techniques of shape bias and the function bias so that the program can now recognize objects in a manner similar to the way human children learn names of objects.

1.4 Organization of this Thesis

Chapter 2 provides background on key concepts needed to understand the rest of the thesis including two machine learning techniques, the Kinect sensor, and existing approaches for object recognition. Chapter 3 describes the architecture of the object recognition model developed in this thesis. Chapter 4 provides a methodology, results, and analysis, describing activity recognition and also object recognition based on shape and function. Chapter 5 summarizes the thesis, its potential impact and identifies areas for future work.

2. BACKGROUND

Section 2.1 of this chapter describes key concepts that the reader will need to understand the research reported in chapters 3 and 4. Section 2.2 describes related work in developmental psychology, object recognition, and activity recognition.

2.1 Key Concepts – Machine Learning and the Kinect

Key concepts in this research include a machine learning technique for object recognition and the use of the gaming technology Kinect for Xbox 360.

2.1.1 Machine Learning

Machine learning is an important subfield of artificial intelligence aimed at giving computers a way to learn many kinds of things without explicitly being programmed. Examples of machine learning occur in domains like autonomous vehicles, checker playing, and signal processing. Machine learning is also commonly used in the image processing area, especially for recognition of objects, which includes the human face and everyday objects.

Machine learning is based on training, and there are two types: supervised learning and unsupervised learning. In supervised learning, training data always gives the right answer y to the corresponding input x so that the program determines a pattern to predict the function $f(x) = y$. For unsupervised learning, a training set does not contain the output y , but instead, the computer must figure out the hidden structure of the data. This thesis uses supervised learning because the names of objects will be explicitly provided by the trainer.

A primitive model of machine learning based on experience is called rote learning. In this model, the program stores all input x and associated output y in the memory. This is very simple algorithm, but this type of learning only works for small discrete number of possible input and output. Another approach is decision tree induction which can be used in an algorithm like ID3 (Iterative Dichotomiser 3) by storing an attribute vector x with a corresponding output y . However, overfitting is a problem in this technique. Even if the model can perfectly predict an output y from an input x for a training set of data, it does not guarantee that the model is perfect for other data as well.

One way to deal with the problem of overfitting is to use neural networks. In this approach, all Boolean values of input vector x will be nodes in the input layer of the network and the corresponding output y will be at the output layer. There can be several layers of nodes between the input layer and the output layer, and, with many iterations, the algorithm figures out a strong association of each node in the network. This is generalizable technique, but the problem is that if the number of nodes in the training set becomes large, then the time to build the network will significantly increase.

Another approach to dealing with the overfitting problem is a K-nearest neighbor algorithm [1]. Like rote learning, it stores all set of $\langle x, y \rangle$ in the memory, but in the testing phase, the algorithm takes k $\langle x, y \rangle$ vectors from the memory of which the x value is most similar to the target input x . Then, the algorithm determines the predicted output y by taking majority vote.

2.1.2 Microsoft Kinect for the Xbox 360

The Kinect is a sensing technology originally created for a controller for the Microsoft gaming console Xbox 360 and was released to the public in November 2010.

The features of this sensor include dynamic depth image retrieval, human body recognition, skeletal joint tracking, and a multi-array microphone, all at a cost of around \$150. Because of the high versatility and the low cost of the sensor, many people have experimented with using the Kinect in their projects for various ways. Even before the Kinect SDK was officially released in June 2011, Kinect videos were already appearing on Internet video streaming websites like YouTube.

This project depends directly on several of the features of the Kinect to build its object recognition model. The Kinect can retrieve pixel-by-pixel distance map using its infrared sensor and a photo image using RGB camera. It provides a feature of recognizing players' real-time motion and posture. It employs a machine learning technique to learn the shape of individual humans so that it can extract the image of a human from a background or tell multiple humans apart [2]. Additionally, by building a classifier to identify body parts, this Kinect sensor can track twenty different joints on a human body. Since a certain region in a human's lateral occipitotemporal cortex selectively responds to images of human bodies [3], we can see this technology is, in a way, a simulation of the body-selective-cells in our brain.

2.2 Related Work

The object recognition model proposed in this thesis is related to knowledge that comes from human children's language acquisition studies in the developmental psychology field (covered in section 2.2.1). In order to build the model, it is important to understand how computer can recognize and learn the shape of objects (covered in section 2.2.2). In order to understand how to recognize the functional use of objects, it

was also important to review research on activity learning research in computer science (covered in section 2.2.3).

2.2.1 Study in Developmental Psychology

In a field of psychology, researchers study the strategy or mechanism that children use to acquire their first languages. They develop theories and models from different perspectives. They attempt to simulate language acquisition by implementing such models on a computer. For infants who have not yet acquired language, findings show that the infants have special skills like pattern recognition that are acquired before later learning more complex structures of linguistic communication.

For infants and toddlers to learn the names of objects, a shape bias is one of the strongest cues used to categorize novel objects; i.e., infants tend to generalize the name of an object based on similar shapes of the objects [4]. Also, researchers have shown that just with the shape bias, three-to-four month olds can categorize objects in a similar way with a basic categorization by semantic meaning [5]. Function-bias is another cue for object-name learning. Different research shows that two year-old children who learned the name of an object of which they can easily infer the function or use, generalized the name to other similarly functioning objects [6].

2.2.2 Object Recognition in Computer Science

In the field of computer science, researchers have studied ways of labeling names of objects by running a machine-learning technique on three dimensional point cloud data retrieved from both real world environments and Google 3D warehouse, which stores 3D

models of objects. In this way, Google 3D warehouse can be a teacher to teach the name of objects in the real world based on the shape of the model objects [7].

An interesting question posed by Dr. Grabner at the Computer Vision Laboratory at ETH Zurich was "What makes a chair a chair?" and his group has implemented something similar to function-bias to recognize a particular object, a chair. First, based on a shape-bias, his group generalized the shape of chairs. Then, they defined a chair as something we can sit on, and built a model that can infer if the target object is sittable or not. They first tested the procedure in a 3D virtual world setting, and then successfully applied the same method in a real world environment data reconstructed with 80 images to recognize chairs even if the shape is unusual [8].

2.2.3 Activity Recognition

In everyday activities, humans perform many routine tasks from brushing their teeth to performing heart operations. These tasks involve objects that they see and touch. Consider a log of data trace in the form <time stamp, location stamp, observation> where observations could be RFID reads, smart phone actions like receiving or responding to a text message, Kinect readings, and other types of "sensory inputs". Also, consider a collection of workflows which are named activities consisting of a set of steps, some of which may result in leaf level trace data. In order to record human activity, previous research studies assigned RFID tag to objects around humans and participants wore RFID reader embedded gloves. Then, after capturing trace data from the participants in the experiment consisting of a log of touch events, the researchers used the log of the order of touched objects to identify probably workflow activities of daily living, performing this task using dynamic Bayesian networks [9]. In order to recognize simpler segments

of activities, other researchers used four seconds of video recorded action data like walking, jogging, and running, and ran a support vector machine (SVM) learning algorithm to train a classifier to recognize sequences of those primitive actions [10].

3. APPROACH AND ARCHITECTURE

In this chapter, section 3.1 describes the background acquired learning about machine learning techniques and the Kinect SDK; then section 3.2 covers the architecture of the object recognition model.

3.1 Machine Learning Techniques and the Kinect SDK

3.1.1 Selecting a Learning Technique

To become familiar with machine learning techniques, the author developed a program for hand-written number recognition using a neural network. The program, written in Java, had a GUI and a control section, and the latter called the *neural* library from the statistical computing language R.

