

# **INTERACTIVE TEXT RESPONSE FOR ASSISTIVE ROBOTICS IN THE HOME**

A Thesis  
Presented to  
The Academic Faculty

By

Morenike Ajulo

In Partial Fulfillment  
Of the Requirements for the Degree  
Master of Science in Electrical and Computer Engineering

Georgia Institute of Technology

August, 2010

# **INTERACTIVE TEXT RESPONSE FOR ASSISTIVE ROBOTICS IN THE HOME**

Approved by:

Dr. Ayanna Howard, Advisor  
School of Electrical and Computer Engineering  
*Georgia Institute of Technology*

Dr. Patricio Vela  
School of Electrical and Computer Engineering  
*Georgia Institute of Technology*

Dr. Linda Wills  
School of Electrical and Computer Engineering  
*Georgia Institute of Technology*

Date Approved: May 13, 2010

## ACKNOWLEDGMENTS

Firstly, I thank God without whom nothing is possible. I owe my deepest gratitude to Dr. Ayanna Howard who granted me the opportunity to work in the HumAnS Laboratory, and graciously guided me throughout this process. I also would like to thank the members of my thesis reading committee, Dr. Ayanna Howard, Dr. Patricio Vela, and Dr. Linda Wills for devoting time to contribute to this work. I appreciate your willingness to carve time out of your hectic schedule to add value to this work. It is an honor for me to show my appreciation to those who have contributed to my success in any way. I thank my colleagues at the HumAnS Lab for your encouragement and contributions. I thank my family for your unwavering support and for believing in me even when I doubted myself. Lastly, I thank all of my friends who have contributed through your advice and prayers. May God bless all of you.

## TABLE OF CONTENTS

|   |     |
|---|-----|
| ACKNOWLEDGMENTS .....                                       | iii |
| LIST OF TABLES.....   | vi  |
| LIST OF FIGURES .....                                       | vii |
| LIST OF EQUATIONS .....                                     | ix  |
| LIST OF SYMBOLS AND ABBREVIATIONS .....                     | x   |
| SUMMARY .....   | xi  |
| CHAPTER 1: INTRODUCTION .....                               | 1   |
| 1.1 Motivation and Objective .....                          | 1   |
| 1.2 Literature Review.....                                  | 5   |
| 1.2.1 Dialogue.....   | 5   |
| 1.2.2 Vision.....   | 8   |
| 1.2.3 Clutter Work .....                                    | 11  |
| 1.2.4 Conclusion .....                                      | 14  |
| CHAPTER 2: RESEARCH METHODOLOGY .....                       | 15  |
| 2.1 Feedback and Complexity of Scenes .....                 | 16  |
| 2.2 Human-influenced measure of Clutter and Workload .....  | 24  |
| 2.3 Human-Perceived Scene Clutter .....                     | 30  |
| CHAPTER 3: EXPERIMENTS AND RESULTS.....                     | 32  |
| 3.1 Real Images.....  | 32  |
| 3.2 Human Subject Test and Human Survey Response.....       | 33  |
| 3.3 Feature Congestion Measure of Clutter .....             | 36  |
| 3.4 Workload Algorithm.....                                 | 41  |
| 3.5 Results.....  | 44  |
| 3.5.1 Feedback Questions and Workload.....                  | 44  |
| 3.5.2 Feature Congestion Measure.....                       | 48  |
| 3.5.3 Human Measures .....                                  | 51  |
| 3.6 Implementation of Interactive Text Response (ITR) ..... | 54  |
| CHAPTER 4: CONCLUSION AND FUTURE WORK .....                 | 60  |
| APPENDIX A: REAL IMAGES OF KITCHEN SCENES.....              | 62  |

|   |    |
|---|----|
| APPENDIX B: VARIABILITY, EASE OF RECOGNITION, CLUTTER, AND WORKLOAD<br>DATA FOR ALL OBJECTS ..... | 64 |
| APPENDIX C: MATLAB CODE FOR SHAPE RECOGNITION.....  | 78 |
| APPENDIX D: MATLAB CODE FOR COLOR RECOGNITION.....  | 84 |
| REFERENCES .....  | 87 |

## LIST OF TABLES

|           |  |    |
|-----------|--|----|
| Table 1.  | Sample images to relate complexity of scenes to number of feedback questions.....        | 24 |
| Table 2.  | Sample of calculation for Variability.....   | 26 |
| Table 3.  | Sample of calculation for Ease of Recognition.....                                       | 27 |
| Table 4.  | Sample of calculation for Clutter.....   | 28 |
| Table 5.  | Sample calculation for Workload.....   | 29 |
| Table 6.  | Categorization of Clutter.....   | 30 |
| Table 7.  | Survey Responses and Standard Deviation of Responses.....                                | 34 |
| Table 8.  | Initial categorization of FC values based on maximum agreement among human subjects..... | 39 |
| Table 9.  | Feature Congestion values and corresponding categorization.....                          | 40 |
| Table 10. | Classification of Workload Values into five different levels.....                        | 46 |
| Table 11. | Categorization of Number of Feedback Questions.....                                      | 47 |
| Table 12. | Percentage Match between Workload and Number of Questions.....                           | 47 |
| Table 13. | Categorizations of Averaged object clutter and Scene clutter.....                        | 49 |
| Table 14. | Averaged object clutter and Scene clutter for each kitchen scene image<br>.....          | 49 |
| Table 15. | Comparisons among FC and Averaged Object clutter and Scene clutter<br>.....              | 51 |
| Table 16. | Comparisons among Human Measures, Averaged Object clutter, and Scene clutter.....        | 51 |
| Table 17. | Comparisons among Averaged Object clutter, FC, and Human Measures of clutter.....        | 53 |

## LIST OF FIGURES

|            |   |    |
|------------|---|----|
| Figure 1.  | A typical IVR Technology.....   | 2  |
| Figure 2.  | Simple diagram to show the function of ITR Technology.....                                  | 3  |
| Figure 3.  | Maggie robot interacting with a child.....  | 6  |
| Figure 4.  | Main and Sub FSMs.....  | 7  |
| Figure 5.  | Thinking Robot (T-Rot).....   | 9  |
| Figure 6.  | A sample of the recognition process for the general case.....                               | 16 |
| Figure 7.  | A sample recognition process where object is not found after shape inquiry.....             | 19 |
| Figure 8.  | A sample of the recognition process when the object of interest is occluded.....            | 22 |
| Figure 9.  | Range of Clutter Categorization showing Average Human Clutter.....                          | 30 |
| Figure 10. | A screen-shot of a survey page.....   | 31 |
| Figure 11. | Augmentation of images to focus on items in scene.....                                      | 33 |
| Figure 12. | Image 74; all human subjects categorize as 'Not cluttered'.....                             | 35 |
| Figure 13. | Image 77; all human subjects categorize as 'Not cluttered'.....                             | 35 |
| Figure 14. | Image 93; no agreement between human subjects.....  | 36 |
| Figure 15. | Feature Congestion Values for each kitchen scene image.....                                 | 37 |
| Figure 16. | Image 77 with FC of 1.565.....  | 37 |
| Figure 17. | Image 25 with FC value of 2.6626.....   | 38 |
| Figure 18. | Illustration of the Factors of Recognition for each object in Kitchen Scene (Image 74)..... | 42 |
| Figure 19. | Illustration of the Factors of Recognition for each object in Kitchen Scene (Image 86)..... | 42 |
| Figure 20. | Illustration of the Factors of Recognition for each object in Kitchen Scene (Image 99)..... | 43 |
| Figure 21. | Truncated Distinct Workload Values.....   | 45 |

|            |  |    |
|------------|--|----|
| Figure 22. | Comparisons between Averaged object rating and scene rating according to a Human subject response..... | 52 |
| Figure 23. | A sample of the wooden blocks used in implementation.....  | 54 |
| Figure 24. | Identified squares as the result of Shape Recognition.....   | 55 |
| Figure 25. | Identified red square as the result of Color Recognition.....  | 56 |
| Figure 26. | Image scene of wooden blocks used in ITR implementation.....   | 57 |
| Figure 27. | Identified rectangles as the result of Shape Recognition.....  | 58 |
| Figure 28. | Identified green rectangles as the result of Color Recognition.....                                    | 58 |

## LIST OF EQUATIONS

|                  |    |
|------------------|----|
| Equation 1.....  | 25 |
| Equation 2.....  | 27 |
| Equation 3.....  | 28 |
| Equation 4.....  | 28 |
| Equation 5.....  | 29 |
| Equation 6.....  | 56 |
| Equation 7.....  | 56 |
| Equation 8.....  | 57 |
| Equation 9.....  | 57 |
| Equation 10..... | 59 |
| Equation 11..... | 59 |
| Equation 12..... | 59 |
| Equation 13..... | 59 |

## LIST OF SYMBOLS AND ABBREVIATIONS

|  |  |
|--|--|
| DTMF – Dual tone multi frequency                   | FOV – Field of View  |
| IVR – Interactive Voice Response                   | V – Variability  |
| ITR – Interactive Text Response                    | $s_{\text{object}}$ – Shape of object of interest  |
| TTS – Text-to-speech                               | $h_{\text{object}}$ – color of object of interest  |
| ASR – Automatic Speech Recognition                 | E – Ease of Recognition  |
| DSN – Dragon Naturally Speaking                    | $\alpha$ – Describes how much the shape of an object is affected by occlusion and color bleeds |
| FSM – Finite State Machine                         | C – Clutter  |
| PRM – Probabilistic roadmap                        | $i_{\text{total}}$ – Total number of objects in a scene  |
| SIFT – Scale Invariant Feature Transform           | j – Object counter   |
| BPANN – Back Propagation Artificial Neural Network | M – Maximum number of objects in a scene   |
| PCA – Principal Component Analysis                 | W - Workload   |
| HRI – Human Robot Interaction                      |  |
| FC – Feature Congestion                            |  |
| NOQ – Number of feedback questions                 |  |

## SUMMARY

In a home environment, there are many tasks that a human may need to accomplish. These activities, which range from picking up a telephone to clearing rooms in the house, all have the common trend of fetching. These tasks can only be completed correctly with the consideration of many things including an understanding of what the human wants, recognition of the correct item from the environment, and manipulation and grasping of the object of interest.

The focus of this work is on addressing one aspect of this problem, decomposing an image scene such that a task-specific object of interest can be identified. In this work, communication between human and robot is represented using a feedback formalism. This involves the back-and-forth transfer of textual information between the human and the robot such that the robot receives all information necessary to recognize the task-specific object of interest. We name this new communication mechanism Interactive Text Response (ITR), which we believe will provide a novel contribution to the field of Human Robot Interaction.

The methodology employed involves capturing a view of the scene that contains an object of interest. Then, the robot makes inquiries based on its current understanding of the scene to disambiguate between objects in the scene. In this work, we discuss development of ITR in human-robot interaction, and understanding of variability, ease of recognition, clutter, and workload needed to develop an interactive robot system.

# CHAPTER 1

## INTRODUCTION

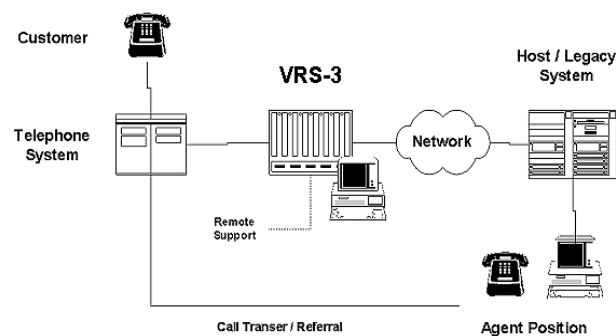
### 1.1 Motivation and Objective

In a home environment, there are many tasks that a human must accomplish. For individuals with disabilities, these tasks are traditionally more difficult to complete without assistance. Depending on the severity of the disability, such individuals are not totally independent, thereby requiring the help of a caregiver or a healthcare worker. There is a way to make things easier for such a group of people while affording them the proper amount of independence they desire. If one such person has a robotic assistant in the home with him, he can send this assistant on errands to help with instrumental activities of daily living [1]. These activities, which range from picking up a telephone to cleaning rooms in the house, all have the common goal of fetching. These tasks can only be done correctly with the consideration of many things, including the understanding of what the human wants, recognition of the correct item from a cluttered and/or uncluttered space, and manipulation and grasping of the object of interest. However, there are underlying problems with the robot's ability to accomplish these tasks, namely understanding the task of interest, and recognition of the object of interest.

The focus of this research is to address one aspect of this problem, which involves decomposing an image scene such that a task-specific object of interest can be identified. In this research, communication between human and robot is structured using a feedback formalism. This involves the back-and-forth transfer of textual information between the

human and the robot such that the robot receives all information necessary to recognize the task-specific object of interest.

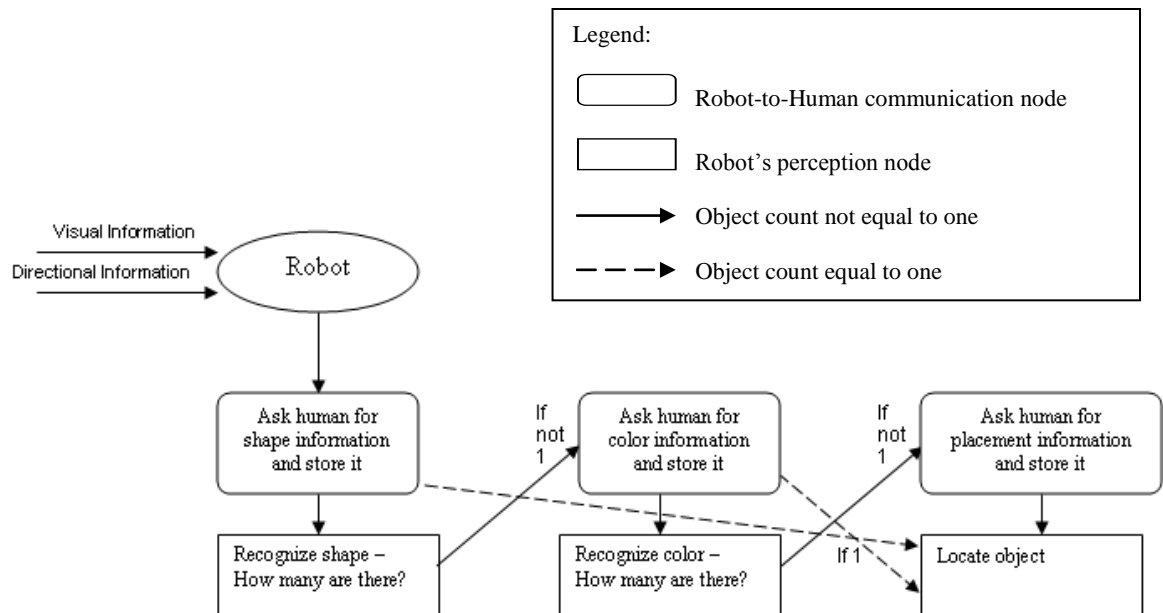
The type of interaction employed in this work is similar to that found in Interactive Voice Response (IVR) technology [2]. IVR is a technology used in major enterprises such as credit card companies and banks to automate interactions with telephone callers. It acts as a replacement for receptionists by interacting with users through voice command, and requiring a user to input information through the pressing of telephone keys or through voice recognition software. If a caller presses a key on the keypad, the system uses that specific key's dual tone multi frequency (DTMF) to determine which number is pressed, and if the caller speaks instead, a voice recognition system is used to determine what is said. After the system recognizes the caller's input, the caller is then routed to the corresponding representative, or sometimes, other menus are activated. Therefore, the main idea of IVR, Figure 1, is that a caller makes his way through a list of menus to reach his service of interest.



**Figure 1:** A typical IVR Technology<sup>1</sup>

<sup>1</sup> <http://www.programmingconcepts.com/MS-CRMIVR.asp>

This process of interactive communication is employed in this master’s thesis work and modified such that it is compatible with robot interaction. The robot takes, as input, textual commands from the user and provides, as output, textual responses. One modification to the technology is the mode of communication in which the output information from the system is given. In the original IVR technology, the output information from the company to the caller is verbal, whereas in the new technology, the information is textual. We name this new communication mechanism Interactive Text Response (ITR), which we believe will contribute something novel to the field of Human Robot Interaction. Figure 2 shows a simple diagram that describes the communication flow utilized in this work.