The control portion of the program represented a drawing canvas as a two dimensional 7x11 grid of 77 cells (see Figure 1). The user is prompted to write a number on the canvas and whenever any portion of a grid is traced, a corresponding index of Boolean array is marked as true. The neural network model used in this program is *Radial Basis Function Network* (RBFN).

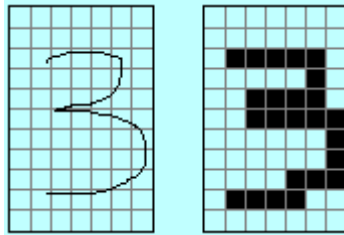


Figure 1: Grid structure for the hand-written number recognition

Hand-written data was collected 10 times for each number and saved as a .csv file format. Cell in Table 1 below represents the Boolean value of each cell of the grid, and the number on the right hand side represents the number the user are prompted to write in the corresponding trial.

trial	cell 0	cell 1	cell 2	...	cell 77	0	1	...	9
0	grid[0,0]	grid [1,0]	grid [2,0]	...	grid [10,6]	0.9	0.1	...	0.1
1	grid [0,0]	grid [1,0]	grid [2,0]	...	grid [10,6]	0.9	0.1	...	0.1
...
m	grid [0,0]	grid [1,0]	grid [2,0]	...	grid [10,6]	0.1	0.1	...	0.1

Table 1: Neural net input data layout

The program was setup with the two intermediate layers between the input and output layers; these intermediate layers contained five nodes and three nodes respectively as shown in Figure 2 below.

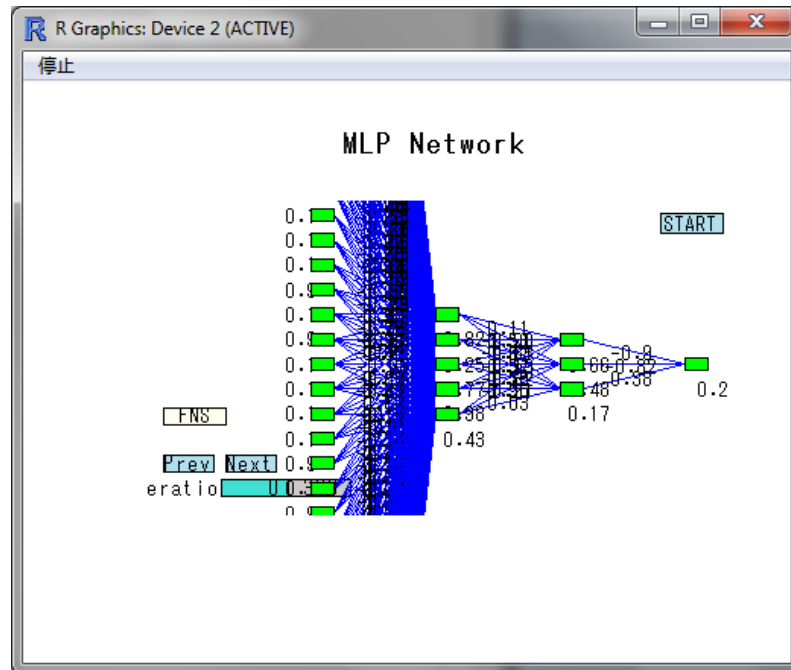


Figure 2: Neural network for the hand-written number recognition

The program took around fifteen minutes on the author’s laptop (an Intel Core i7 720QM (1.6GHz) with 4 GB of memory) for 300 iteration. The result of one run is shown in Figure 2 below.

pred. # / actual #	0	1	2	3	4	5	6	7	8	9
0	0.838	0.088	0.087	0.090	0.094	0.091	0.091	0.099	0.085	0.302
1	0.090	0.594	0.123	0.140	0.093	0.103	0.087	0.096	0.085	0.091
2	0.142	0.127	0.907	0.092	0.109	0.099	0.108	0.093	0.109	0.764
3	0.088	0.096	0.146	0.768	0.106	0.096	0.094	0.108	0.093	0.129
4	0.104	0.110	0.091	0.089	0.882	0.100	0.146	0.095	0.114	0.106
5	0.092	0.099	0.105	0.088	0.101	0.937	0.100	0.091	0.107	0.090
6	0.108	0.102	0.102	0.237	0.122	0.349	0.860	0.092	0.090	0.090
7	0.107	0.184	0.186	0.094	0.159	0.252	0.097	0.760	0.102	0.236
8	0.100	0.093	0.089	0.089	0.594	0.087	0.088	0.096	0.106	0.093
9	0.088	0.297	0.091	0.130	0.177	0.102	0.089	0.113	0.094	0.852

Table 2: Neural net program output for one experiment

Except the trials for recognizing the number “8”, the program successfully recognized the hand-written numbers. However, a problem was the speed to build the network. In order to build and test a model more quickly, the author decided to use the K-nearest neighbor learning algorithm instead of this approach.

3.1.2 Learning to Use the Kinect SDK

In this research, the Microsoft Kinect SDK for Windows was first used to retrieve the dynamic depth map in C#. Since the Kinect sensor supports depth detection in the range between 0.85 m to 4 m, code was written so any information out of bounds of that range would be displayed in white. A seek bar was then added to make it easy to change the range of focus. See Figure 3.

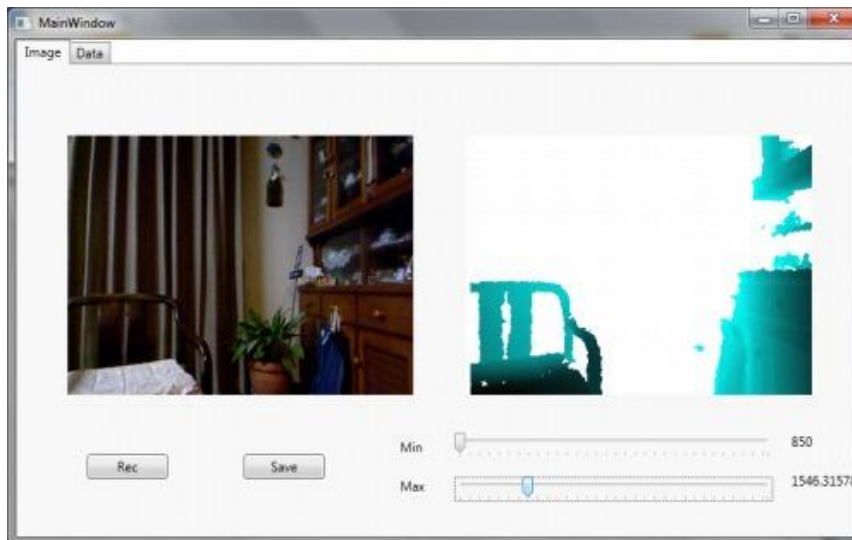
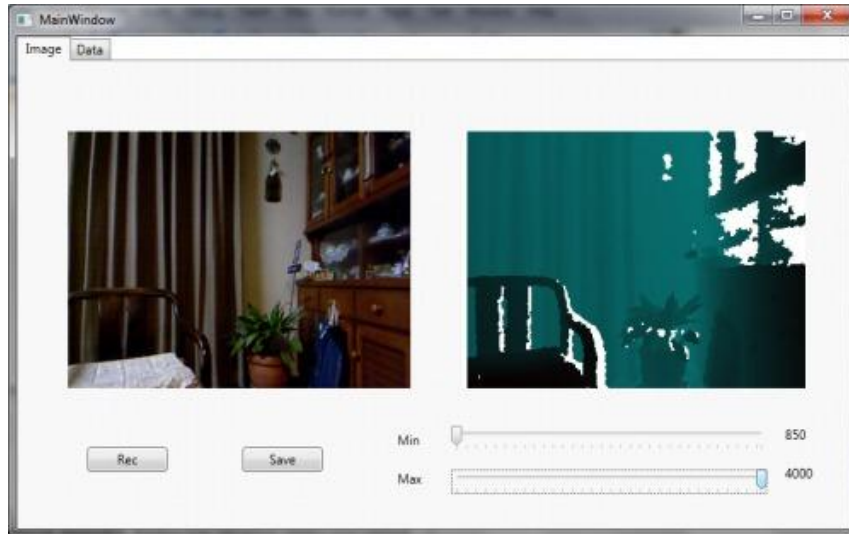