**Figure 2:** Simple diagram to show the function of ITR Technology

We will use ITR to help the robot accomplish its goals in stages, that is, the robot incrementally obtains specific information from its human counterpart in order to gradually accomplish its goal. The full methodology employed involves capturing a view of the scene that contains an object of interest. Then, the robot makes inquiries based on its current understanding of the scene to disambiguate between objects in the scene. These inquiries include the shape and color information of the object of interest, and placement information for clarification. Through intensive feedback, the robot eventually recognizes the object of interest.

In this work, we discuss development of the ITR role in human-robot interaction, and understanding of variability, ease of recognition, clutter, and workload needed to develop an interactive system. Variability describes the differences among all the items in a display image; the more unlike the items are, the easier they will be to recognize. Ease of recognition describes how well an item can be recognized, and clutter is described as “the state in which excess items or their representation or organization, lead to a degradation of performance at some task” [3]. Workload is described as the number of inquiries made to correctly identify an object of interest in a scene. The importance of all these factors attests to the need for quantifying them and establishing relationships amongst them, both of which we address in this work.

## 1.2 Literature Review

### 1.2.1 Dialogue

A large number of the successful interactions between human and robot are based on the dialogues between the two entities in both master-slave and peer-to-peer setups. The majority of the previous works on the topic of dialogue are implemented with social robots and service robots. The dialogues implemented range from text to tactile to gestures to speech covering the wide spectrum of verbal and non verbal communication. In our work, we implement a non verbal (text) form of dialogue for communication between the robot and human for service tasks in the home environment.

Maggie [4], shown in Figure 3, is a personal robot developed for social interaction at RoboticsLab of the University of Carlos III in Madrid. Social robots interact with humans by following behavioral norms in the society in which they interact. To achieve this goal of normalcy, Maggie was designed to use a similar combination of communication modes that humans utilize in their human-to-human interaction. It has been said that the majority of the communication between humans is non-verbal; therefore, a successful robot in peer-to-peer robot-human interaction needs to be able to incorporate other nonverbal communication mechanisms. Maggie incorporates different communication modalities such as verbal, emotional expression, and audiovisual expression. It uses base sensors and webcams to perceive human gestures, and invisible tactile sensors on its body and computer screen to detect touching, mouse movements and drawn gestures. It also uses a Dragon Naturally Speaking (DSN) Client SDK as a speech recognizer and an Automatic Speech Recognition (ASR) module to convert spoken

words into text. On the robot-to-human side of the dialogue, a Text-to-Speech (TTS) module is implemented to convert the robot's text to audible speech.



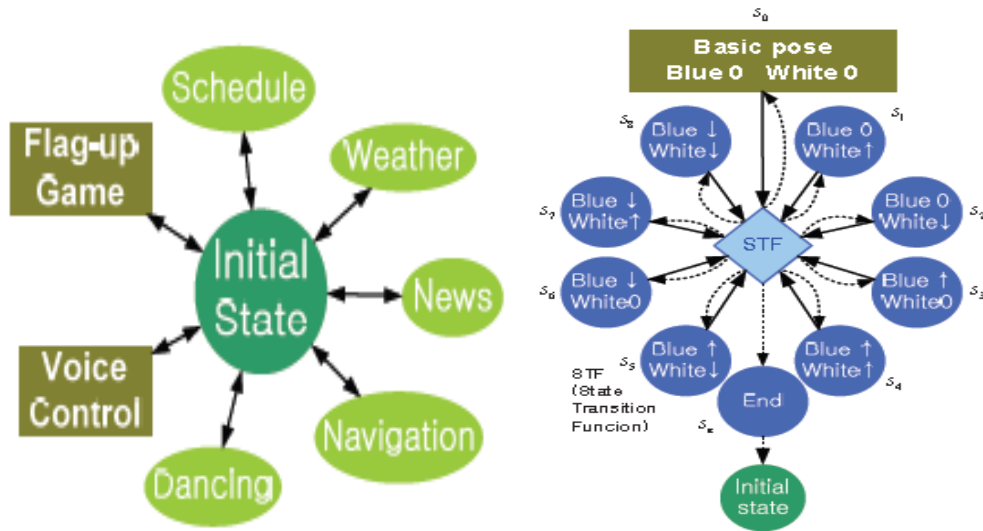
**Figure 3:** Maggie robot interacting with a child<sup>2</sup>

A dialogue system for a service robot application was developed by Lee et al. [5] to use on a general mobile robot platform. The system consists of an ASR module, TTS module, and a dialogue manager for supervision of conversion of input command into action. The dialogue manager utilizes embedded Finite State Machines (FSM) along with voice command to direct the robot to execute given commands. When the robot receives a command, it moves from initial rest state to carry out one of its many services using a main FSM. Within each service state, there may be another FSM with which the robot

---

<sup>2</sup> <http://roboticslab.uc3m.es/roboticslab/gallery.php?albumname=maggie>

accomplishes the specified task. After service is complete, the robot exits the inner FSM, and then the main FSM to return to the initial rest state. A structure of the main and sub FSMs can be seen in Figure 4.



**Figure 4:** Main and Sub FSMs [5]

There is a vast array of work in robotics that utilizes different types of dialogues for communication between robotic systems and humans. The works we have described above are representative of the work being done in this broad area.

### 1.2.2 Vision

A substantial amount of the sensory information humans receive are visual and in order to design robotic systems that can perform well in human surroundings, there is a need for them to be equipped with hardware needed to obtain this visual information. Previous works utilize vision in conjunction with other sensors for a range of applications including motion planning [6] for safe transport from start to goal positions, gestural recognition [7,8] to understand given commands or signals, and facial detection [8] for human tracking security. Our work utilizes vision for object recognition using the features associated with an object of interest.

Hong Liu and Jie Zhou developed a motion planning method for safe Human Robot Interaction in dynamic environments using visual feedback [6]. Motion planning enables robots to independently plan their paths based on roadmaps to move from an initial position to a goal position. The method developed in this work utilizes binocular stereo vision to detect the position of people, Scale Invariant Feature Transform (SIFT) to detect and match features in people, and an improved probabilistic roadmap (PRM) to plan a safe path for the robot. Images were obtained from two cameras, yielding two different directions of view. Using the SIFT algorithm, the areas that contain people were cut out and matched between two images to compute the position of the people in the live environment. The modified PRM method was used to plan the shortest path without human obstacles. This path is continuously evaluated to check for new obstacles as someone may have moved into a node that was previously free.

Yang et al. [7] developed a new method for recognizing whole-body gestures in HRI by addressing the known problems of segmentation and recognition in classifying human motions. Their method focused on modeling the transition motions between key gestures explicitly. In order to correctly generate a 3D model of humans in a recorded scene, they developed an algorithm to extract feature points from human images. They also constructed key gesture and transition models by finding the end points of legitimate gestures and considered everything else as transition gestures. They tested their method on T-Rot, a personal service robot for the elderly. The robot, shown in Figure 5 is equipped with two stereo cameras mounted on a pan tilt unit in its head to recognize gestures and nearby faces or objects. It utilizes the method developed in this work to recognized whole-body gestures to identify emergency situations such as sitting down or falling down to the floor.



**Figure 5:** Thinking Robot (T-Rot)<sup>3</sup>

---

<sup>3</sup> <http://www.plasticpals.com/?p=2506>

Chang and Chou [8] developed a system to detect face and hand-gestures in a Human Robot Interaction scenario. They incorporated three different schemes based on shape, similarity and lip color to extract possible human faces from images. After processing the images for illumination and edge enhancement, they approximated the human face as an ellipse, and used a fixed ratio of length to width to search for potential faces. Skin color is used as a criterion to implement the similarity aspect, and a human lip-color model is used to search for lip-color pixels in the skin-colored ellipse. In order to simply detect gestures, they restricted the portion needed for detection to the section beyond the wrist, and used a combination of Principal Component Analysis (PCA) and Back Propagation Artificial Neural Network (BPANN) classifiers.

We have highlighted a few of the various robotic works that utilize vision as a mode of sensing. There are several works in robotics that use vision to enable robotic systems to accomplish many tasks including sensing of surrounding, recognition of commands, and path-planning for identification of an optimal, collision-free path from location to location.

### 1.2.3 Clutter Work

In prior literature, clutter has many definitions. It is the background signal similar to a target signal [9], structures that distract or confuse an observer about the properties of a target [10], and the state in which excess items or their representation lead to degradation of performance of a task [3]. Regardless of its definition, clutter is present in our daily lives, and in order for robots to perform well in the human world, they will need to be able to deal with clutter as humans have. Many publications are available on the topic of clutter, but they mostly address the issue of how humans view clutter and the effect of clutter on human beings' performance [10, 11], how to quantify clutter [3], and how to draw attention to targets regardless of clutter [12]. In our work, we are using the understanding of how humans view clutter to help robots assess the level of clutter in their surroundings, especially when performing service tasks for the humans.

Ewing et al. [10] addresses the effect clutter has on a human's ability to detect a target. They defined clutter as the structure that distracts or confuses an observer about the properties of a target. They considered four factors of clutter which they called 'clumpiness', background radius, target radius, and target contrast in determining the effects of clutter on visual performance while using response time and hit rate, the probability of a correct decision, to measure performance. Their results showed that as background radius increased, hit rate decreased for targets regardless of target radius. Additionally, they found that for a value of clutter factor, hit rate increased with increasing target radius, increased with increasing target contrast, and increased with increase in both target contrast and size.

Bulakowski et al. [11] addressed issues relating to acting in cluttered environments. They tasked themselves with finding out how crowding affects visually guided behavior, specifically grasping, and testing the hypothesis that the visuomotor system disregards global context information when a target object is placed within a cluttered environment. Visual crowding is defined as the degradation of peripheral feature discrimination that varies with viewing eccentricity and density of surrounding objects. Their first experiment to confirm the stated hypothesis was based on a stimulus of a target bar radially surrounded with an array of similar bars. An observer, while fixing his gaze on a dot to the left of the stimulus, attempts to make judgments on the orientation of the target bar and then proceeds to grasp it. The second experiment was setup in a similar way to the first, but with the targets placed at a fixed distance to the top and bottom of the observer's focus to determine how crowding affects grasping in different fields of view. From the first experiment, they found that crowding had a similar effect on perception and visually guided action, with discrimination performance decreasing with increasing eccentricity and increasing density. They also found that as the density of distractors grow, the perceived orientation of the target bar more closely resembles its actual orientation, while the grasped orientation deviates more from the actual orientation. From the second experiment, they found that crowding effects do not differ for grasping and perception, but they are stronger in the upper versus the lower fields of view.

Rosenholtz et al. [3] developed the Feature Congestion measure of display clutter based on the modeling of the saliency of elements of display. Their saliency model was based on the idea that an item is salient if its features (color, luminance contrast, and

orientation) are outliers to the distribution of the features of other items in the display. Therefore, the higher the saliency of an item, the easier it will be to detect. They suggested that increased congestion leads to degraded performance because as more items are added to a display, it increases the volume of the feature space, thus becoming more difficult to add a salient item. In other words, the features that distinguish the present items in the display become less outstanding. Therefore, in order to measure clutter correctly, there is a need to track the level of feature congestion in an image. The Feature Congestion measure treats the display image as a background and questions how easy it will be to add a salient item. Their algorithm, designed to implement this measure of clutter, accepts an image as input and computes a clutter value for the image, where a high clutter value signifies a highly cluttered scene and a low value signifies a minimally cluttered scene.

Along with the issue of clutter is the issue of saliency since there is a need for target items in cluttered and uncluttered scenes to be prominent. There are many efforts that discuss saliency in relation to recognition, grasping and clutter. Itti et al. [12] developed a computer implementation of a bottom-up scheme for the control of visual attention using information from multiple modalities of orientation, intensity and color. Using a combination of normalization and filtering techniques as described in [12], they fused the information from the different modalities to create a single saliency map which is a two-dimensional map that encodes the saliency of objects in an environment. Their model was tested on a variety of real color images with significant amounts of noise, varying light effects, varying degrees of occlusion, and different textures. They found that their computer model scanned the images in a behaviorally correct order and

performed well in selecting salient targets from a cluttered scene. In order to make quantitative comparison between the model and human's visual system, they presented the same images as were given to the computer to 62 human subjects with prior close-up views of what the target objects looked like. The search times were compared and it was found that in the majority of the test images, the model found the target objects faster than the humans even though the humans had the advantage of prior knowledge of the target objects.

These are just few examples of the works in Human Cognition that have attempted to define clutter, study its effect on humans, and quantify it.

#### 1.2.4 Conclusion

Our work does not deal with any of the applications discussed above separately, but brings them together in a way to make human robot interaction richer. The robotic system in this work is designed to continuously interact with a human user through textual dialogue. It decomposes a visual capture of its surrounding in order to identify a task-specific object. We have also developed a metric for a robotic system in an interactive situation with a human counterpart to autonomously evaluate its surrounding and provide feedback about its ability to recognize each of the objects in it. This enables the system to provide a notification to its counterpart about the environment and about the efficiency of its performance, thereby, providing a novel contribution to the field of Human Robot Interaction.

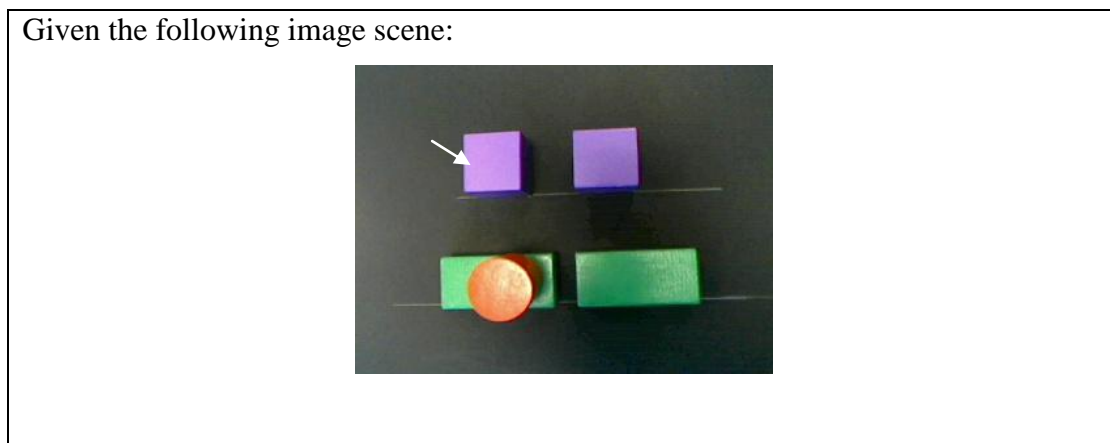
## **CHAPTER 2**

### **RESEARCH METHODOLOGY**

Our robotic system utilizes the techniques of Image Processing along with Interactive Text Response to identify task specific objects of interest. The focus in our human-robot interaction scenario is to enable the two entities to work closely together to assist the robot in recognizing an object of interest in a scene. This is done through a feedback mechanism, allowing the human to feed the robot information regarding the object of interest. This information, which includes shape, color, and location, are provided to the robot when it inquires about them. Our theory is that the number of feedback questions depends on the complexity of the scene of interest; recognition of an object in a cluttered scene will be harder, thereby requiring more feedback inquiries, than recognition of the same object from a simple scene. Also, since some of the autonomous factors required for the robot to recognize an object of interest lies with the human, there should be a definition of clutter that brings in the human aspect of HRI. For this, we have developed metrics for a new measure of clutter based on the four factors of recognition which are variability of objects, ease of recognition of objects, clutter, and associated workload.

## 2.1 Feedback and Complexity of Scenes

Figure 2 in Chapter 1 shows the flow of information between the robot and the human; where the robot first obtains shape information from the human, and if the shape is found in the scene, the system highlights all the objects that have that shape. If there is more than one highlighted object, the system inquires about the color of the object of interest, and all the objects with the same color are highlighted. If, again, more than one object is highlighted (having the same shape and color as the object of interest), the system asks about the location of the object of interest while segmenting the scene. The system first divides the scene into two (left and right), and asks the human if the object of interest is on the left side or the right. If after the first segmentation, the correct object is still not recognized, the system, again, divides the first segment into two (top and bottom), and asks the human if the object is on the top or the bottom. This process of segmentation is repeated until the object of interest is found. A simple example is outlined in Figure 6.



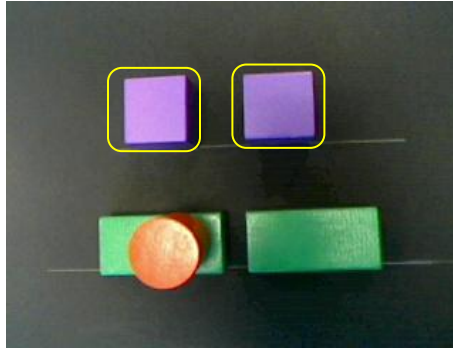
**Figure 6:** A sample of the recognition process for the general case

Recognize the top-left purple square

1. First Robot Inquiry: Shape?

Human Response: Square

a. Shape recognition:



Robot's considerations:

- a. Was the shape found? Yes
- b. Is there more than one object with the same shape? Yes.

2. Second Robot Inquiry: Color?

Human Response: Purple

a. Color recognition:

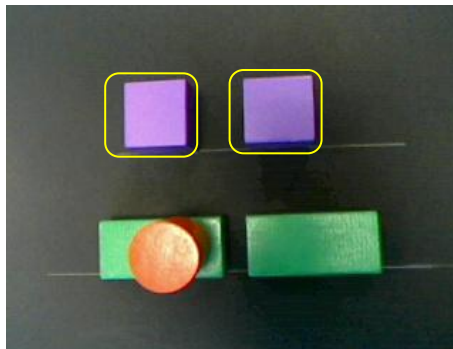


Figure 6 continued

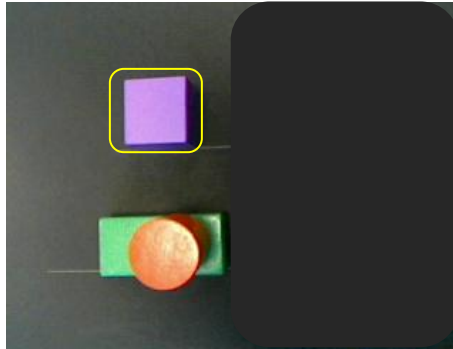
Robot's considerations:

- a. Was the color found? Yes
- b. Is there more than one object with the same color? Yes.

3. Third Robot Inquiry: Location?

- a. Location Segmentation: Left?

Human Response: Yes



Robot's consideration: Is there more than one object on the left side? No.

4. Is the recognized object the object of interest?

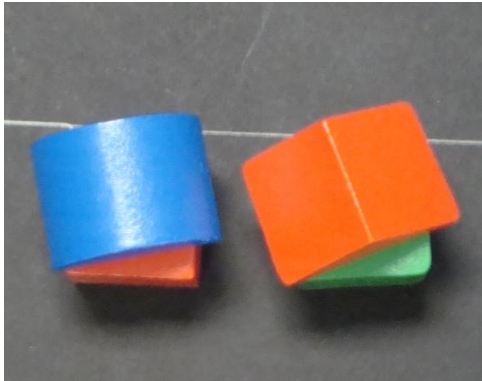
Human Response: Yes

5. Number of Questions asked: 3

**Figure 6 continued**

Since it is assumed that the object is in the scene, in the cases where the shape is not found after the first (shape) inquiry, the robot asks for the color of the object of interest and carries out color recognition on the whole original scene. The same process as that above is carried out to recognize the object of interest. A simple example is illustrated in Figure 7.

Given the following image scene:



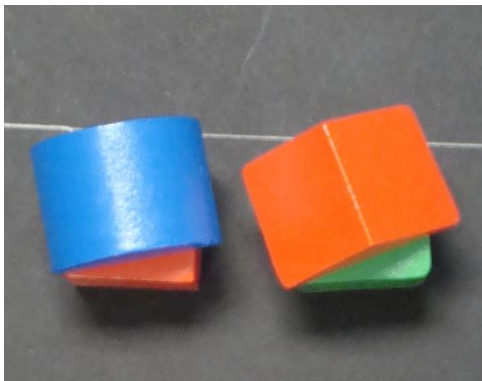
Recognize the following red square block identified below:



1. First Robot Inquiry: Shape?

Human Response: Square

a. Shape recognition:



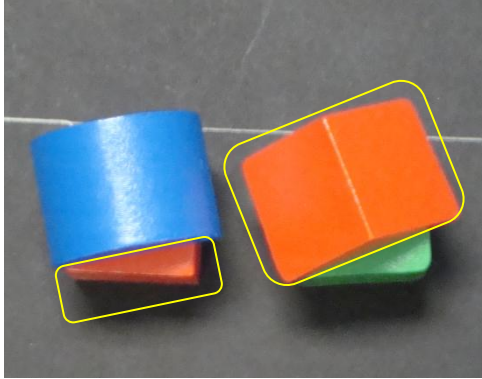
Robot's consideration: Was the shape found? No.

2. Second Robot Inquiry: Color?

Human Response: Red

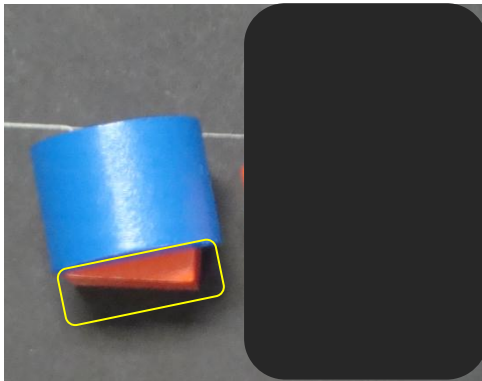
a. Color recognition:

**Figure 7:** A sample recognition process where object is not found after shape inquiry



Robot's considerations:

- a. Was the color found? Yes.
  - b. Is there more than one object with the same color? Yes.
3. Third Robot Inquiry: Location?
- a. Location Segmentation: Left?  
Human Response: Yes



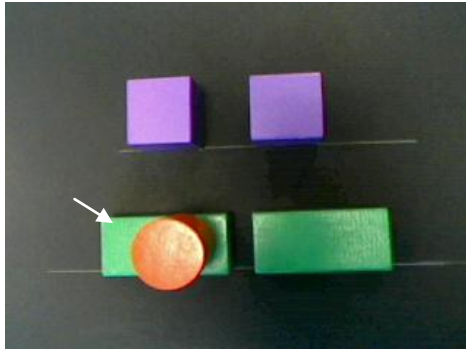
Robot's consideration: Is there more than one object on the left side? No.

4. Is the recognized object the object of interest?  
Human Response: Yes
5. Number of Questions asked: 3

**Figure 7 continued**

In all cases, after final recognition is made, the system asks the human if the correct object has been recognized. If the human gives a negative response, the system states that the object may be occluded and therefore it is unable to recognize it. This failsafe measure addresses cases in which there is more than one object with the same shape as that of interest, but the object of interest is occluded by the others, as seen in Figure 8. Therefore, in this case, shape recognition highlights the wrong objects, and color recognition is carried out on the wrong (highlighted) objects, on which location segmentation is done, thereby resulting in the wrong object being recognized as the correct one.

Given the following image scene:

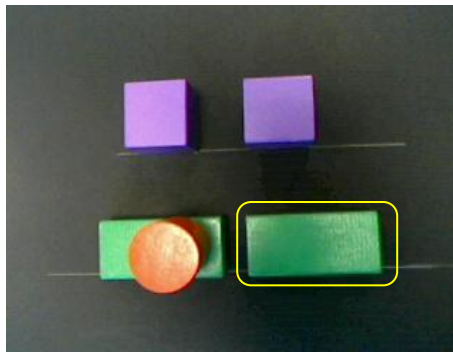


Recognize the bottom-left green rectangle that is under the red circle

1. First Robot Inquiry: Shape?

Human Response: Rectangle

a. Shape recognition:



Robot's considerations:

a. Was the shape found? Yes

b. Is there more than one object with the same shape? No.

2. Is the recognized object the object of interest?


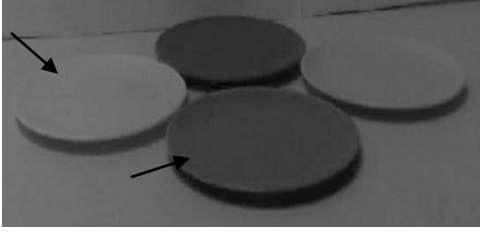
Human Response: No

3. Number of Questions asked: Maximum (=50)

**Figure 8:** A sample of the recognition process when the object of interest is occluded

We theorize that the number of inquiries made by the robot depends on the complexity of the scenes. Therefore, in order to test this theory and to test the ITR technology of the system on real scenarios, we captured some real kitchen scenes and asked a human subject to go through them and note how many questions the person asks in order to aid the robotic system in recognition of each of the objects within the scene. The human is tasked to use three basis of recognition, namely shape, color, and location segmentation, as described. For the scenes in which the system is unable to recognize the object correctly and those having objects inseparable from nearby objects due to difficult segmentations, the subject was told to use a maximum question boundary value. We found for the twenty-seven images provided, there was a range of 1 to 12 questions for the identifiable objects, and a maximum value to the unidentifiable ones (maximum number is 50). Table 1 shows examples of images and the corresponding number of questions (NOQ) for some objects of interest. The complex scene, on the left, has a variety of values for NOQ ranging from 1 to 50, whereas the simpler scene, on the right, has two values for NOQ ranging from 3 to 4. This shows an example of the relationship between complexity of scenes and number of inquiries made per object.

**Table 1:** Sample images to relate complexity of scenes to number of feedback questions

|   |  |
|---|--|
|  |  |
| <p>NOQ(glass) = 1<br/>NOQ(plate) = Maximum (=50)</p>                              | <p>NOQ(white plate) = 3<br/>NOQ(grey plate) = 4</p>                                |

## 2.2 Human-influenced measure of Clutter and Workload

Previous works use methods relating to computer vision to measure clutter.

Therefore, these measures don't relate to human perception; hence, we believe these measures are inaccurate for measuring clutter in situations such as ours where the human works closely with the robot in the recognition process. Therefore, we feel that the classical measures of clutter need to be adjusted to allow for the human's view of clutter. One of the available measures is discussed in a work by Rosenholtz, et.al. [3] where they developed a Feature Congestion measure to measure clutter as seen in displays. This measure is based upon the idea that as more items are added to a display, populating a greater volume of feature space, the harder it is to add a new salient item to the display. They measure the level of congestion in an image to predict how easy it will be to add a salient item. Therefore, their measure of clutter is based on how congested is the feature space. To do this, they compute the local feature covariance across multiple scales for each feature (color and luminance contrast), combine the covariance values across scales

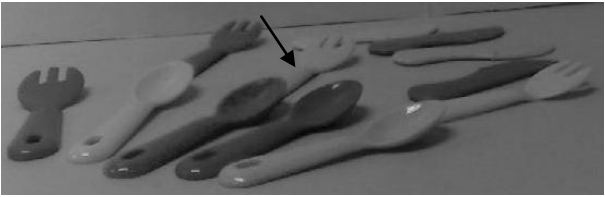

by calculating the maximum at each pixel, combine the covariance values across features by taking an average of the two features, and finally, combine the values across space by taking the average over the entire image, thereby resulting in a single measure of clutter. This measure therefore only tells how cluttered a scene is, and not how well each of the items in a scene can be recognized.

We have developed an algorithm to relate workload, which we define as the number of feedback questions required to recognize an object of interest, to the objects in a scene. Our algorithm starts by measuring the variability of an object within a scene, and then, it measures the ease of recognition of the object, the clutter factor of the object, and the associated workload.

Variability is defined for each object in the surrounding scene based on how many other objects in the scene are similar to that of interest. Similarities are classified based on two features of shape and color. We believe that variability of an object within a scene depends on how many objects in that scene resemble it in shape,  $s_{object}$ , and how many resemble it in color,  $h_{object}$ . We developed a metric to capture these two aspects in an autonomous fashion. A robotic system is capable of counting distinct similar shapes and colors in a scene. Equation 1 shows the metric for measuring variability of an object,  $V(object)$ , within a scene, and Table 2 shows the calculation for variability of some objects of interest in two sample scenes.

$$V(object) = \frac{1}{s_{object} * h_{object}} \quad (Equation 1)$$

**Table 2:** Sample of calculation for Variability

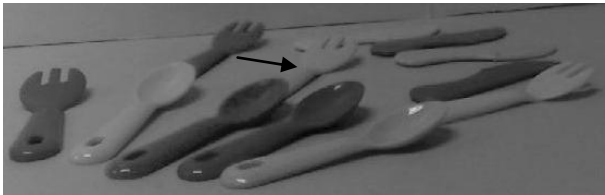
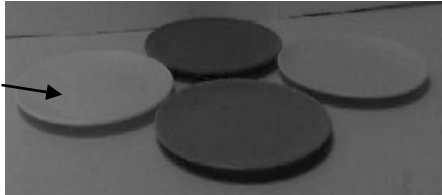
|   |  |
|---|--|
|  |  |
| $V(\text{fork}) = \frac{1}{4 * 6} = \frac{1}{24} = 0.0417$                        | $V(\text{plate}) = \frac{1}{4 * 2} = 0.125$  |

Ease of recognition describes how well an object in a scene can be recognized. There are many things that factors into how easy an object is to recognize, however we focus on variability, occlusion or insertion, and color bleeds (i.e. when an object is difficult to distinguish from a nearby object of similar color). A highly variable object is easy to recognize since it is very much unlike other objects in the scene. An occluded object or one inserted into another or another inserted into it is very hard to recognize in any scene, because its shape will appear altered by the occluding objects. We have designed the metric illustrated by Equation 2 to account for all these aspects of ease of recognition, where  $E(\text{object})$  is the ease of recognition of the object of interest, and  $\alpha$  describes the level of occlusion or bleeding issues affecting the recognition of the shape of the object of interest. Level of occlusion,  $\alpha$ , is 0 when there are no issues of occlusion or color bleed affecting the recognition of an object. It is 1 when the object of interest is affected by one other object or by the field of view, that is, it is on the edge of the FOV, such that part of the object is cut off. It is set to a maximum number of 4 when more than one object is affecting the recognition of the shape of the object of interest, and when the

object is undistinguishable. Table 3 shows the calculation for ease of recognition of some objects of interest in two sample scenes.

$$E(object) = \frac{V(object)}{2^\alpha} \quad (Equation 2)$$

**Table 3:** Sample of calculation for Ease of Recognition

|  |  |
|--|--|
|  $E(fork) = \frac{V(fork)}{2^2} = 0.0104$ |  $E(plate) = \frac{V(plate)}{2^0} = 0.125$ |
|--|--|

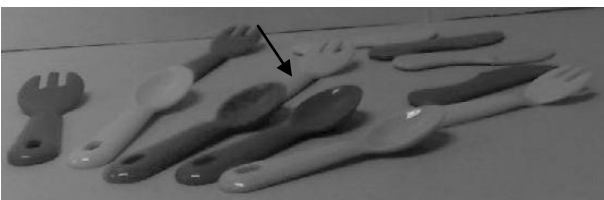
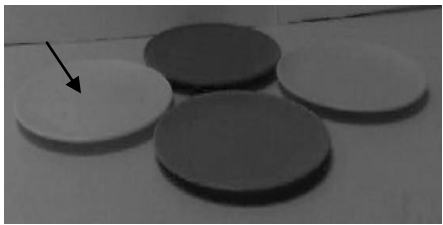
Clutter is described as the state in which excess items or their representation or organization, lead to a degradation of performance at some task [3]. According to this definition of clutter, the clutter value of a scene describes how much the environment affects recognition of the objects within the scene. We feel that clutter is dependent on the ease of recognition of each of the objects in a scene. If each of the objects in a scene is easily recognizable, then the scene has a small amount of clutter. We have developed a metric to measure the clutter value of a scene,  $C(scene)$ , as expressed in Equation 3, where  $i_{total}$  is the total number of items in the scene, and  $E(j)$  is the ease of recognition of the object denoted by  $j$ . We obtain the clutter value by averaging ease of recognition over the objects in the scene and inverting it to show that clutter increases with decreasing

ease of recognition. Then, we normalized the value by a maximum number of items in a scene, denoted by  $M$ , which we take as 100. The scene-specific clutter measure can be simplified to create an object-specific measure,  $C(\text{object})$ , expressed in Equation 4, by taking  $i_{\text{total}}$  to be 1. Table 4 shows the calculation for the clutter value of some objects of interest in two sample scenes and the clutter value of the scenes in which they are placed.

$$C(\text{scene}) = \frac{1}{M} * \left[ \sum_{j=1}^{i_{\text{total}}} E(j) \right]^{-1} \quad (\text{Equation 3})$$

$$C(\text{object}) = \frac{1}{M} * \left[ \sum_{j=1}^1 E(j) \right]^{-1} \quad (\text{Equation 4})$$



**Table 4:** Sample of calculation for Clutter

|   |  |
|---|--|
|    |    |
| $C(\text{scene}) = \frac{1}{100} * \left[ \sum_{j=1}^{12} E(j) \right]^{-1} = 0.0259$ $C(\text{fork}) = \frac{1}{100} * [E(\text{fork})]^{-1} = 0.96$ | $C(\text{scene}) = \frac{1}{100} * \left[ \sum_{j=1}^4 E(j) \right]^{-1} = 0.02$ $C(\text{plate}) = \frac{1}{100} * [E(\text{plate})]^{-1} = 0.08$ |

Workload,  $W(\text{object})$ , describes how much work is required to recognize an object within a scene, as evident by the number of questions the robot needs to ask a human to be able to recognize an object of interest in a scene. We feel that it is very important to be able to measure this factor such that the robotic system can inform its human counterpart ahead of the recognition process, and the human can in turn make a more-informed decision about whether he/she wants to go through the recognition process. That is, if the workload for a certain object is too high, the human can decide to disregard that object because the object is already considered one of the unrecognizable ones. This measure, expressed in Equation 5, is dependent on the object-specific clutter measure,  $C(\text{object})$ . We feel that the relationship between workload and clutter is exponential because workload increases rapidly as a scene becomes more cluttered. Table 5 shows the calculation for the workload of some objects of interest in two sample scenes.

$$W(\text{object}) = e^{C(\text{object})} \quad \text{Equation(5)}$$

**Table 5:** Sample calculation for Workload

|  |   |
|--|---|
|  $W(\text{fork}) = e^{C(\text{fork})} = 2.6117$ |  $W(\text{plate}) = e^{C(\text{plate})} = 1.0833$ |
|--|---|

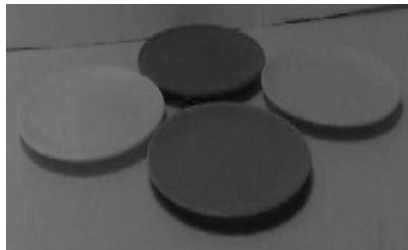
We theorize that if we run our workload algorithm on real images, the number of questions obtained from the first human subject test should be comparable to the workload values that the algorithm yields.