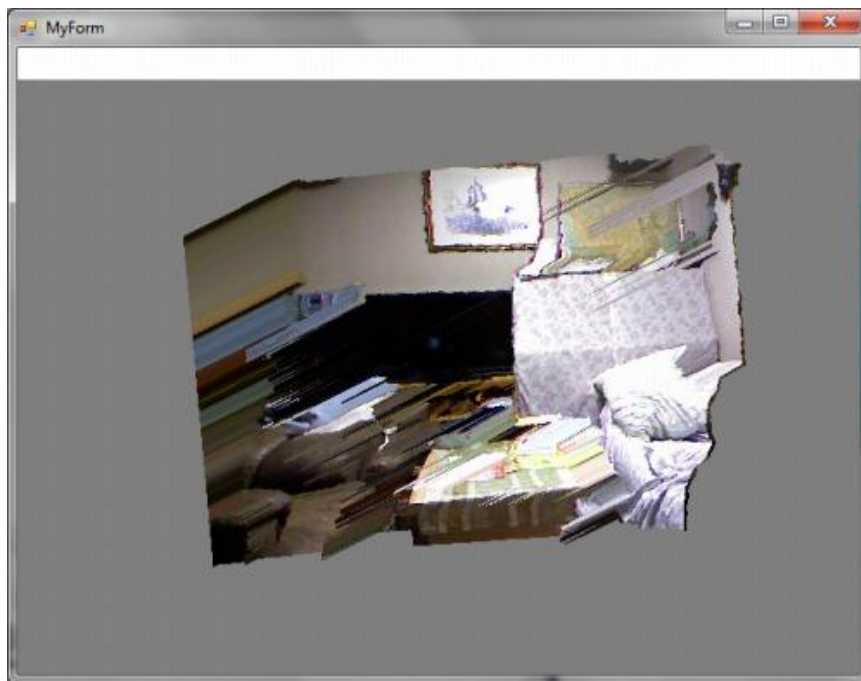
Figure 3: Depth image retrieved from Kinect

Because there is considerable noise in the raw depth map data, the author decided to apply a simple noise reduction by using a horizontal interpolation which ran a left-to-right and top-to-bottom loop through the two-dimensional-depth array to fill small holes as shown in Figure 4.



Figure 4: Noise reduction by horizontal interpolation

Using the OpenGL library, the author wrote a code to display a dynamic 3D image of environment. Initially, the approach was to draw the image by drawing a set of polygons made out of all the points retrieved from Kinect, but because of gaps between object and the background, the 3D image was poor. Instead, it was decided to add a parameter for a tolerance gap so that the program does not connect any points that exceed the gap size. See Figure 5 below.



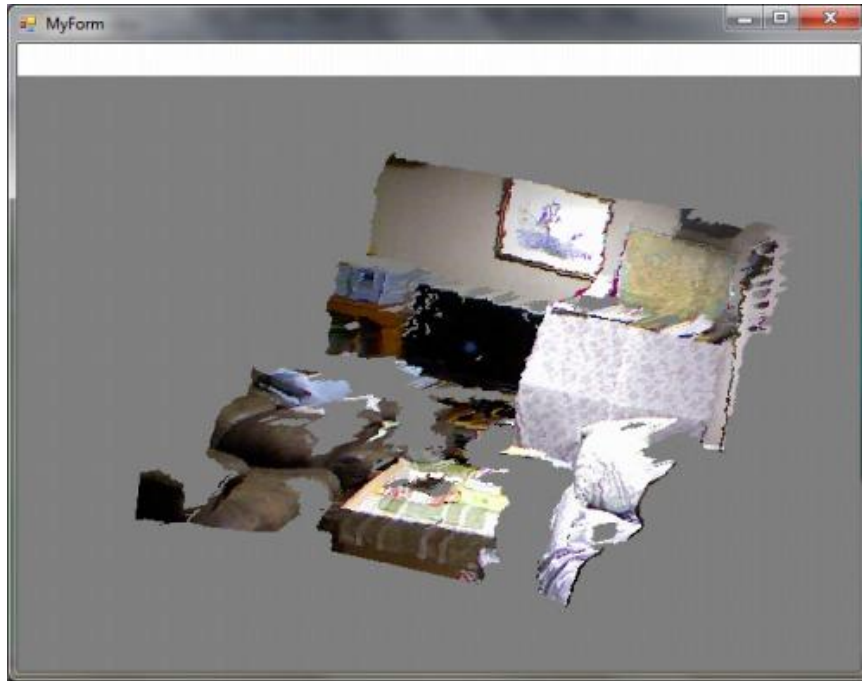


Figure 5: Set a tolerance of the gap between points to draw a smoother image

In order to build a more realistic 3D view, it was decided to use two Kinects at the same time. However, a weakness of the Microsoft Kinect SDK is that it does not support the use of multiple Kinects in a program. Therefore, it was decided to write a server and client program so that two computers, both of which are connected with one Kinect sensor, can collaborate to draw one environment from different points of view. A video can be seen at the following link: <http://www.youtube.com/watch?v=5iLW6MO6uAY>. However, because two Kinect are facing and interfere with each other in this demo, a significant amount of noise was observed as shown in Figure 6. Figure 7 shows the comparison of the actual layout of the room and the retrieved image,



Figure 6: 3D image using two Kinects

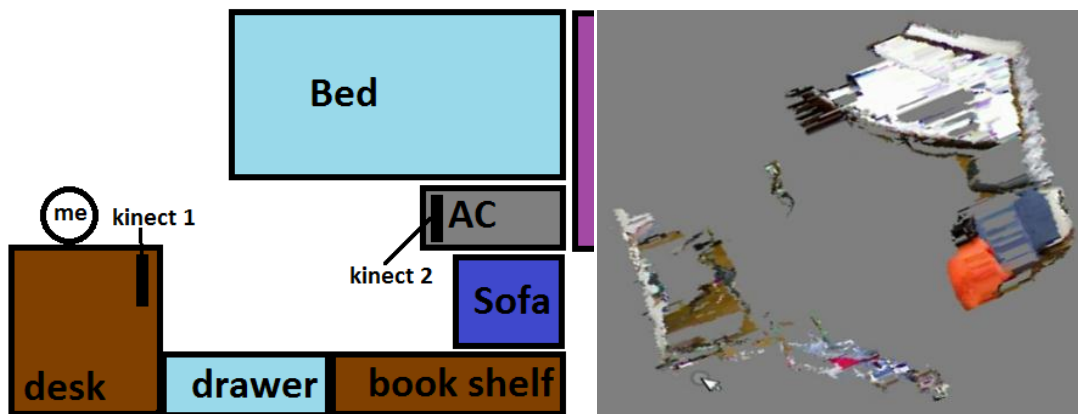


Figure 7: Room layout

3.2 Architecture of the Object Recognition Model

The proposed object recognition model consists of two distinct methods: to infer the name of object based on the object's shape and its functional use. In a teaching phase,

the user sets a target object in front of the Kinect sensor and the program learns the object's shape and name. Then the user can perform some action associated with the object to teach object's function. Later, in the testing phase, the program infers the name of an object presented based on the object's shape and its function. Figure 8 below shows the teaching use case and the testing use case.