### 2.3 Human-Perceived Scene Clutter

We obtained twenty-seven kitchen scenes and asked three human subjects to rate each scene and place each in one of five categories (1-5), as explained by Table 6 and shown in Figure 9. Each person rated the scenes without knowledge of this work or how their ratings will be used. They were asked to denote their responses on a survey sheet as seen in Figure 10.

**Table 6:** Categorization of Clutter

| Categories | Explanation          |
|------------|----------------------|
| 1          | Not Cluttered        |
| 2          | Moderately Cluttered |
| 3          | Very Cluttered       |
| 4          | Very, Very Cluttered |
| 5          | Extremely Cluttered  |






a.) Average Human Clutter is 1



b.) Average Human Clutter is 5

**Figure 9:** Range of Clutter Categorization showing Average Human Clutter

Please rank the following images on how cluttered you think they are.

|  |               |                     |
|--|---------------|---------------------|
|   | Not cluttered | Extremely cluttered |
|  | 1 2           | 3 4 5               |
|   | Not cluttered | Extremely cluttered |
|  | 1 2           | 3 4 5               |
|  | Not cluttered | Extremely cluttered |
|  | 1 2           | 3 4 5               |

**Figure 10:** A screen-shot of a survey page

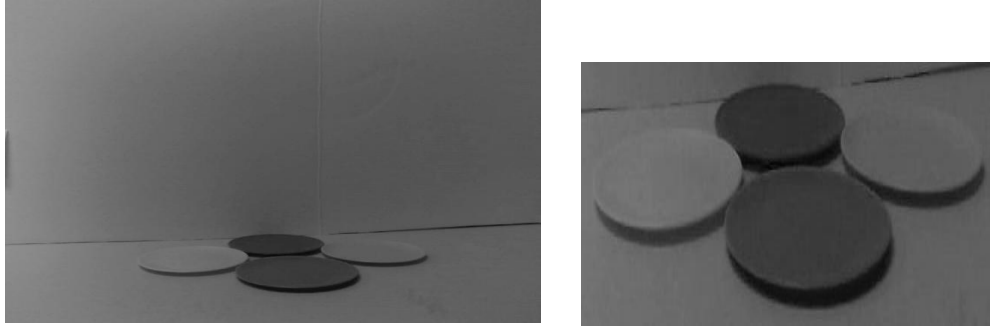
We theorize that these purely human ratings are indeed scene clutter measures and will be comparable to the clutter measures we develop in our algorithm.

## **CHAPTER 3**

### **EXPERIMENTS AND RESULTS**

#### **3.1 Real Images**

We captured twenty-seven real kitchen images, shown in Appendix A, with a Logitech QuickCam® Orbit AF webcam with a 2.0 Megapixel sensor. Then, we augmented the images after we captured them to crop out the background. This process was executed to ensure compatibility with the feature congestion condition used as the basis for the algorithm developed in the work by Rosenholtz et al. [3]. Their description states “As more items are added to a display, populating a greater volume of feature space, there is less room in feature space to add new salient items [3]”. Therefore, the more open space a scene has, the less cluttered it is considered. So, in order to avoid the issue of low feature congestion values, and to bring the values into perspective, we cut out the open background such that each image focuses on all the items present within the scene instead of a few items and a large open background. Figure 11 shows an example of an image for which we cropped out its background to focus on the items in the scene.



a.) Original Image

b.) Cropped Image

**Figure 11:** Augmentation of images to focus on items in scene

### 3.2 Human Subject Test and Human Survey Response

We developed a survey with the kitchen images to categorize each scene in one of five categories of clutter ranging from ‘Not cluttered’ to ‘Extremely cluttered’, where 1 denotes ‘Not cluttered’ and 5 denotes ‘Extremely cluttered’. This survey was given to three individuals to rate them separately on the basis of how cluttered the scenes are in general. We also gave the survey to one individual to rate on the basis of how cluttered each scene is, while generally considering how easy each object within the scene is to recognize. Table 7 shows the response of surveys where the three-individual survey responses are denoted by A, B, and C, and the single-individual survey response is denoted ‘singular’. The standard deviation among the three-individual is denoted by  $\delta(A,B,C)$ , while the standard deviation among all the responses, including the singular response is denoted  $\delta(A,B,C,Singular)$ .

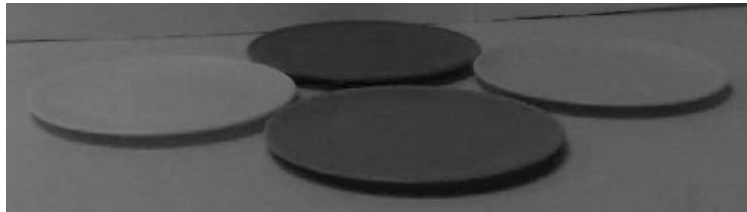
**Table 7:** Survey Responses and Standard Deviation of Responses

| <b>Images</b> | <b>Singular</b> | <b>A</b> | <b>B</b> | <b>C</b> | <b>Average(A,B,C)</b> | <b><math>\delta</math>(A,B,C)</b> | <b>Average(A, B,C, Singular)</b> | <b><math>\delta</math>(A,B,C, Singular)</b> |
|---------------|-----------------|----------|----------|----------|-----------------------|-----------------------------------|----------------------------------|---|
| Image 25      | 3               | 5        | 5        | 5        | 5                     | 0                                 | 5                                | 1   |
| Image 36      | 5               | 4        | 3        | 5        | 4                     | 1                                 | 4                                | 0.957                                       |
| Image 38      | 4               | 4        | 3        | 5        | 4                     | 1                                 | 4                                | 0.816                                       |
| Image 39      | 4               | 5        | 3        | 5        | 4                     | 1.155                             | 4                                | 0.957                                       |
| Image 40      | 1               | 3        | 2        | 4        | 3                     | 1                                 | 3                                | 1.291                                       |
| Image 74      | 1               | 1        | 1        | 1        | 1                     | 0                                 | 1                                | 0   |
| Image 76      | 1               | 2        | 2        | 2        | 2                     | 0                                 | 2                                | 0.5   |
| Image 77      | 1               | 1        | 1        | 1        | 1                     | 0                                 | 1                                | 0   |
| Image 78      | 2               | 3        | 2        | 3        | 3                     | 0.577                             | 3                                | 0.577                                       |
| Image 79      | 2               | 3        | 2        | 4        | 3                     | 1                                 | 3                                | 0.957                                       |
| Image 80      | 2               | 3        | 2        | 5        | 3                     | 1.528                             | 3                                | 1.414                                       |
| Image 82      | 2               | 4        | 3        | 5        | 4                     | 1                                 | 4                                | 1.291                                       |
| Image 83      | 2               | 3        | 2        | 3        | 3                     | 0.577                             | 3                                | 0.577                                       |
| Image 84      | 2               | 4        | 3        | 5        | 4                     | 1                                 | 4                                | 1.291                                       |
| Image 85      | 3               | 4        | 3        | 5        | 4                     | 1                                 | 4                                | 0.957                                       |
| Image 86      | 3               | 4        | 3        | 5        | 4                     | 1                                 | 4                                | 0.957                                       |
| Image 87      | 3               | 4        | 3        | 4        | 4                     | 0.577                             | 4                                | 0.577                                       |
| Image 88      | 3               | 4        | 3        | 5        | 4                     | 1                                 | 4                                | 0.957                                       |
| Image 89      | 4               | 4        | 3        | 4        | 4                     | 0.577                             | 4                                | 0.5   |
| Image 90      | 4               | 4        | 3        | 4        | 4                     | 0.577                             | 4                                | 0.5   |
| Image 91      | 4               | 4        | 3        | 4        | 4                     | 0.577                             | 4                                | 0.5   |
| Image 92      | 5               | 5        | 2        | 5        | 4                     | 1.732                             | 4                                | 1.5   |
| Image 93      | 5               | 4        | 1        | 4        | 3                     | 1.732                             | 4                                | 1.732                                       |
| Image 94      | 4               | 5        | 2        | 5        | 4                     | 1.732                             | 4                                | 1.414                                       |
| Image 95      | 5               | 5        | 2        | 5        | 4                     | 1.732                             | 4                                | 1.5   |

**Table 7 continued**

|          |   |   |   |   |   |       |   |       |
|----------|---|---|---|---|---|-------|---|-------|
| Image 98 | 5 | 5 | 3 | 5 | 4 | 1.155 | 5 | 1     |
| Image 99 | 3 | 2 | 1 | 2 | 2 | 0.577 | 2 | 0.816 |

Images 74 and 77 show the smallest standard deviation of 0 which means everyone agrees that these images are not cluttered, and Image 93 shows the largest deviations across the responses, meaning that there is more disagreement about how to categorize the image. These images (Images 74, 77, and 93) are shown in Figures 12, 13, and 14.



**Figure 12:** Image 74; all human subjects categorize as 'Not cluttered'



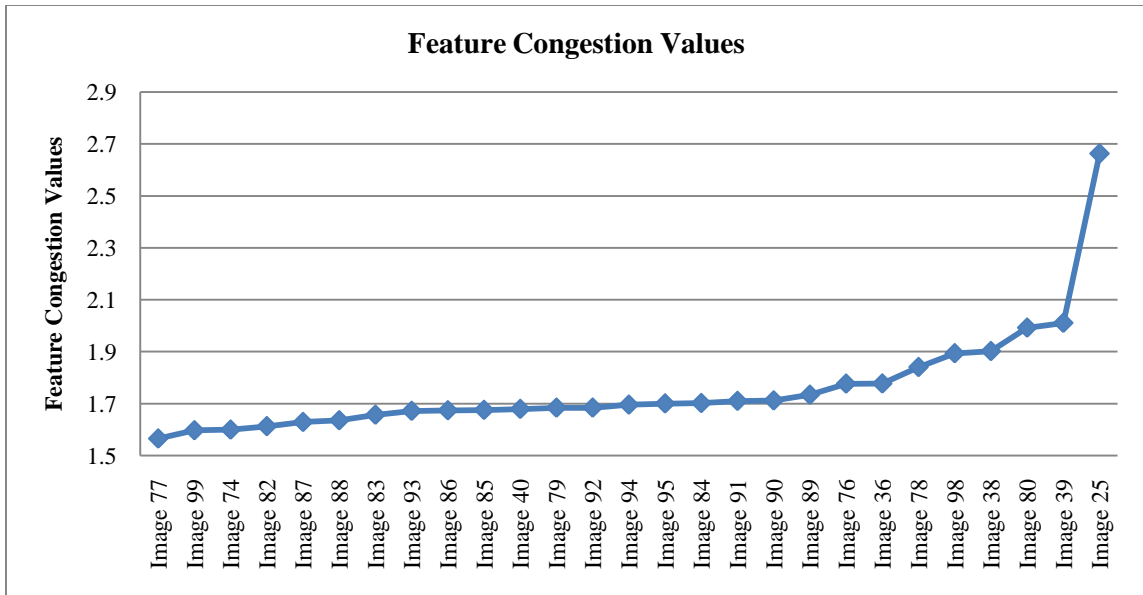
**Figure 13:** Image 77; all human subjects categorize as 'Not cluttered'



**Figure 14:** Image 93; no agreement between human subjects

### **3.3 Feature Congestion Measure of Clutter**

The Feature Congestion (FC) algorithm takes an image as input and outputs the feature congestion value. Since the algorithm measures how easy it will be to add a salient item to an existing display image, the higher the FC value, the more cluttered the input scene. We ran the algorithm for the twenty-seven kitchen scenes. Figure 15 shows FC values for the scenes in increasing order with the images on the horizontal axis and the Feature Congestion values on the vertical axis.



**Figure 15:** Feature Congestion Values for each kitchen scene image

Images 77 and 25 which have the smallest and largest FC value are shown below in Figures 16 and 17 respectively.



**Figure 16:** Image 25 with FC value of 2.6626



**Figure 17:** Image 77 with FC of 1.565

In order to categorize the FC values into one of the five categories of clutter, we used the FC values corresponding with the images that have the least standard deviation among the human survey responses within each category to determine the range of FC values that belong in each category. Thus, we used the images that the human subjects agree on the most to set the boundary conditions of the FC values. For example, all the subjects agree that Images 74 and 77 belong in category 1, and the FC values for these images are 1.5993 and 1.565, therefore, the range of the FC values that are categorized as ‘not cluttered’ are 1.565 to 1.5993. The complete categorizations of the FC values are shown in Table 8.

**Table 8:** Initial categorization of FC values based on maximum agreement among human subjects

| <b>Category</b> | <b>Feature Congestion Values</b> |
|-----------------|----------------------------------|
| 1               | 1.565 - 1.5993                   |
| 2               | 1.5968 – 1.7764                  |
| 3               | 1.6288 – 1.8403                  |
| 4               | 1.7099 – 1.7337                  |
| 5               | 1.684 – 2.6626                   |

There were some overlaps after the categorization was done, but we averaged all the categories together such that the images that fall in more than one category were placed in only one after such adjustment. The final categorizations along with corresponding values are shown in Table 9.

**Table 9:** Feature Congestion values and corresponding categorization

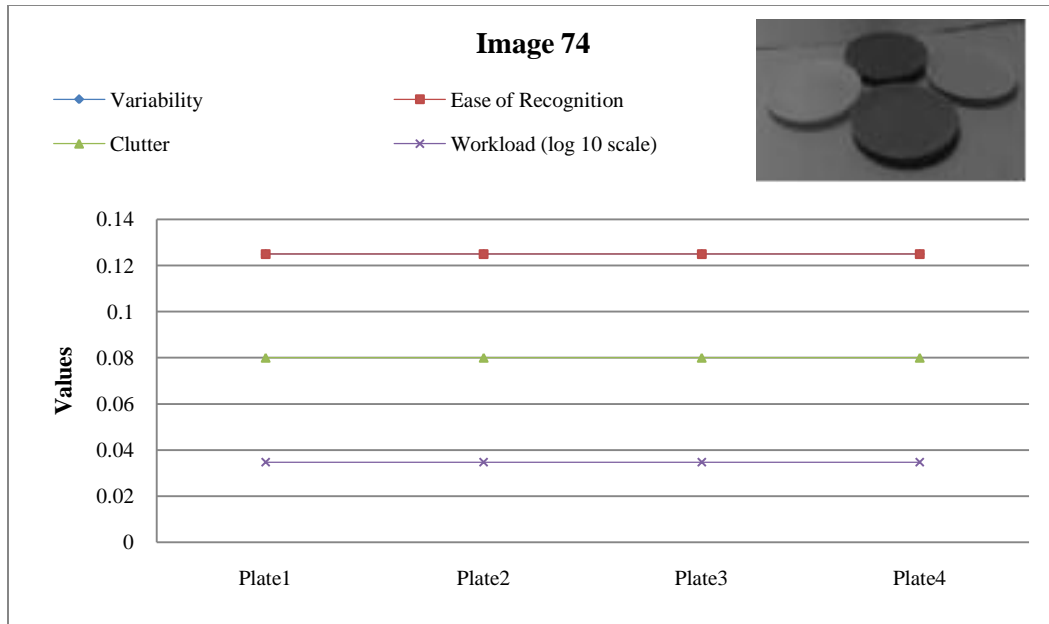
| <b>Images</b> | <b>FC values</b> | <b>FC Categories</b> |
|---------------|------------------|----------------------|
| Image 25      | 2.6626           | 5                    |
| Image 36      | 1.777            | 4                    |
| Image 38      | 1.9018           | 5                    |
| Image 39      | 2.0105           | 5                    |
| Image 40      | 1.679            | 3                    |
| Image 74      | 1.5993           | 2                    |
| Image 76      | 1.7764           | 3                    |
| Image 77      | 1.565            | 1                    |
| Image 78      | 1.8403           | 4                    |
| Image 79      | 1.6839           | 3                    |
| Image 80      | 1.9921           | 5                    |
| Image 82      | 1.6126           | 2                    |
| Image 83      | 1.6567           | 3                    |
| Image 84      | 1.7017           | 3                    |
| Image 85      | 1.6752           | 3                    |
| Image 86      | 1.6736           | 3                    |
| Image 87      | 1.6288           | 3                    |
| Image 88      | 1.6353           | 3                    |
| Image 89      | 1.7337           | 4                    |
| Image 90      | 1.7117           | 4                    |
| Image 91      | 1.7099           | 4                    |
| Image 92      | 1.684            | 3                    |
| Image 93      | 1.6716           | 3                    |
| Image 94      | 1.6954           | 3                    |
| Image 95      | 1.7001           | 3                    |
| Image 98      | 1.8935           | 5                    |

**Table 9 continued**

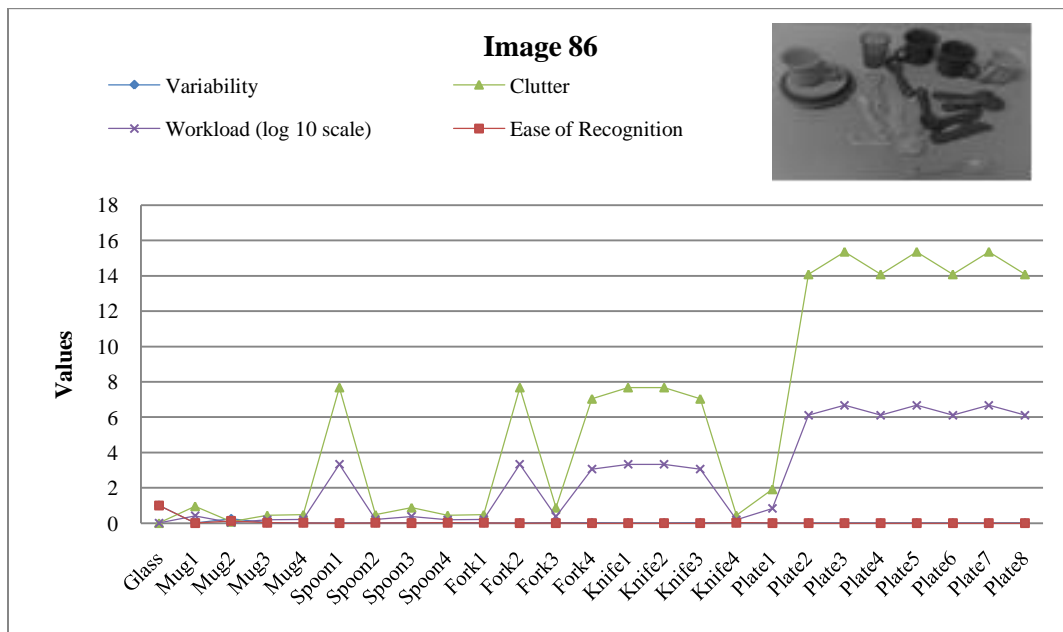
|          |        |   |
|----------|--------|---|
| Image 99 | 1.5968 | 2 |
|----------|--------|---|

### **3.4 Workload Algorithm**

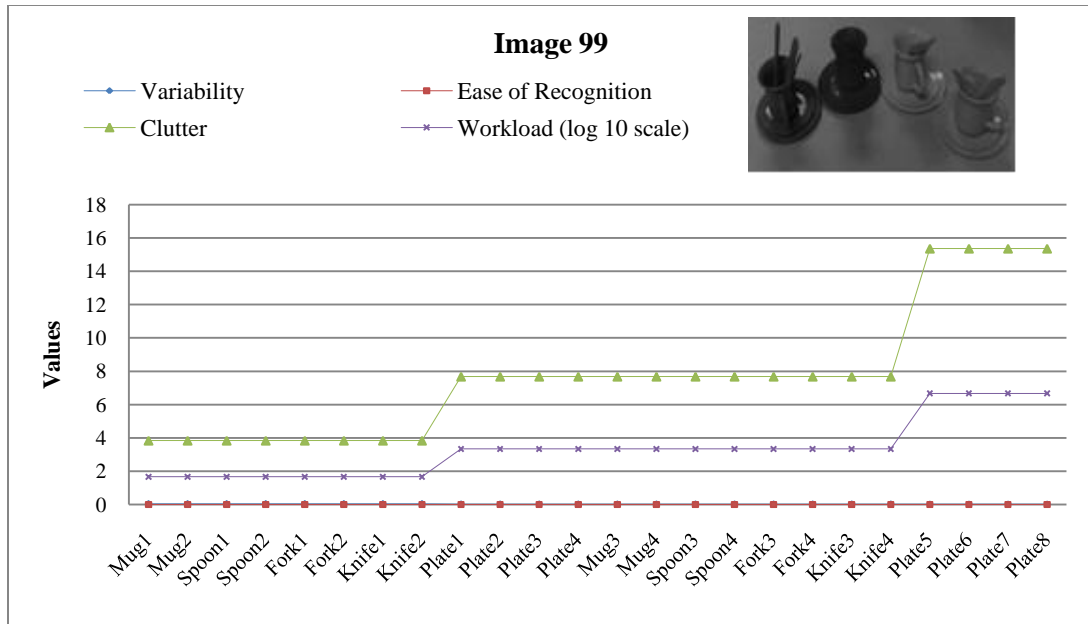
We developed an algorithm to measure four factors of recognition, namely variability, ease of recognition, clutter, and workload. We calculated the values of each of the factors for each of the objects in each scene to obtain the associated workload for recognition of each object. Since there is a high amount of total number of objects from the twenty-seven images, we have constructed the graphs in Figures 18 through 20 to show how the variability, ease of recognition, clutter, and workload (in log 10 scale) changes for the objects in three images, Images 74, 86 and 99. These images were chosen to illustrate how each of the factors contributing to recognition vary for each of the objects in each image scene. Every one of the objects in Image 74 has low workload value classified as ‘not cluttered’, and every one of the objects in Image 99 has high workload value classified as ‘extremely cluttered’. There is a wide variety of workload values for the objects in Image 86 ranging from ‘not cluttered’ to ‘extremely cluttered’.



**Figure 18:** Illustration of the Factors of Recognition for each object in Kitchen Scene (Image 74)



**Figure 19:** Illustration of the Factors of Recognition for each object in Kitchen Scene (Image 86)



**Figure 20:** Illustration of the Factors of Recognition for each object in Kitchen Scene (Image 99)

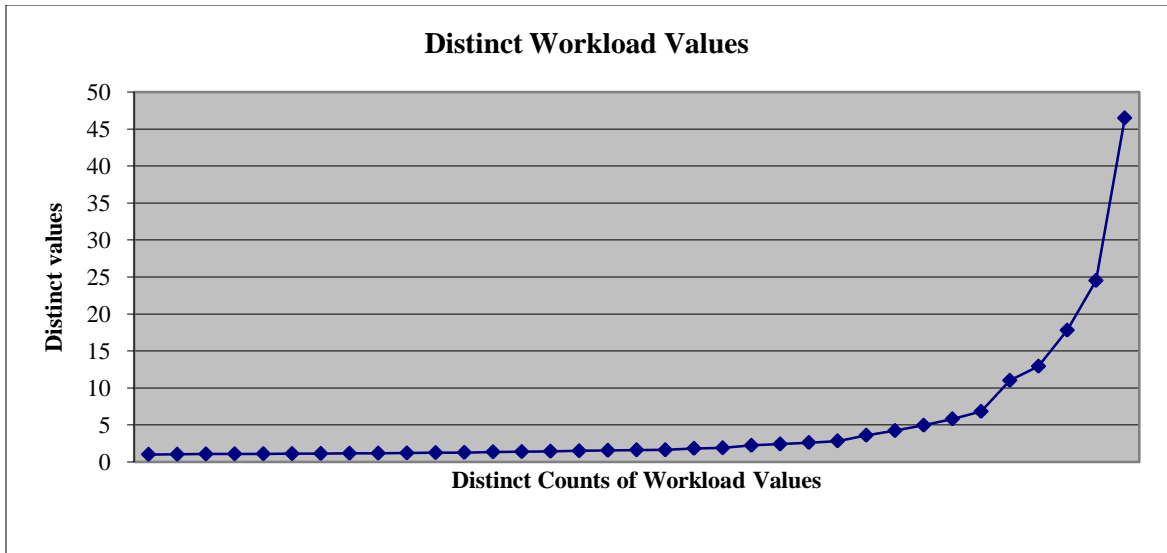
From the graphs in the above figures, it is clear that workload increases with increasing clutter and decreasing ease of recognition; the more an object is affected, i.e. cluttered, the harder it is to recognize. The complete data that shows variability, ease of recognition, clutter, and workload obtained for all the objects can be found in Appendix B.

## 3.5 Results

Our aim is to show that a more accurate measure of clutter in an environment where a robot and human work closely together to achieve proper recognition of items in their surrounding, is one that assesses how each object is affected. This will help the robot to notify its human counterpart about the workload associated with recognizing each object. We wanted to show that the number of feedback questions (NOQ) asked by a human counterpart correlates with workload; low NOQ correlate with low workload values and high NOQ with high workload. We also wanted to show that the Feature Congestion measure developed in [3] is indeed a scene measure based on the averaging of the values across feature type and across display and compare it with the scene clutter and object clutter measures we developed in our algorithm. Finally, we wanted to show that human measures of clutter as obtained by our survey responses are truly scene measures and compare them with our scene clutter measure and object clutter measure.

### 3.5.1 Feedback Questions and Workload

We classified the workload values into one of five categories of clutter discussed in earlier sections to maintain continuance between the human measures, FC measures, and the workload algorithm measures. Workload is an exponentially increasing measure, and as a result, we have values ranging from 1 to  $10^6$ , but we consider any workload value that is greater than 24 as extremely cluttered. We therefore truncate the values and graphed them as shown in Figure 21, then classified the remaining values to determine the ranges for the remaining four categories of clutter.



**Figure 21:** Truncated Distinct Workload Values<sup>4</sup>

The knee section of the curve in Figure 21, from count 30 to 33 which corresponds to workload range of 6.82 to 17.81 is classified as level 4, ‘very, very cluttered’. The lower section, below the knee, is subdivided into three to determine the range of the three lower categories. Values above the knee are classified as ‘extremely cluttered’. The complete classifications of the workload values we obtained for each of the objects in each scene is outlined in Table 10.

---

<sup>4</sup> Horizontal axis represents all workload values calculated with respect to all objects in all images. Therefore, graph points are associated with multiple objects/images.

**Table 10:** Classification of Workload Values into five different levels

| <b>Category</b> | <b>Workload Range</b> |
|-----------------|-----------------------|
| 1               | 0 – 1.6239            |
| 2               | 1.8221 – 2.8292       |
| 3               | 3.5966 – 5.8124       |
| 4               | 6.821 – 17.8143       |
| 5               | 24.5325 & above       |

Our human subject denoted that the number of feedback questions range from 1 to 12 for distinguishable objects and assigned a maximum number of 50 for non-distinguishable objects. We classified these numbers into five categories seen in Table 11. The lowest level of workload, which corresponds to the lowest category of clutter ('not cluttered'), is specified by a maximum of two feedback questions which are the shape and color inquiries. All other levels are dependent on how many times the images are segmented to achieve location placement. Level 2 consists of three to four feedback questions, level 3 has five to six, level 4 has seven to eight, and level 5, the highest level of workload, consists of nine to fifty feedback questions.

**Table 11:** Categorization of Number of Feedback Questions

| Category | Number of Questions |
|----------|---------------------|
| 1        | 1 – 2               |
| 2        | 3 – 4               |
| 3        | 5 – 6               |
| 4        | 7 – 8               |
| 5        | 9 – 12, 50          |

We compare the categories of the NOQs and workload for each object (507 objects), and found that there is a high correlation among workloads and number of questions. We found that there is a 96.67% match among the first category of both workload and NOQ, and 70.93% match among the fifth category of both measures. The result is outlined in Table 12.

**Table 12:** Percentage Match between Workload and Number of Questions

| Category | Percentage Match between NOQ and Workload |
|----------|---|
| 1        | 96.67%                                    |
| 2        | 39.29%                                    |
| 3        | 10.81%                                    |
| 4        | 8.6%                                      |
| 5        | 70.93%                                    |

The percentage match in the middle categories is low because classification is hard to accomplish for these. When the third and fourth categories are combined into one, the percent match increases from 10.81% and 8.6% individually to a combined 16.92%. However, it is clear that low workload correlates to low number of feedback questions, and high workload correlates to high number of feedback questions.

### 3.5.2 Feature Congestion Measure

To compare the feature congestion measure to the clutter measure from our algorithm, we averaged the clutter values for all the objects within each scene and compared this to the FC measure and the scene clutter measure we developed in Equation 3. For example, the clutter measured for a particular kitchen scene image, Image 74, which has four objects (all plates), shows that each object (plate) within the scene has a clutter measure of 0.08. If this is averaged for all objects within the scene, the averaged object clutter is 0.08. We did this for all the objects that we have, and then classified them into five categories by dividing the maximum value by 5. The classifications for the averaged object clutter and scene clutter (Equation 3) are specified in Table 13.

**Table 13:** Categorizations of Averaged object clutter and Scene clutter

| Category | Averaged Object Clutter | Scene Clutter   |
|----------|-------------------------|-----------------|
| 1        | 0 – 0.6134              | 0 – 0.0128      |
| 2        | 0.6135 – 1.2269         | 0.0129 – 0.0256 |
| 3        | 1.227 – 1.8403          | 0.0257 – 0.0384 |
| 4        | 1.8404 – 2.4538         | 0.0385 – 0.0512 |
| 5        | 2.4539 – 3.0672         | 0.0513 – 0.064  |

The scene clutter and averaged object clutter measure as obtained from our algorithm, as well as the FC clutter measure are combined in Table 14 for comparison.