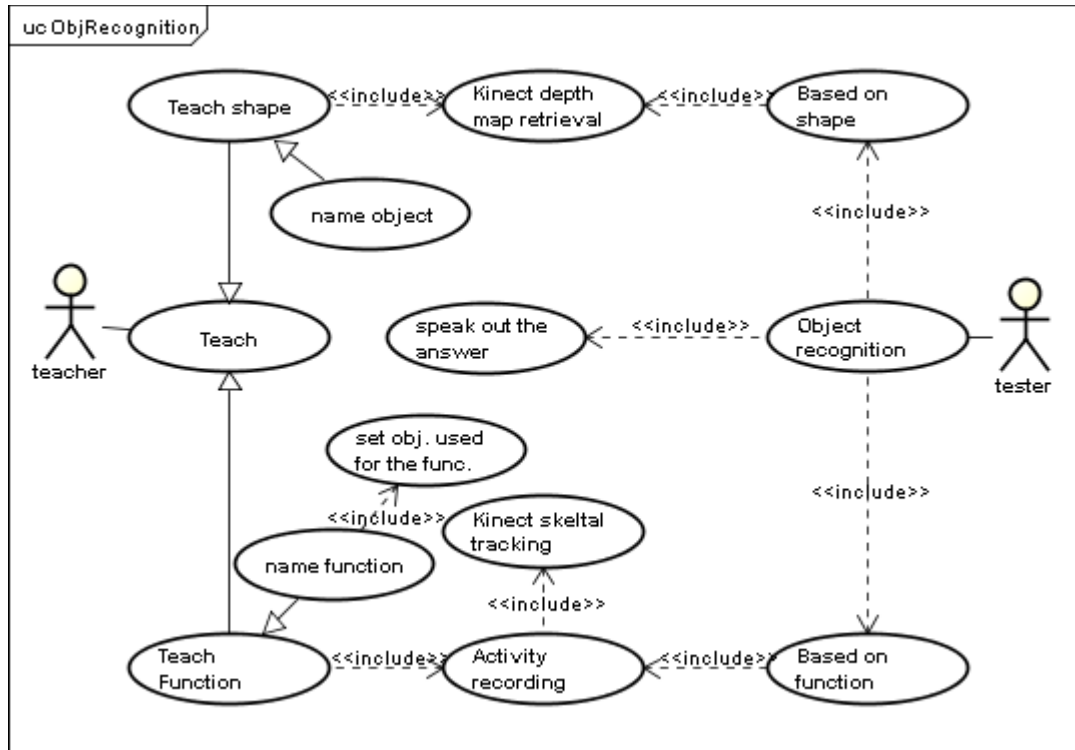


Figure 8: Use case for the object recognition model

3.2.1 Plain Surface Removal with RANSAC Algorithm

Kinects can instantly retrieve a depth map and an RGB image from a real world environment, so we might assume the Kinect provides a similar input we humans retrieve from using our eyes. However, a problem is how to separate the target object to learn from its background. We can use the already implemented seek-bar to adjust the range of focus. It is then necessary to remove the surface where the object is placed. One simple

way to do that is to use the *Random Sample Consensus* (RANSAC) routine for the 3D point clouds. The idea is that, first the program randomly takes three points from the point cloud to determine a random plain; then, it counts how many points are on the plain. By iterating through this steps many times, the program finds the plain that has the maximum number of points, assuming the plain is the surface where the objects are placed. See Figure 9.

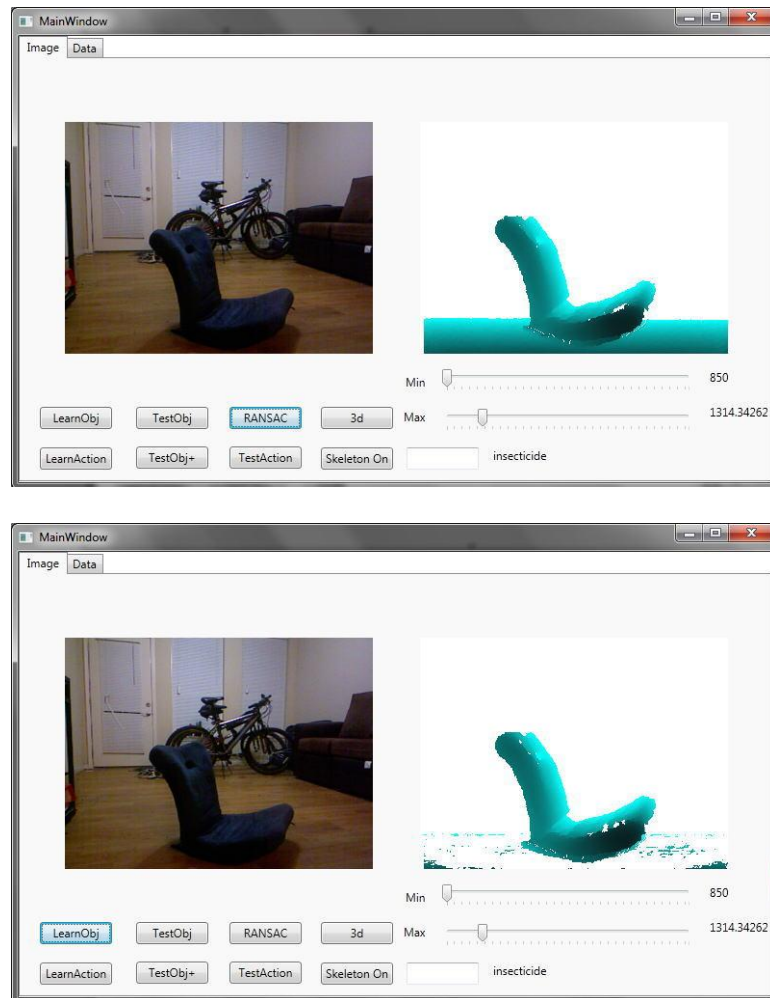


Figure 9: RANSAC for surface detection

3.2.2 K-Nearest Neighbor Algorithm for Shape Learning

Once we get a depth image of the target object, the program asks the user to name the object or asks the user to choose the name from a list of names the user has already told the program. See Figure 10 below. The user can change the angle of the same target object and label the object with the same name. Then, in the testing session, the program compares the shape of the target object with all the shape information in the memory and chooses the closest shape of objects using the K-nearest neighbor algorithm. Then, the program determines the name of the target object based on a majority vote by those k chosen sets of data.

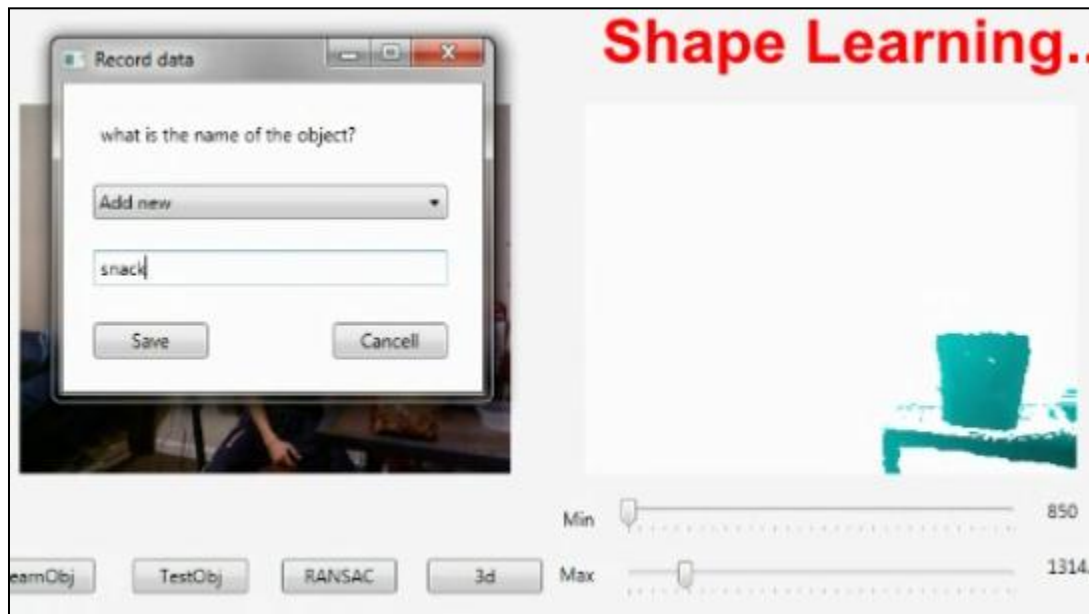


Figure 10: Naming snack based on the shape

3.2.3 Activity Recognition using the Kinect

A weakness of an object recognition model based only on shape is that although it can recognize an object that has a similar shape to the objects in its memory, it cannot