**Table 14:** Averaged object clutter and Scene clutter for each kitchen scene image

| Images   | FC Measure | Averaged Object Clutter | Scene Clutter |
|----------|------------|-------------------------|---------------|
| Image 25 | 5          | 1                       | 1             |
| Image 36 | 4          | 1                       | 2             |
| Image 38 | 5          | 1                       | 1             |
| Image 39 | 5          | 1                       | 2             |
| Image 40 | 3          | 1                       | 2             |
| Image 74 | 2          | 1                       | 2             |
| Image 76 | 3          | 1                       | 2             |
| Image 77 | 1          | 1                       | 1             |
| Image 78 | 4          | 1                       | 1             |
| Image 79 | 3          | 2                       | 1             |
| Image 80 | 5          | 2                       | 1             |

**Table 14 continued**

|          |   |   |   |
|----------|---|---|---|
| Image 82 | 2 | 2 | 1 |
| Image 83 | 3 | 1 | 1 |
| Image 84 | 3 | 1 | 1 |
| Image 85 | 3 | 3 | 1 |
| Image 86 | 3 | 4 | 1 |
| Image 87 | 3 | 4 | 1 |
| Image 88 | 3 | 4 | 1 |
| Image 89 | 4 | 4 | 1 |
| Image 90 | 4 | 4 | 1 |
| Image 91 | 4 | 4 | 2 |
| Image 92 | 3 | 4 | 3 |
| Image 93 | 3 | 5 | 4 |
| Image 94 | 3 | 5 | 4 |
| Image 95 | 3 | 4 | 3 |
| Image 98 | 5 | 4 | 2 |
| Image 99 | 2 | 4 | 5 |

The three measures are compared and percent matches among them are determined and expressed in Table 15. The table shows that FC is more comparable to averaged object clutter measure as it should be because in [3], the FC measure is obtained by computing the local feature covariance and combining them across scale by taking the maximum at each pixel, combining them across feature types by averaging across feature types, and finally combining across space by taking the average clutter over the entire image. Their method of determining FC is more similar to our method of determining averaged object

clutter. Both measures are averages over the entire image, while considering the total number of objects within the image.

**Table 15:** Comparisons among FC and Averaged Object clutter and Scene clutter

|           | <b>Averaged Object Clutter</b> | <b>Scene Clutter</b> |
|-----------|--------------------------------|----------------------|
| <b>FC</b> | 22.22%                         | 14.81%               |

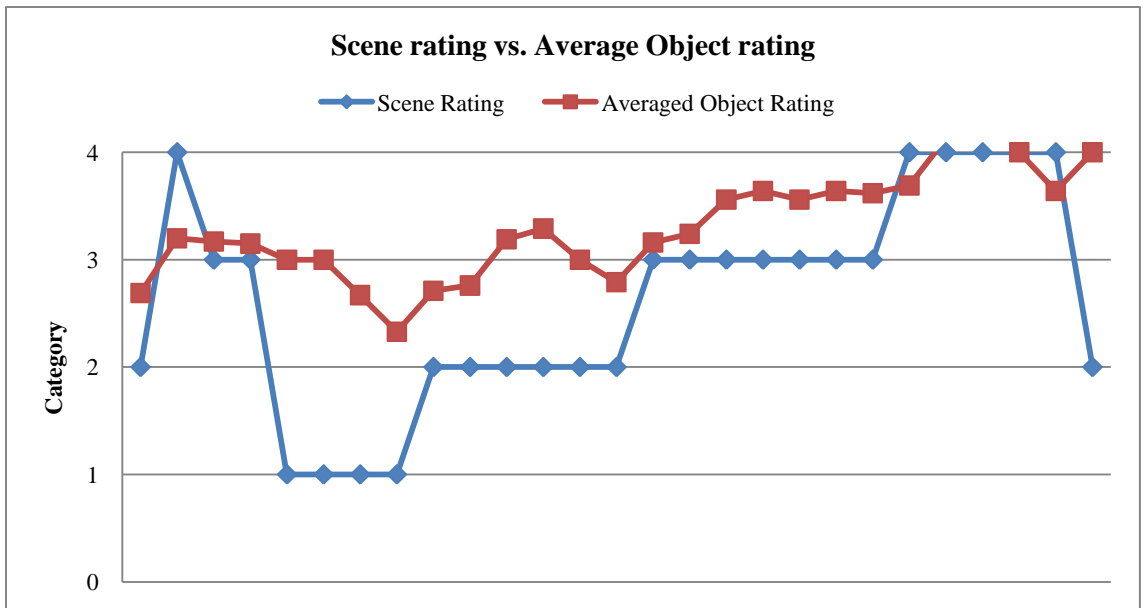
### 3.5.3 Human Measures

The human measures of clutter as determined by our survey responses and demonstrated in Table 7 are compared to the averaged object and scene clutter measures that we developed. The result of the comparison can be seen in Table 16, which shows that the human measures are more comparable to averaged object clutter which indicates that humans subconsciously average what they see. The table included the average of each of the three human subjects (A,B, and C) and the singular response.

**Table 16:** Comparisons among Human Measures, Averaged Object clutter, and Scene clutter

|   | <b>Averaged Object Clutter</b> | <b>Scene Clutter</b> |
|---|--------------------------------|----------------------|
| <b>Singular Human Scene Clutter Rating</b>        | 44.44%                         | 7.41%                |
| <b>Average Human Scene Clutter Rating (A,B,C)</b> | 40.74%                         | 11.11%               |

A human subject was asked to go through the twenty-seven kitchen scene images and categorize each object within the each scene on how cluttered it is, and also categorize each scene on how cluttered it is. The subject was asked to do this using the template illustrated in Table 6, classifying each scene and each object into one of the five categories. Figure 22 shows the comparisons between the average of the responses for all objects and for each overall display clutter. The horizontal axis denotes the different scenes assessed, and the vertical axis denotes the categories of classification. As can be seen, the subject felt that there was no scene that is extremely cluttered, thus, the vertical axis ranges from 1 to 4; ‘not cluttered’ to ‘very, very cluttered’. The figure shows that 70 to 80 percent of the time, the human subject’s ratings are within 40% of each other.



**Figure 22:** Comparisons between Averaged object rating and scene rating according to a Human subject response

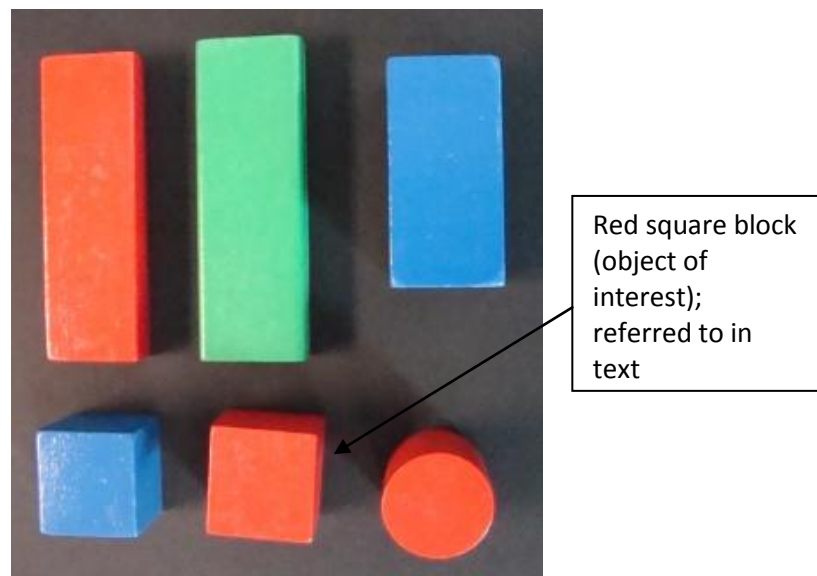
We developed this new algorithm to show that there is a need for a new metric to quantify clutter such that it considers the human’s view of clutter. This is accomplished with our algorithm which compares more to human measure of clutter than to a classical clutter measure such as the Feature Congestion measure. Our averaged object clutter measure, which is the average of all object clutter, obtained by Equation 4, within a particular scene is more comparable to the human measures that we obtained from the survey responses than to the FC measures. Table 17 confirms this theory.

**Table 17:** Comparisons among Averaged Object clutter, FC, and Human Measures of clutter

|  | <b>FC</b> | <b>Singular Human Rating</b> | <b>Averaged (A,B&amp;C) Human Ratings</b> |
|--|-----------|------------------------------|---|
| <b>Averaged Object Clutter (Our Algorithm)</b> | 22.22%    | 44.44%                       | 40.74%                                    |

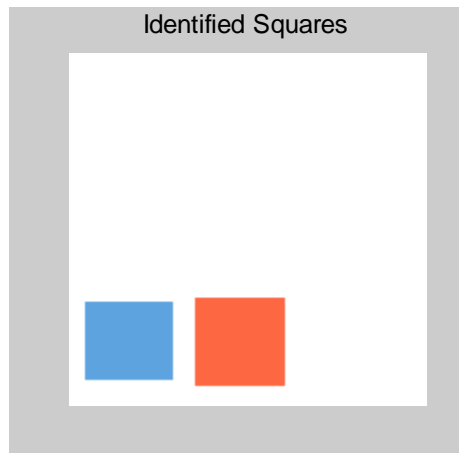
### 3.6 Implementation of Interactive Text Response (ITR)

We implemented the methodology we developed in this work by combining shape recognition (Appendix C) with color recognition (Appendix D) algorithms. We were able to demonstrate that it is possible to go through the ITR steps (as outlined in Figure 2) to identify an object of interest based on three features of shape, color, and placement. The kitchen scenes we have been working with were simplified to colored wooden blocks as seen in Figure 23.

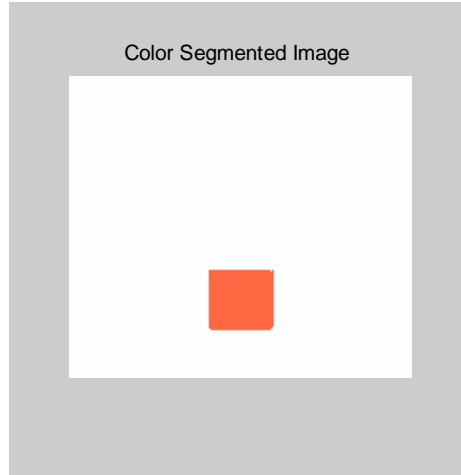


**Figure 23:** A sample of the wooden blocks used in implementation

A simple kitchen scene is constructed such that a circular block represents a plate, while a square one represents a spoon, and a rectangular one represents a cup. When a human subject asks for a particular object such as the red square block in Figure 23, the system begins by searching for squares in the image scene and then, of the recognized squares, it searches for red, and prints out the recognized object at the location at which it was found. We did not execute placement recognition for the vision implementation, since this information can be simply extracted by obtaining the centroid of the recognized object. Figures 24 and 25 show the result of the shape recognition and color recognition processes used to recognize the red block as highlighted by the arrow in Figure 23.



**Figure 24:** Identified squares as the result of Shape Recognition



**Figure 25:** Identified red square as the result of Color Recognition

The red square in Figure 23 is obviously not affected by clutter, and this is evident in the ITR process as the shape and color recognition tools are sufficient to recognize the object of interest. No placement recognition is needed as the object of interest was recognized by its two features of shape and color. This corresponds to two feedback questions, which according to our clutter classification in Table 11, is in the lowest category of clutter, ‘not cluttered’. The image in Figure 23 has three colors; there are three red objects, two blue, and one green. There are also three shapes which are three rectangles, two squares, and one circle. Combining the feature information present in Figure 23, we calculate the workload associated with the red block in the image in Equations 6-9.

$$V(\text{red square}) = \frac{1}{s_{\text{red square}} \times h_{\text{red square}}} = \frac{1}{2 \times 3} = \frac{1}{6} \quad (\text{Equation 6})$$

$$E(\text{red square}) = \frac{V(\text{red square})}{2^\alpha} = \frac{1/6}{2^0} = \frac{1}{6} \quad (\text{Equation 7})$$

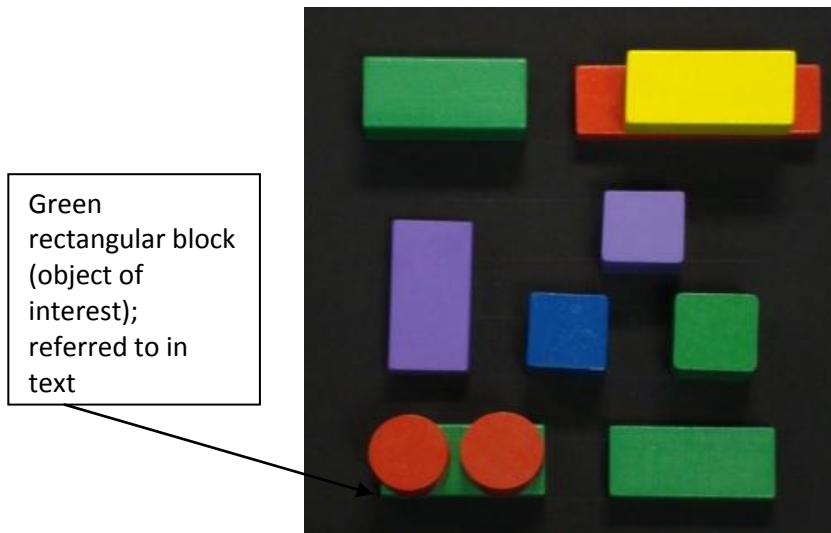
$$C(\text{red square}) = \frac{1}{100} \times \left[ \sum E(\text{red square}) \right]^{-1} = \frac{1}{100} \times \left[ \frac{1}{6} \right]^{-1}$$

$$= 0.06 \quad (\text{Equation 8})$$

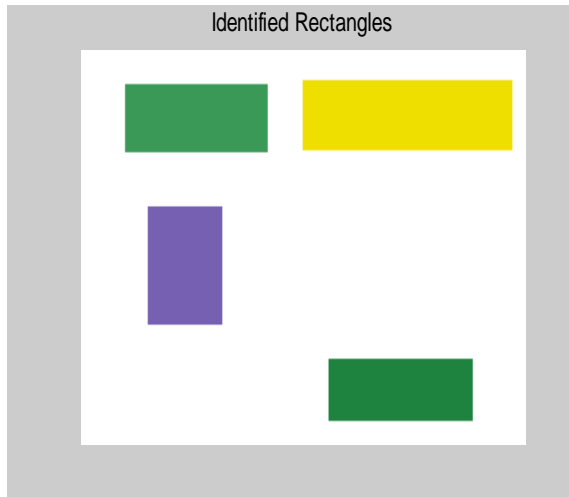
$$W(\text{red square}) = e^{C(\text{red square})} = e^{0.06} = 1.06 \quad (\text{Equation 9})$$

The clutter value associated with the square block is 0.06 which according to our classification of clutter in Table 13 is considered ‘not cluttered’. The workload associated with the object is 1.06 which according to our classification of workload in Table 10 is considered ‘not cluttered’. This categorization agrees with the classification of the number of feedback questions for the same object, which confirms the correlation between number of feedback questions and workload.

We also implemented the ITR process for the image scene in Figure 26 to recognize the hidden green rectangular block. The results of the shape and color recognition processes are shown in Figures 27 and 28.

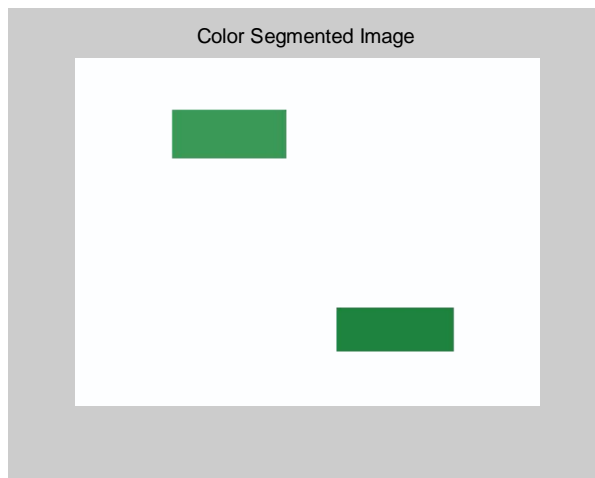


**Figure 26:** Image scene of wooden blocks used in ITR implementation



**Figure 27:** Identified rectangles as the result of Shape Recognition

Only the completely visible rectangles are recognized by the shape recognition process; the object of interest was not recognized. The result of the color recognition process is shown in Figure 28 below.



**Figure 28:** Identified green rectangles as the result of Color Recognition

The ITR process to recognize the object of interest, the hidden green block in Figure 26, yields the recognition of the wrong objects, the other green blocks. This, therefore, corresponds to a maximum number of feedback questions (50), which according to the clutter classification in Table 11, is in the highest category of clutter, ‘extremely cluttered’. The image in Figure 26 has five colors; there are two purple blocks, one yellow block, four green blocks, one blue block, and three red blocks. There are also three shapes; there are two circular blocks, six rectangular blocks, and three square blocks. Combining these feature information, we calculate (Equations 10-13) the workload associated with the recognition of the hidden green rectangular block.

$$V(\text{green rectangle}) = \frac{1}{s_{\text{green rectangle}} \times h_{\text{green rectangle}}} = \frac{1}{6 \times 4}$$

$$= \frac{1}{24} \quad (\text{Equation 10})$$

$$E(\text{green rectangle}) = \frac{V(\text{green rectangle})}{2^\alpha} = \frac{1/24}{2^4} = \frac{1}{384} \quad (\text{Equation 11})$$

$$C(\text{green rectangle}) = \frac{1}{100} \times \left[ \sum E(\text{green rectangle}) \right]^{-1} = \frac{1}{100} \times \left[ \frac{1}{384} \right]^{-1}$$

$$= 3.84 \quad (\text{Equation 12})$$

$$W(\text{green rectangle}) = e^{C(\text{green rectangle})} = e^{3.84} = 46.53 \quad (\text{Equation 13})$$

The clutter value associated with the hidden green rectangular block is 46.53 which according to the classification of workload in Table 10 is considered ‘extremely cluttered’. This categorization agrees with the number of feedback questions for the same object, which again confirms the correlation between high number of feedback questions and workload.

## CHAPTER 4

### CONCLUSION AND FUTURE WORK

In this work, we have developed an Interactive Text Response (ITR) methodology for continuous interaction between human and robot in order to accomplish tasks together. We have also developed a workload metric that can enable a robot to autonomously assess its environment and report to its human counterpart about the complexity of recognition associated with each object in the environment. This work has applicability in the field of HRI where a robot is serving a human user and requires the human's input to adequately accomplish its tasks.

The main limitation associated with the ITR methodology is inefficiency in relevance and order of inquiries. The first limitation correlates to its inability to recognize and omit unnecessary questions in order to make the system more efficient. For instance, if the objects recognized by shape are the same recognized by color, then there is no need to ask the human counterpart about the color of the object of interest. The second issue involves the system's inability to adapt its inquiries to the image scene that it is currently focusing on. For instance, it may be more relevant to inquire about color first instead of always adhering to the same order of shape, color, and placement inquiries.

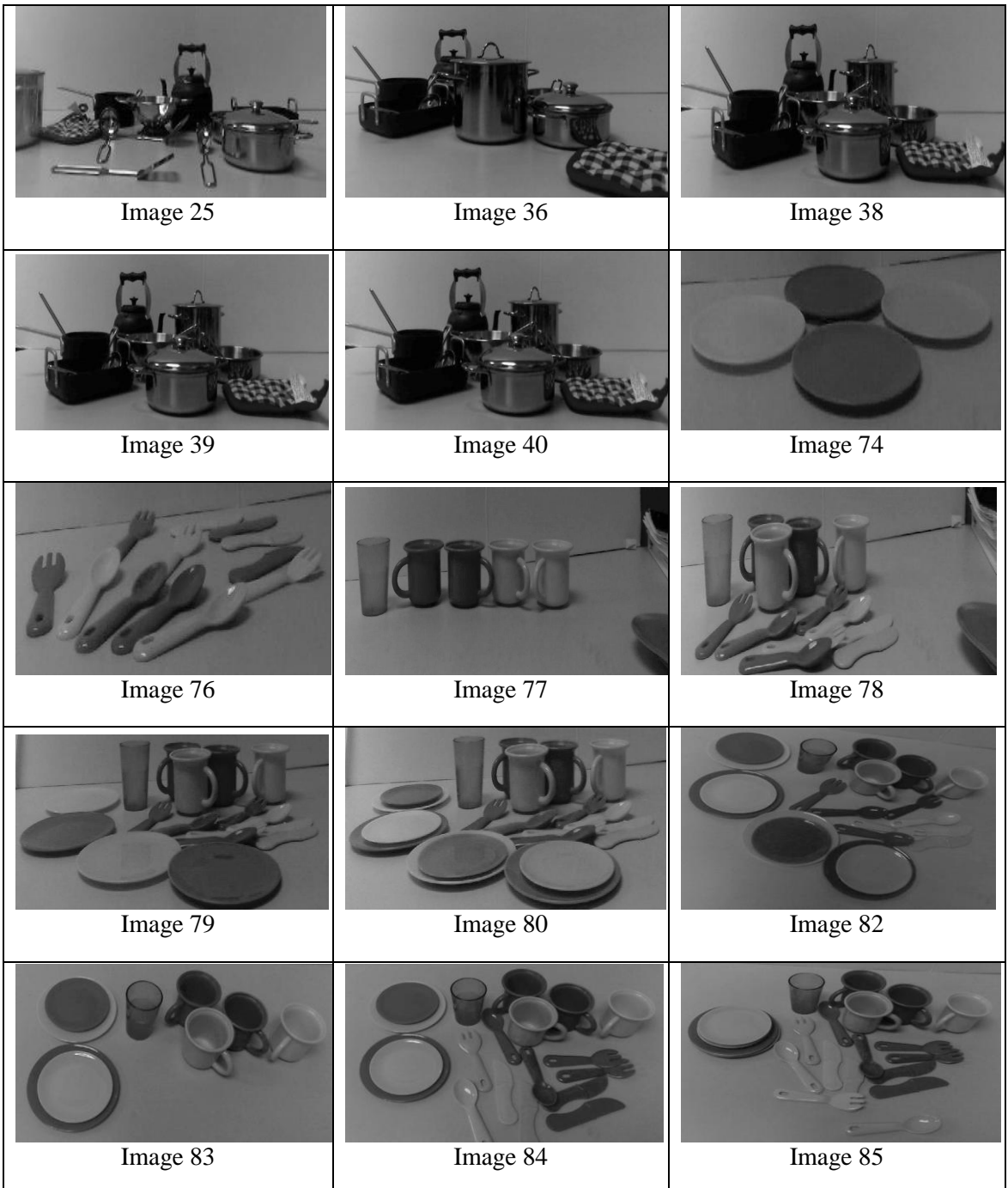
In this work, we did not address the issue of how to interpret the meaning of the quantitative workload values with respect to difficulty of recognition, such that the human counterpart is able to immediately understand the level of difficulty associated with the recognition of each object. When the robot calculates the workload value of an













object of interest, it should be equipped to inform the user about the correlation of the workload value to difficulty in recognition. This can be done using verbal feedback and/or visual feedbacks such as color-coded feedback, or a lighted indicator.

A future work for this research could focus on implementing a more efficient ITR system which better addresses the limitations of the present system. A future work could also address the user-interface problem of interpreting the workload values.

## APPENDIX A

### REAL IMAGES OF KITCHEN SCENES



|   |  |   |
|---|--|---|
|    |    |    |
| <p>Image 86</p>   | <p>Image 87</p>  | <p>Image 88</p>   |
|    |    |    |
| <p>Image 89</p>   | <p>Image 90</p>  | <p>Image 91</p>   |
|    |    |    |
| <p>Image 92</p>   | <p>Image 93</p>  | <p>Image 94</p>   |
|  |  |  |
| <p>Image 95</p>   | <p>Image 98</p>  | <p>Image 99</p>   |

**APPENDIX B**

**VARIABILITY, EASE OF RECOGNITION, CLUTTER, AND WORKLOAD**

**DATA FOR ALL OBJECTS**

| <b>Images &amp; Objects</b> | <b>NOQ</b> | <b><math>\alpha</math></b> | <b>V</b> | <b>E</b> | <b>C</b> | <b>W</b> |
|-----------------------------|------------|----------------------------|----------|----------|----------|----------|
| Image 25                    |            |                            |          |          |          |          |
| Napkin                      | 2          | 1                          | 1        | 0.5      | 0.02     | 1.0202   |
| Pan1                        | 7          | 4                          | 0.33     | 0.020625 | 0.484848 | 1.6239   |
| Pan2                        | 4          | 4                          | 0.33333  | 0.020833 | 0.48     | 1.6161   |
| Pan3                        | 11         | 4                          | 0.11111  | 0.006944 | 1.44     | 4.2207   |
| Kettle                      | 4          | 4                          | 0.33333  | 0.020833 | 0.48     | 1.6161   |
| Spatula1                    | 9          | 1                          | 0.11111  | 0.055556 | 0.18     | 1.1972   |
| Spatula2                    | 1          | 0                          | 0.11111  | 0.111111 | 0.09     | 1.0942   |
| Tall pot                    | 6          | 1                          | 0.11111  | 0.055556 | 0.18     | 1.1972   |
| Spoon1                      | 50         | 4                          | 0.11111  | 0.006944 | 1.44     | 4.2207   |
| Spoon2                      | 9          | 1                          | 0.11111  | 0.055556 | 0.18     | 1.1972   |
| Spoon3                      | 1          | 0                          | 0.11111  | 0.111111 | 0.09     | 1.0942   |
| Pot                         | 7          | 4                          | 0.11111  | 0.006944 | 1.44     | 4.2207   |
| Strainer                    | 7          | 1                          | 0.11111  | 0.055556 | 0.18     | 1.1972   |
| Image 36                    |            |                            |          |          |          |          |
| Pan1                        | 5          | 4                          | 0.33     | 0.020625 | 0.484848 | 1.6239   |
| Pan2                        | 4          | 4                          | 0.33333  | 0.020833 | 0.48     | 1.6161   |
| Pan3                        | 11         | 4                          | 0.16667  | 0.010417 | 0.96     | 2.6117   |
| Kettle                      | 5          | 4                          | 0.33     | 0.020625 | 0.484848 | 1.6239   |
| Napkin                      | 2          | 4                          | 1        | 0.0625   | 0.16     | 1.1735   |
| Spoon1                      | 50         | 4                          | 0.16667  | 0.010417 | 0.96     | 2.6117   |
| Spoon2                      | 5          | 1                          | 0.16667  | 0.083333 | 0.12     | 1.1275   |
| Strainer                    | 50         | 4                          | 0.16667  | 0.010417 | 0.96     | 2.6117   |
| Tall pot                    | 7          | 4                          | 0.16667  | 0.010417 | 0.96     | 2.6117   |
| Pot                         | 7          | 4                          | 0.16667  | 0.010417 | 0.96     | 2.6117   |
| Image 38                    |            |                            |          |          |          |          |
| Pan1                        | 8          | 4                          | 0.33     | 0.020625 | 0.484848 | 1.6239   |
| Pan2                        | 8          | 4                          | 0.33333  | 0.020833 | 0.48     | 1.6161   |
| Pan3                        | 9          | 4                          | 0.125    | 0.007813 | 1.28     | 3.5966   |
| Kettle                      | 4          | 4                          | 0.33333  | 0.020833 | 0.48     | 1.6161   |
| Napkin                      | 1          | 0                          | 1        | 1        | 0.01     | 1.0101   |
| Spoon1                      | 50         | 4                          | 0.125    | 0.007813 | 1.28     | 3.5966   |

|          |    |   |         |          |      |        |
|----------|----|---|---------|----------|------|--------|
| Spoon2   | 5  | 1 | 0.125   | 0.0625   | 0.16 | 1.1735 |
| Spoon3   | 50 | 4 | 0.125   | 0.007813 | 1.28 | 3.5966 |
| Strainer | 10 | 4 | 0.125   | 0.007813 | 1.28 | 3.5966 |
| Spatula  | 50 | 4 | 0.125   | 0.007813 | 1.28 | 3.5966 |
| Pot      | 7  | 4 | 0.125   | 0.007813 | 1.28 | 3.5966 |
| Tall pot | 8  | 4 | 0.125   | 0.007813 | 1.28 | 3.5966 |
| Image 39 |    |   |         |          |      |        |
| Pan1     | 4  | 4 | 0.33333 | 0.020833 | 0.48 | 1.6161 |
| Pan2     | 3  | 1 | 0.33333 | 0.166667 | 0.06 | 1.0618 |
| Pan3     | 10 | 4 | 0.1     | 0.00625  | 1.6  | 4.9530 |
| Kettle   | 4  | 4 | 0.33333 | 0.020833 | 0.48 | 1.6161 |
| Spoon1   | 50 | 4 | 0.1     | 0.00625  | 1.6  | 4.9530 |
| Spoon2   | 50 | 4 | 0.1     | 0.00625  | 1.6  | 4.9530 |
| Spoon3   | 50 | 4 | 0.1     | 0.00625  | 1.6  | 4.9530 |
| Lid1     | 5  | 4 | 0.1     | 0.00625  | 1.6  | 4.9530 |
| Lid2     | 8  | 4 | 0.1     | 0.00625  | 1.6  | 4.9530 |
| Strainer | 10 | 4 | 0.1     | 0.00625  | 1.6  | 4.9530 |
| Pot      | 7  | 4 | 0.1     | 0.00625  | 1.6  | 4.9530 |
| Spatula  | 50 | 4 | 0.1     | 0.00625  | 1.6  | 4.9530 |
| Tall pot | 10 | 4 | 0.1     | 0.00625  | 1.6  | 4.9530 |
| Image 40 |    |   |         |          |      |        |
| Pan1     | 5  | 1 | 0.33333 | 0.166667 | 0.06 | 1.0618 |
| Pan2     | 5  | 1 | 0.33333 | 0.166667 | 0.06 | 1.0618 |
| Pan3     | 5  | 4 | 0.125   | 0.007813 | 1.28 | 3.5966 |
| Kettle   | 3  | 1 | 0.33333 | 0.166667 | 0.06 | 1.0618 |
| Spoon1   | 5  | 1 | 0.125   | 0.0625   | 0.16 | 1.1735 |
| Spoon2   | 6  | 4 | 0.125   | 0.007813 | 1.28 | 3.5966 |
| Spoon3   | 50 | 4 | 0.125   | 0.007813 | 1.28 | 3.5966 |
| Tall pot | 8  | 4 | 0.125   | 0.007813 | 1.28 | 3.5966 |
| Spatula  | 50 | 4 | 0.125   | 0.007813 | 1.28 | 3.5966 |
| Strainer | 9  | 4 | 0.125   | 0.007813 | 1.28 | 3.5966 |
| Pot      | 7  | 4 | 0.125   | 0.007813 | 1.28 | 3.5966 |
| Image 74 |    |   |         |          |      |        |
| Plate1   | 3  | 0 | 0.125   | 0.125    | 0.08 | 1.0833 |
| Plate2   | 4  | 0 | 0.125   | 0.125    | 0.08 | 1.0833 |
| Plate3   | 3  | 0 | 0.125   | 0.125    | 0.08 | 1.0833 |
| Plate4   | 4  | 0 | 0.125   | 0.125    | 0.08 | 1.0833 |
| Image 76 |    |   |         |          |      |        |
| Spoon1   | 2  | 0 | 0.125   | 0.125    | 0.08 | 1.0833 |
| Spoon2   | 50 | 1 | 0.125   | 0.0625   | 0.16 | 1.1735 |