generalize the name to other object's which have the same functionality but a different shape (e.g., a chair and a sofa), or it makes a wrong answer to a similar looking object where the function is different (e.g., a glass and a vase). This problem can be solved by implementing function bias, which human children use to infer the name of objects. This thesis uses the human body recognition feature provided by Kinect SDK to distinguish uses of an object. It enables overlaying the skeletal information of a human body on top of the 3D image reconstructed based on the depth map and the RGB image.

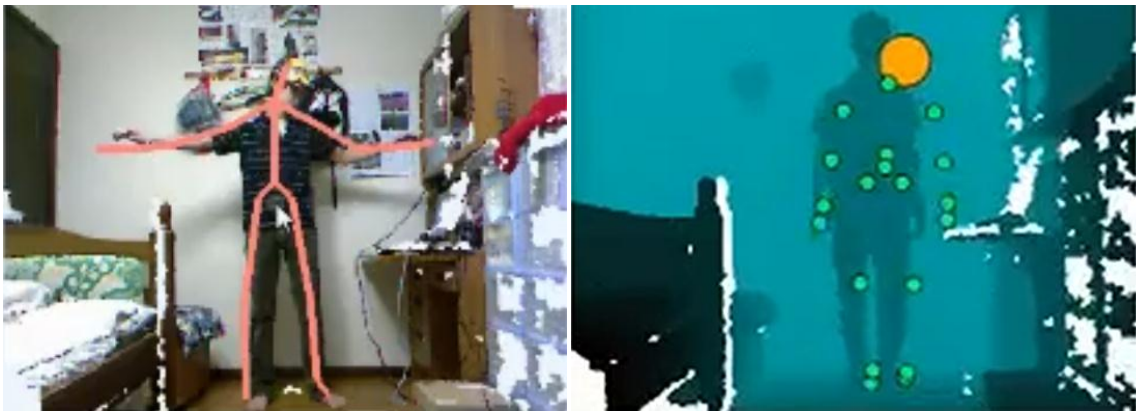


Figure 11: Skeleton on the 3D image and body joint tracking

In order to implement this kind of functional bias, one has to first recognize human action. By using the feature of Kinect to track twenty different joints of skeletal information, the program records coordinates for each joint every 0.1 seconds for 10 seconds while the teacher performs some activity. Then the system asks the teacher to name the activity. Instead of storing absolute distance of each joint from the Kinect sensor, by storing relative distance of each joint from the coordinate of the head position, the recorded data become independent of the direction the actor is facing when performing the activity. In the testing session, the program again uses the K-nearest neighbor algorithm on the stored data to infer the name of a target action. The demo can

be seen at the following link: <http://www.youtube.com/watch?v=AxCn0eKWkiQ>. In the demonstration, a teacher demonstrates walking, skipping, and running activities and provides labels for these. Later, during testing, the user goes through a sequence of walkings or skipplings or runnings and the system identifies which the sequence of activities. This demonstration by the author replicates the results from [9] but uses the Kinect instead of a more expensive solution.

3.2.4 Object Recognition Model with Shape Bias and Function Bias

In order to use both biases for object recognition, it is necessary to train two different classifiers; one is based on the shape and the other is based on the function. In addition to the feature of activity recognition, users are now prompted to choose the name of the object associated with the action. For example, the action drinking can be associated with a cup and a glass. See Figure 12.

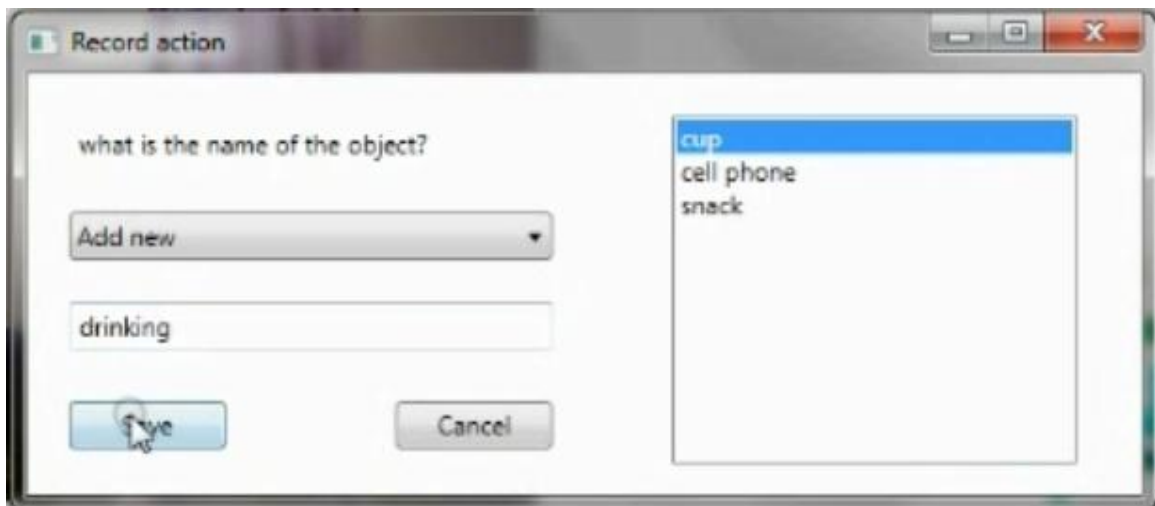


Figure 12: Choosing name of object corresponding to the action

Then, in the testing session, if the inferred name of the target object based on the shape and the function matches, the program tells the user the answer; on the other hand, if it does not match, the program tells the user the uncertainty of the answer by saying

"Maybe the object is [answer based on the shape]. But the object may be a [answer based on the action] because you used the object for [the name of action]".

If the shape is quite different from the one in the memory, the program will answer that

"I think the object is [answer based on the action] because you used the object for [name of action]. But it might be a [answer based on the shape] based on the shape."

A demo can be seen at: <http://www.youtube.com/watch?v=4ia76fzxm68>.

4. METHODOLOGY, RESULTS AND ANALYSIS

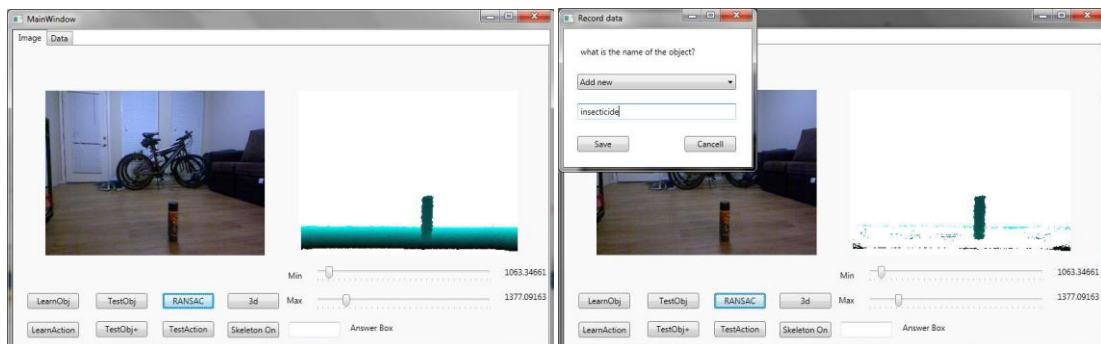
4.1 Methodology

To test the accuracy of the model, the system was tested with two objects that look similar but have a different use (a can of antiperspirant and a can of insecticide), and two other objects that look different but have the same name and function (a conventional chair and an oddly shape of a chair). The model was then tested based on three different bases. One was object recognition based on a shape; the second used activity recognition based on the body joint movement; and finally, an object recognition based on both the shape and the function. The tests worked most of the time; however, there were several constraints needed to be addressed.

4.2 Results

4.1.1 Object Recognition based on Shape

First, the program learned the shape of an insecticide can, a chair, and an antiperspirant can.



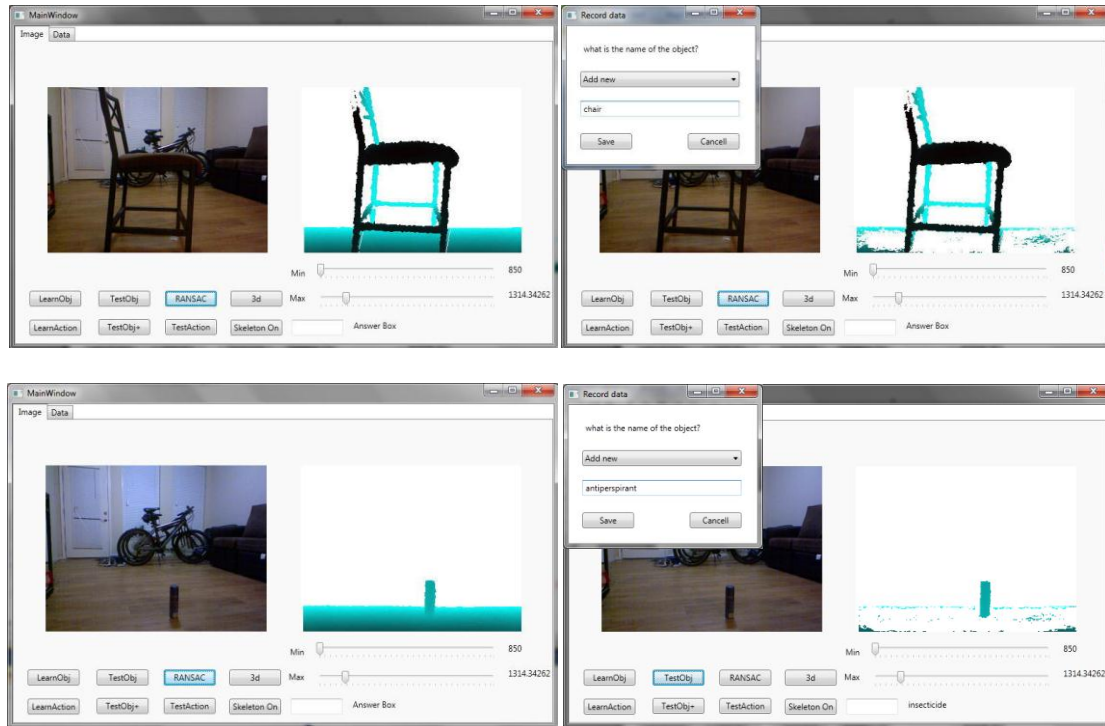


Figure 13: Shape learning of insecticide, chair, and antiperspirant

During testing, the program successfully recognized the chair all of the time, but it sometimes confused the insecticide and the antiperspirant because the shape is quite similar. Without training and when asked to name the strangely shaped chair (see figure 14) and asked the name of the object, the program failed by answering the object is antiperspirant because the program knows neither the shape of this type of object nor the use, which is to sit.

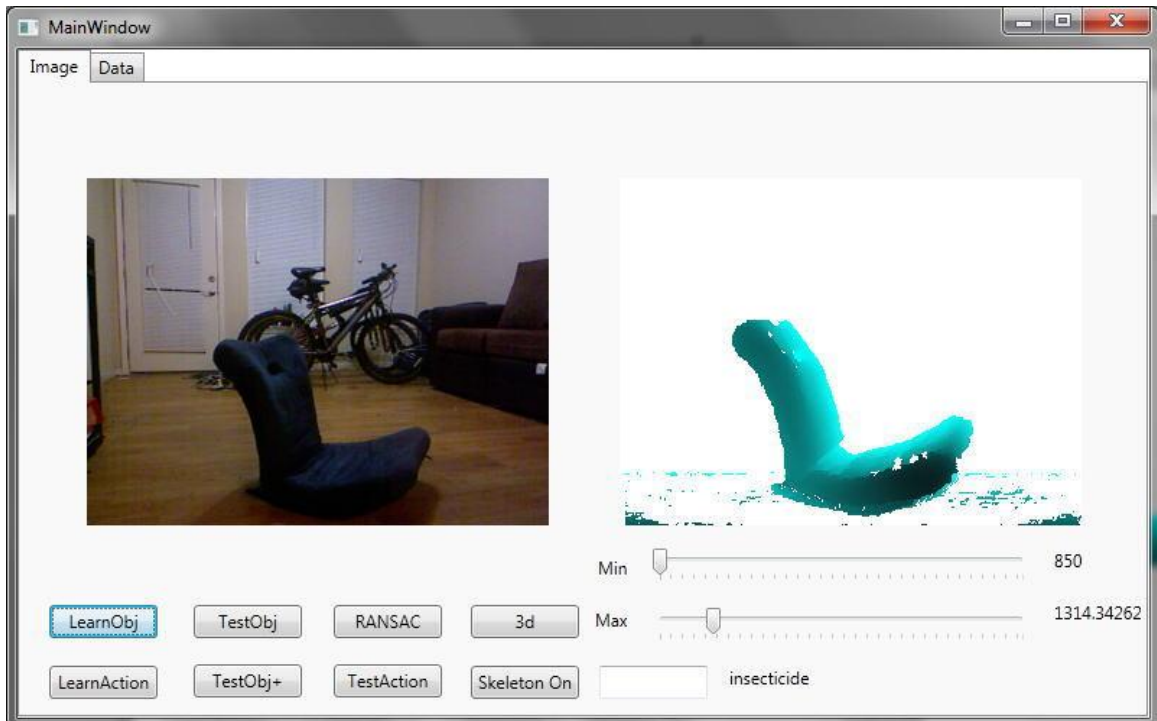


Figure 14: Different shape of a chair

4.2.2 Object Recognition based on Function

During activity learning, each action of sitting, killing bugs, and deodorizing is associated with a chair, a can of insecticide, and a can of antiperspirant respectively as shown in Figures 15-17.

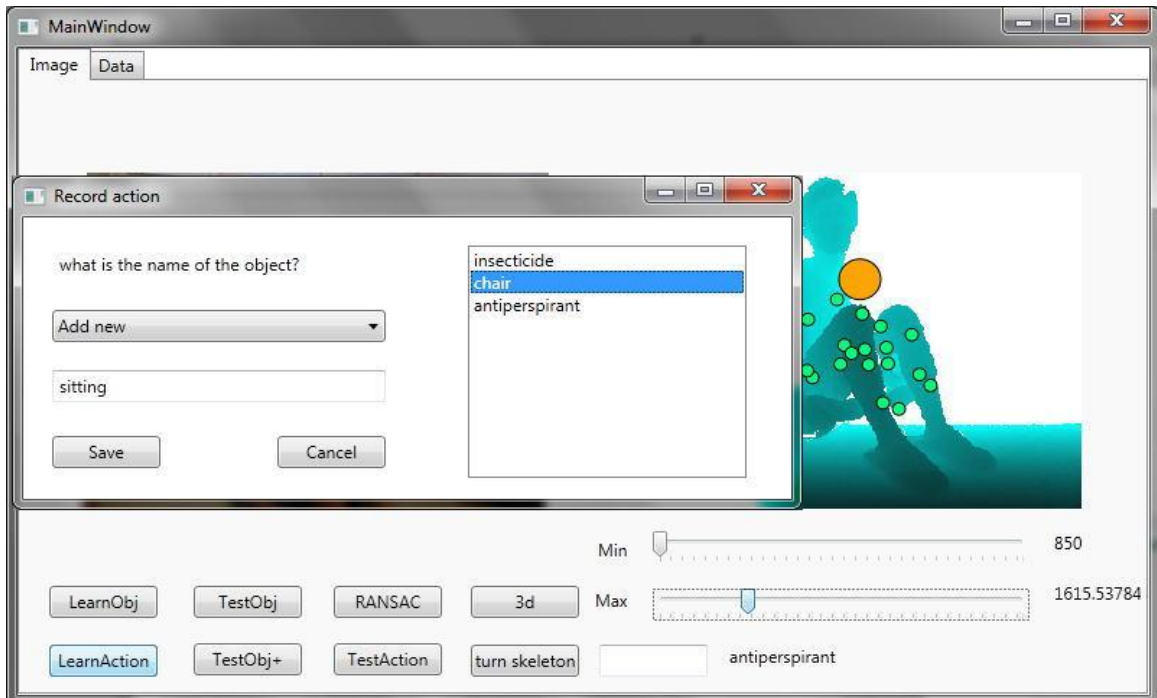


Figure 15: Sitting on a chair

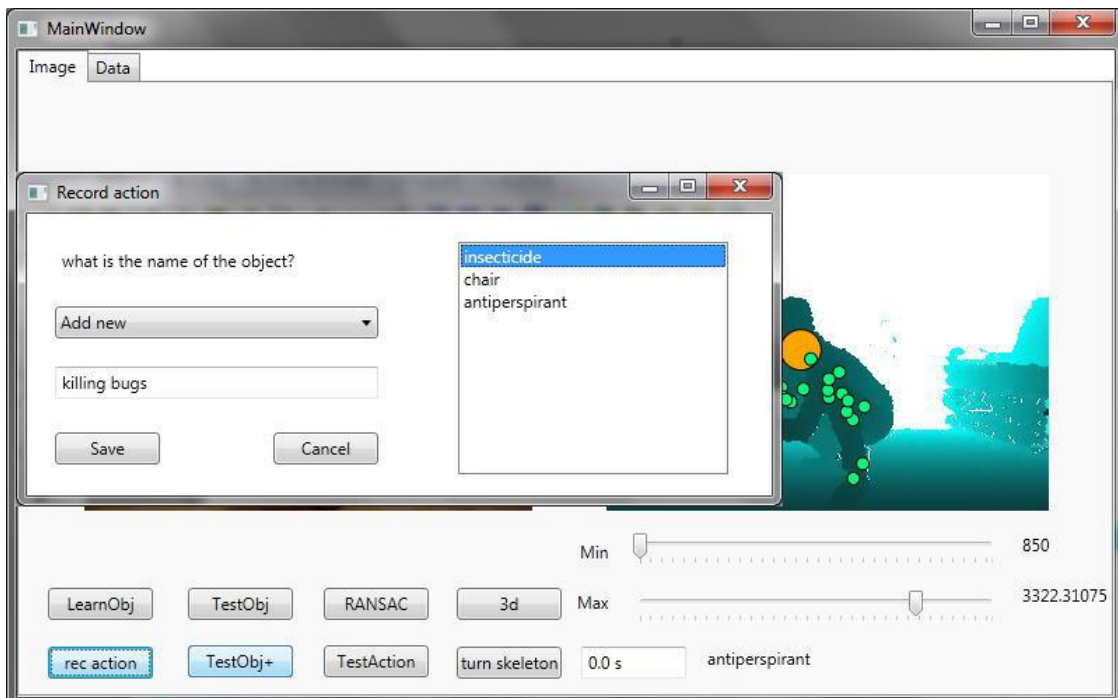


Figure 16: Killing bugs with insecticide

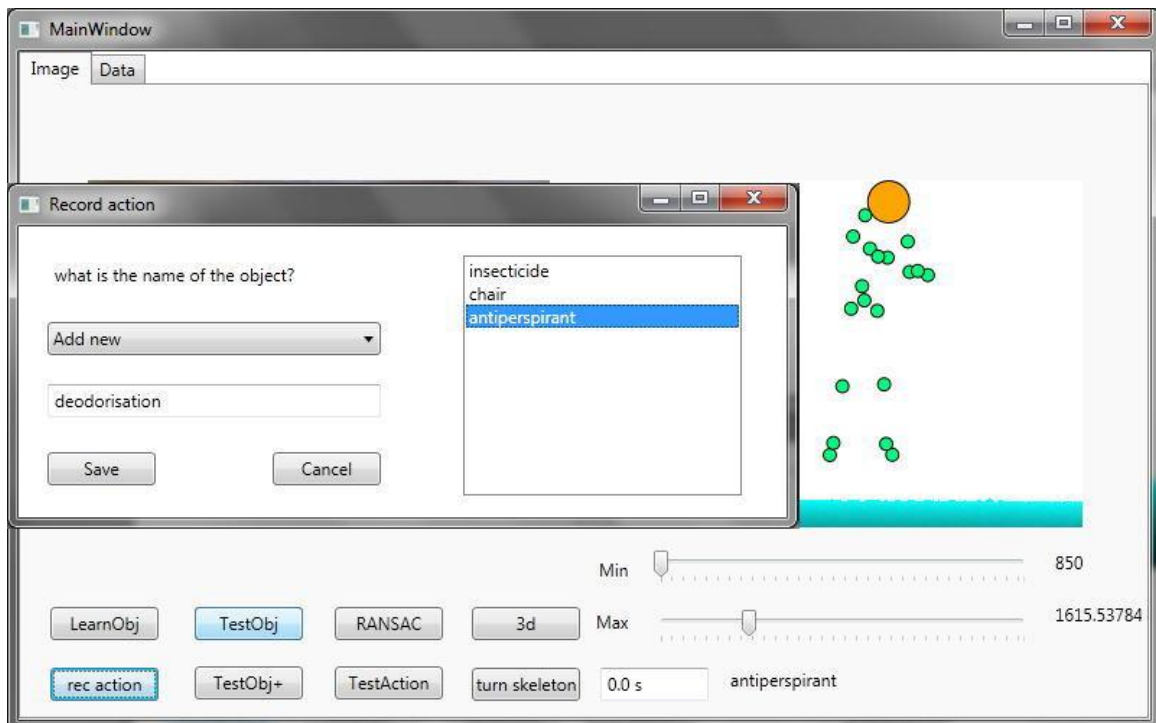


Figure 17: Deodorizing with an antiperspirant

Then, in the testing session, the program successfully identified all of those actions. Additionally, even though the action sitting was learned with an unconventional shape of chair, the program successfully answered sitting for the sitting action on a different shape of a chair as shown in Figure 18.

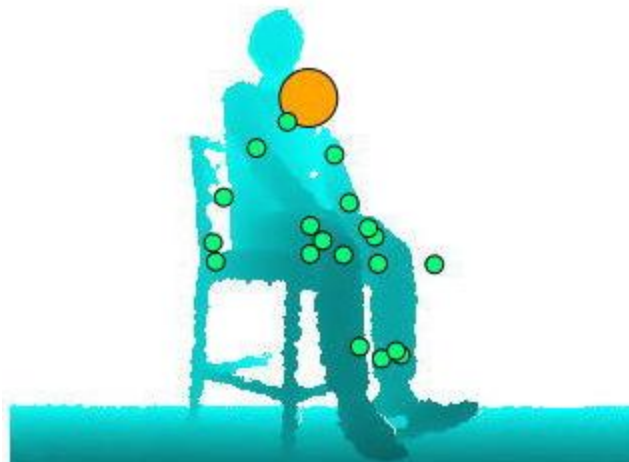


Figure 18: Testing the action sitting on a different chair

4.2.3 Object Recognition based on both Shape and Function

In the final tests, the program uses both the shape and the function to infer the target objects. In section 4.2.1, with the object recognition based only on the shape, an antiperspirant and an insecticide were sometimes confused; however, with this object recognition method, for the antiperspirant, it either correctly answered or said that

"Maybe the object is insecticide. But the object may be antiperspirant because you used the object for deodorizing".

Also, for the insecticide, it either correctly answered or said that

"Maybe the object is antiperspirant. But the object may be insecticide because you used the object for killing bugs".

Similarly, with the previous model, the different shape of chair cannot be properly recognized, but by showing the use of the object, sitting, the program answered that

"I think the object is a chair because you used the object for sitting. But it might be a antiperspirant based on the shape."

4.3 Analysis

Although the object recognition model worked as expected to properly identify the objects learned by shape or function, there were several constraints in the current model. First, because of the use of simple K-nearest neighbor algorithm on the 240×360 depth grid map, the object has to be set at the exact same position every time to be recognized as the same. This is similarly true for the activity learning. If the action associated with the object does not involve a lot of movement, like sitting, the program works well; however, if the action involves movement like walking, the shifting of timing

can be a problem. Figure 19 describes the problem of this model by the shift in position and the timing of input data.

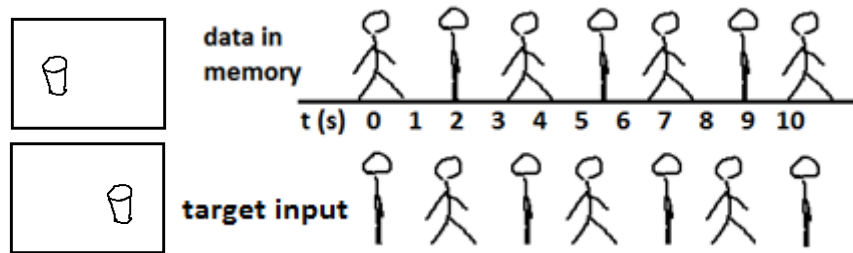


Figure 19: Case where the object or action cannot be recognized properly

5. CONCLUSIONS

5.1 Summary

This thesis proposes a new way of designing a computational object recognition model by introducing knowledge from developmental psychology. Whereas most past object recognition models use the shape of objects for the recognition, the model discussed in this thesis also uses the function of the target object for more accurate recognition. The powerful features of the Kinect sensor like depth map retrieval and human body joint recognition made the development of this proposing model easier to construct and also much less expensive. Results of testing showed that the model works as expected but also identified possible improvements for a future model.

5.2 Potential Impact

This research indicates that knowing the variety of shapes of common objects (like a chair) is important but knowing the functional use of the object also plays a role on object recognition. The thesis benefitted from its interdisciplinary use of results from psychology about how children learn to associate words with things that informed the computational model of object recognition.

If computers can learn to recognize everyday objects, many new applications will be enabled. If computers can monitor and recognize sequences of activities, many additional applications will be enabled. For only a few examples, paring both recognition capabilities may help computers including robots or even other objects recognize objects in a room or on an assembly line or activities helpful in driving a car or watching, recording and providing advice during heart operations.

5.3 Future Work

The machine learning technique used for this model can be improved by dealing with the constraints of the current model or by replacing the algorithm with a more advanced one. The name teaching part could be implemented with speech recognition technology so that the teacher does not have to manually type the words.

Improving the accuracy of object recognition would accelerate the idea of building a semantic world with smart objects. Instead of labeling objects with RFID tags as discussed in a previous paper [11], the object recognition technique can be used for identifying any object which then identifies an associated API so we can communicate with the object. Also, the accuracy of the recognition can be even improved by combining the idea with ontology field of study to narrow down the search space of objects that are mostly likely to exist in a certain environment like a kitchen where we most likely find a sink, a refrigerator, cabinets, plates, and so on [12].

Also, the activity recognition feature can be improved by parsing or recognizing the logs of trace data observations and workflow rules to identify higher level named activities. For instance, we might be able to understand that someone is packing a truck (a higher level workflow) by observing a sequence of lower level workflow like, <go to object, pick up, move object into truck> triples. In order to accomplish this work, the ideas from formal language (grammars, terminals, rules) might be useful to recognize real world activities from trace observations.

REFERENCES

- [1] E. Fix and J. L. Hodges. "Discriminatory analysis, nonparametric discrimination: Consistency properties", *Technical Report 4, USAF School of Aviation Medicine, Randolph Field, Texas*, 1951.
- [2] R. Girshick, J. Shotton, P. Kohli, A. Criminisi, and A. Fitzgibbon, "Efficient Regression of General-Activity Human Poses from Depth Images", *13th International Conference on Computer Vision*, Barcelona, Spain, 2011.
- [3] P. E. Downing, Y. Jiang, M. Shuman, and N. Kanwisher, "A Cortical Area Selective for Visual Processing of the Human Body," *Science*, vol. 293, no. 5539, 2001, pp. 2470-2473.
- [4] B. Landau, L. Smith, and S. Jones, "The Importance of Shape in Early Lexical Learning," *Cognitive Development*, vol. 3, no. 3, 1988, pp. 299-321.
- [5] P. D. Eimas and P. C. Quinn, "Studies on the Formation of Perceptually-based Basic-level Categories in Young Infants," *Child Development*, vol. 65, no. 3, Jun. 1994, pp. 903-917.
- [6] D. G. K. Nelson, R. Russell, N. Duke, and K. Jones, "Two-Year-Olds Will Name Artifacts by Their Functions," *Child Development*, vol. 71, no. 5, 2000, pp. 1271-1288.
- [7] K. Lai and D. Fox, "Object Recognition in 3D Point Clouds Using Web Data and Domain Adaptation," *The International Journal of Robotics Research*, vol. 29, no. 8, July 2010.
- [8] H. Grabner, J. Gall, and L. V. Gool, "What Makes a Chair a Chair?," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'11)*, Colorado Springs, CO, June 20-25, 2011, pp. 1529-1536.
- [9] M. Philipose, K. P. Fishkin, M. Perkowitz, D.J. Patterson, D. Fox, H. Kautz, and D. Hahnel, "Inferring Activities from Interactions with Objects," *IEEE Pervasive Computing*, vol. 3, no. 4, Oct.-Dec. 2004, pp. 50-57.
- [10] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing Human Actions: a Local SVM Approach," *Proceedings of the 17th International Conference on Pattern Recognition (ICPR)*, 2004.

[11] A. Eguchi and C. W. Thompson. "Towards a Semantic World: Smart Objects in a Virtual World," *International Journal of Computer Information Systems and Industrial Management*, vol. 3, no. 4, 2011, pp. 905-911.

[12] J. D. Eno and C. W. Thompson, "Virtual and Real-World Ontology Services," *IEEE Internet Computing*, vol. 15, no. 5, pp. 46-52, Sep./Oct. 2011, pp. 46-52.