|            |    |   |         |          |      |         |
|------------|----|---|---------|----------|------|---------|
| Spoon3     | 2  | 0 | 0.04167 | 0.041667 | 0.24 | 1.2712  |
| Spoon4     | 50 | 1 | 0.04167 | 0.020833 | 0.48 | 1.6161  |
| Knife1     | 50 | 1 | 0.0625  | 0.03125  | 0.32 | 1.3771  |
| Knife2     | 2  | 0 | 0.0625  | 0.0625   | 0.16 | 1.1735  |
| Knife3     | 50 | 1 | 0.04167 | 0.020833 | 0.48 | 1.6161  |
| Knife4     | 2  | 0 | 0.04167 | 0.041667 | 0.24 | 1.2712  |
| Fork1      | 50 | 1 | 0.0625  | 0.03125  | 0.32 | 1.3771  |
| Fork2      | 1  | 0 | 0.0625  | 0.0625   | 0.16 | 1.1735  |
| Fork3      | 50 | 4 | 0.04167 | 0.002604 | 3.84 | 46.5255 |
| Fork4      | 50 | 1 | 0.04167 | 0.020833 | 0.48 | 1.6161  |
| Image 77   |    |   |         |          |      |         |
| Glass      | 1  | 0 | 1       | 1        | 0.01 | 1.0101  |
| Mug1       | 5  | 0 | 0.125   | 0.125    | 0.08 | 1.0833  |
| Mug2       | 5  | 0 | 0.125   | 0.125    | 0.08 | 1.0833  |
| Mug3       | 50 | 1 | 0.125   | 0.0625   | 0.16 | 1.1735  |
| Mug4       | 2  | 0 | 0.125   | 0.125    | 0.08 | 1.0833  |
| Half Plate | 2  | 1 | 1       | 0.5      | 0.02 | 1.0202  |
| Image 78   |    |   |         |          |      |         |
| Glass      | 1  | 0 | 1       | 1        | 0.01 | 1.0101  |
| Half Plate | 3  | 1 | 0.2     | 0.1      | 0.1  | 1.1052  |
| Mug1       | 50 | 1 | 0.125   | 0.0625   | 0.16 | 1.1735  |
| Mug2       | 50 | 1 | 0.125   | 0.0625   | 0.16 | 1.1735  |
| Mug3       | 1  | 0 | 0.04167 | 0.041667 | 0.24 | 1.2712  |
| Mug4       | 50 | 1 | 0.04167 | 0.020833 | 0.48 | 1.6161  |
| Spoon1     | 8  | 0 | 0.06667 | 0.066667 | 0.15 | 1.1618  |
| Spoon2     | 6  | 0 | 0.06667 | 0.066667 | 0.15 | 1.1618  |
| Spoon3     | 50 | 4 | 0.05556 | 0.003472 | 2.88 | 17.8143 |
| Fork1      | 50 | 1 | 0.06667 | 0.033333 | 0.3  | 1.3499  |
| Fork2      | 1  | 0 | 0.06667 | 0.066667 | 0.15 | 1.1618  |
| Fork3      | 50 | 4 | 0.05556 | 0.003472 | 2.88 | 17.8143 |
| Knife1     | 50 | 4 | 0.08333 | 0.005208 | 1.92 | 6.8210  |
| Knife2     | 50 | 1 | 0.08333 | 0.041667 | 0.24 | 1.2712  |
| Image 79   |    |   |         |          |      |         |
| Glass      | 1  | 0 | 1       | 1        | 0.01 | 1.0101  |
| Plate1     | 2  | 0 | 0.03125 | 0.03125  | 0.32 | 1.3771  |
| Plate2     | 2  | 0 | 0.04167 | 0.041667 | 0.24 | 1.2712  |
| Plate3     | 50 | 1 | 0.03125 | 0.015625 | 0.64 | 1.8965  |
| Plate4     | 50 | 1 | 0.04167 | 0.020833 | 0.48 | 1.6161  |
| Mug1       | 50 | 1 | 0.125   | 0.0625   | 0.16 | 1.1735  |

|          |    |   |         |          |       |             |
|----------|----|---|---------|----------|-------|-------------|
| Mug2     | 50 | 1 | 0.125   | 0.0625   | 0.16  | 1.1735      |
| Mug3     | 1  | 0 | 0.03125 | 0.03125  | 0.32  | 1.3771      |
| Mug4     | 50 | 1 | 0.03125 | 0.015625 | 0.64  | 1.8965      |
| Fork1    | 7  | 1 | 0.05556 | 0.027778 | 0.36  | 1.4333      |
| Fork2    | 7  | 1 | 0.05556 | 0.027778 | 0.36  | 1.4333      |
| Fork3    | 7  | 4 | 0.04167 | 0.002604 | 3.84  | 46.5255     |
| Spoon1   | 8  | 1 | 0.05556 | 0.027778 | 0.36  | 1.4333      |
| Spoon2   | 10 | 1 | 0.05556 | 0.027778 | 0.36  | 1.4333      |
| Spoon3   | 8  | 4 | 0.04167 | 0.002604 | 3.84  | 46.5255     |
| Knife1   | 12 | 4 | 0.0625  | 0.003906 | 2.56  | 12.9358     |
| Knife2   | 10 | 1 | 0.0625  | 0.03125  | 0.32  | 1.3771      |
| Image 80 |    |   |         |          |       |             |
| Glass    | 1  | 0 | 1       | 1        | 0.01  | 1.0101      |
| Mug1     | 50 | 1 | 0.125   | 0.0625   | 0.16  | 1.1735      |
| Mug2     | 50 | 1 | 0.125   | 0.0625   | 0.16  | 1.1735      |
| Mug3     | 1  | 0 | 0.025   | 0.025    | 0.4   | 1.4918      |
| Mug4     | 50 | 1 | 0.025   | 0.0125   | 0.8   | 2.2255      |
| Plate1   | 4  | 0 | 0.01563 | 0.015625 | 0.64  | 1.8965      |
| Plate2   | 50 | 4 | 0.0125  | 0.000781 | 12.8  | 362217.4496 |
| Plate3   | 3  | 0 | 0.0125  | 0.0125   | 0.8   | 2.2255      |
| Plate4   | 50 | 4 | 0.01563 | 0.000977 | 10.24 | 28001.1259  |
| Plate5   | 3  | 0 | 0.01563 | 0.015625 | 0.64  | 1.8965      |
| Plate6   | 50 | 4 | 0.0125  | 0.000781 | 12.8  | 362217.4496 |
| Plate7   | 3  | 0 | 0.0125  | 0.0125   | 0.8   | 2.2255      |
| Plate8   | 50 | 4 | 0.01563 | 0.000977 | 10.24 | 28001.1259  |
| Fork1    | 10 | 1 | 0.04167 | 0.020833 | 0.48  | 1.6161      |
| Fork2    | 7  | 1 | 0.04167 | 0.020833 | 0.48  | 1.6161      |
| Fork3    | 50 | 4 | 0.03333 | 0.002083 | 4.8   | 121.5104    |
| Spoon1   | 8  | 1 | 0.04167 | 0.020833 | 0.48  | 1.6161      |
| Spoon2   | 9  | 4 | 0.04167 | 0.002604 | 3.84  | 46.5255     |
| Spoon3   | 8  | 4 | 0.03333 | 0.002083 | 4.8   | 121.5104    |
| Knife1   | 50 | 4 | 0.05    | 0.003125 | 3.2   | 24.5325     |
| Knife2   | 50 | 1 | 0.05    | 0.025    | 0.4   | 1.4918      |
| Image 82 |    |   |         |          |       |             |
| Glass    | 1  | 0 | 1       | 1        | 0.01  | 1.0101      |
| Plate1   | 4  | 0 | 0.025   | 0.025    | 0.4   | 1.4918      |
| Plate2   | 50 | 1 | 0.0125  | 0.00625  | 1.6   | 4.9530      |
| Plate3   | 3  | 0 | 0.0125  | 0.0125   | 0.8   | 2.2255      |
| Plate4   | 50 | 1 | 0.025   | 0.0125   | 0.8   | 2.2255      |
| Plate5   | 4  | 0 | 0.025   | 0.025    | 0.4   | 1.4918      |

|          |    |   |         |          |      |             |
|----------|----|---|---------|----------|------|-------------|
| Plate6   | 50 | 4 | 0.0125  | 0.000781 | 12.8 | 362217.4496 |
| Plate7   | 3  | 0 | 0.0125  | 0.0125   | 0.8  | 2.2255      |
| Plate8   | 50 | 1 | 0.025   | 0.0125   | 0.8  | 2.2255      |
| Mug1     | 50 | 1 | 0.05    | 0.025    | 0.4  | 1.4918      |
| Mug2     | 50 | 1 | 0.05    | 0.025    | 0.4  | 1.4918      |
| Mug3     | 5  | 0 | 0.025   | 0.025    | 0.4  | 1.4918      |
| Mug4     | 5  | 0 | 0.025   | 0.025    | 0.4  | 1.4918      |
| Fork1    | 5  | 4 | 0.06667 | 0.004167 | 2.4  | 11.0232     |
| Fork2    | 7  | 4 | 0.06667 | 0.004167 | 2.4  | 11.0232     |
| Fork3    | 8  | 4 | 0.03333 | 0.002083 | 4.8  | 121.5104    |
| Spoon1   | 7  | 4 | 0.06667 | 0.004167 | 2.4  | 11.0232     |
| Spoon2   | 3  | 4 | 0.06667 | 0.004167 | 2.4  | 11.0232     |
| Spoon3   | 10 | 1 | 0.03333 | 0.016667 | 0.6  | 1.8221      |
| Knife1   | 12 | 4 | 0.05    | 0.003125 | 3.2  | 24.5325     |
| Knife2   | 8  | 4 | 0.05    | 0.003125 | 3.2  | 24.5325     |
| Image 83 |    |   |         |          |      |             |
| Glass    | 1  | 0 | 1       | 1        | 0.01 | 1.0101      |
| Plate1   | 2  | 0 | 0.125   | 0.125    | 0.08 | 1.0833      |
| Plate2   | 50 | 1 | 0.0625  | 0.03125  | 0.32 | 1.3771      |
| Plate3   | 2  | 0 | 0.0625  | 0.0625   | 0.16 | 1.1735      |
| Plate4   | 50 | 1 | 0.125   | 0.0625   | 0.16 | 1.1735      |
| Mug1     | 50 | 1 | 0.125   | 0.0625   | 0.16 | 1.1735      |
| Mug2     | 50 | 1 | 0.125   | 0.0625   | 0.16 | 1.1735      |
| Mug3     | 5  | 0 | 0.0625  | 0.0625   | 0.16 | 1.1735      |
| Mug4     | 5  | 0 | 0.0625  | 0.0625   | 0.16 | 1.1735      |
| Image 84 |    |   |         |          |      |             |
| Glass    | 1  | 0 | 1       | 1        | 0.01 | 1.0101      |
| Plate1   | 2  | 0 | 0.03125 | 0.03125  | 0.32 | 1.3771      |
| Plate2   | 50 | 1 | 0.03125 | 0.015625 | 0.64 | 1.8965      |
| Plate3   | 2  | 0 | 0.03125 | 0.03125  | 0.32 | 1.3771      |
| Plate4   | 50 | 1 | 0.03125 | 0.015625 | 0.64 | 1.8965      |
| Mug1     | 50 | 4 | 0.125   | 0.007813 | 1.28 | 3.5966      |
| Mug2     | 50 | 1 | 0.125   | 0.0625   | 0.16 | 1.1735      |
| Mug3     | 5  | 0 | 0.03125 | 0.03125  | 0.32 | 1.3771      |
| Mug4     | 5  | 0 | 0.03125 | 0.03125  | 0.32 | 1.3771      |
| Fork1    | 50 | 1 | 0.04167 | 0.020833 | 0.48 | 1.6161      |
| Fork2    | 50 | 4 | 0.04167 | 0.002604 | 3.84 | 46.5255     |
| Fork3    | 1  | 0 | 0.04167 | 0.041667 | 0.24 | 1.2712      |
| Knife1   | 50 | 4 | 0.03125 | 0.001953 | 5.12 | 167.3354    |
| Knife2   | 50 | 4 | 0.03125 | 0.001953 | 5.12 | 167.3354    |

|          |    |   |         |          |       |              |
|----------|----|---|---------|----------|-------|--------------|
| Knife3   | 2  | 0 | 0.03125 | 0.03125  | 0.32  | 1.3771       |
| Knife4   | 2  | 0 | 0.03125 | 0.03125  | 0.32  | 1.3771       |
| Spoon1   | 5  | 1 | 0.04167 | 0.020833 | 0.48  | 1.6161       |
| Spoon2   | 7  | 1 | 0.04167 | 0.020833 | 0.48  | 1.6161       |
| Spoon3   | 7  | 1 | 0.04167 | 0.020833 | 0.48  | 1.6161       |
| Image 85 |    |   |         |          |       |              |
| Glass    | 1  | 0 | 1       | 1        | 0.01  | 1.0101       |
| Mug1     | 50 | 4 | 0.125   | 0.007813 | 1.28  | 3.5966       |
| Mug2     | 50 | 1 | 0.125   | 0.0625   | 0.16  | 1.1735       |
| Mug3     | 5  | 0 | 0.02083 | 0.020833 | 0.48  | 1.6161       |
| Mug4     | 5  | 0 | 0.02083 | 0.020833 | 0.48  | 1.6161       |
| Spoon1   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Spoon2   | 1  | 0 | 0.02083 | 0.020833 | 0.48  | 1.6161       |
| Spoon3   | 50 | 1 | 0.025   | 0.0125   | 0.8   | 2.2255       |
| Spoon4   | 50 | 4 | 0.025   | 0.001563 | 6.4   | 601.8450     |
| Fork1    | 1  | 0 | 0.02083 | 0.020833 | 0.48  | 1.6161       |
| Fork2    | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Fork3    | 50 | 1 | 0.025   | 0.0125   | 0.8   | 2.2255       |
| Fork4    | 50 | 4 | 0.025   | 0.001563 | 6.4   | 601.8450     |
| Knife1   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife2   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife3   | 50 | 4 | 0.025   | 0.001563 | 6.4   | 601.8450     |
| Knife4   | 5  | 0 | 0.025   | 0.025    | 0.4   | 1.4918       |
| Plate1   | 1  | 0 | 0.01042 | 0.010417 | 0.96  | 2.6117       |
| Plate2   | 50 | 4 | 0.0125  | 0.000781 | 12.8  | 362217.4496  |
| Plate3   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate4   | 50 | 4 | 0.0125  | 0.000781 | 12.8  | 362217.4496  |
| Plate5   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate6   | 50 | 4 | 0.0125  | 0.000781 | 12.8  | 362217.4496  |
| Plate7   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate8   | 50 | 4 | 0.0125  | 0.000781 | 12.8  | 362217.4496  |
| Image 86 |    |   |         |          |       |              |
| Glass    | 1  | 0 | 1       | 1        | 0.01  | 1.0101       |
| Mug1     | 50 | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Mug2     | 50 | 1 | 0.25    | 0.125    | 0.08  | 1.0833       |
| Mug3     | 2  | 0 | 0.02273 | 0.022727 | 0.44  | 1.5527       |
| Mug4     | 2  | 0 | 0.02083 | 0.020833 | 0.48  | 1.6161       |
| Spoon1   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Spoon2   | 2  | 0 | 0.02083 | 0.020833 | 0.48  | 1.6161       |
| Spoon3   | 50 | 1 | 0.02273 | 0.011364 | 0.88  | 2.4109       |

|          |    |   |         |          |       |              |
|----------|----|---|---------|----------|-------|--------------|
| Spoon4   | 2  | 0 | 0.02273 | 0.022727 | 0.44  | 1.5527       |
| Fork1    | 1  | 0 | 0.02083 | 0.020833 | 0.48  | 1.6161       |
| Fork2    | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Fork3    | 50 | 1 | 0.02273 | 0.011364 | 0.88  | 2.4109       |
| Fork4    | 50 | 4 | 0.02273 | 0.00142  | 7.04  | 1141.3876    |
| Knife1   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife2   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife3   | 50 | 4 | 0.02273 | 0.00142  | 7.04  | 1141.3876    |
| Knife4   | 1  | 0 | 0.02273 | 0.022727 | 0.44  | 1.5527       |
| Plate1   | 8  | 1 | 0.01042 | 0.005208 | 1.92  | 6.8210       |
| Plate2   | 50 | 4 | 0.01136 | 0.00071  | 14.08 | 1302765.6686 |
| Plate3   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate4   | 50 | 4 | 0.01136 | 0.00071  | 14.08 | 1302765.6686 |
| Plate5   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate6   | 50 | 4 | 0.01136 | 0.00071  | 14.08 | 1302765.6686 |
| Plate7   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate8   | 50 | 4 | 0.01136 | 0.00071  | 14.08 | 1302765.6686 |
| Image 87 |    |   |         |          |       |              |
| Glass    | 1  | 0 | 1       | 1        | 0.01  | 1.0101       |
| Mug1     | 50 | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Mug2     | 2  | 0 | 0.02083 | 0.020833 | 0.48  | 1.6161       |
| Mug3     | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Mug4     | 2  | 0 | 0.02083 | 0.020833 | 0.48  | 1.6161       |
| Spoon1   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Spoon2   | 7  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Spoon3   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Spoon4   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Fork1    | 1  | 0 | 0.02083 | 0.020833 | 0.48  | 1.6161       |
| Fork2    | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Fork3    | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Fork4    | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife1   | 8  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Knife2   | 8  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife3   | 8  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife4   | 10 | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Plate1   | 8  | 1 | 0.01042 | 0.005208 | 1.92  | 6.8210       |
| Plate2   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate3   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate4   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate5   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |

|          |    |   |         |          |       |              |
|----------|----|---|---------|----------|-------|--------------|
| Plate6   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate7   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate8   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Image 88 |    |   |         |          |       |              |
| Glass    | 1  | 0 | 1       | 1        | 0.01  | 1.0101       |
| Mug1     | 8  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Mug2     | 7  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Mug3     | 7  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Mug4     | 6  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Spoon1   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Spoon2   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Spoon3   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Spoon4   | 7  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Fork1    | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Fork2    | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Fork3    | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Fork4    | 1  | 0 | 0.02083 | 0.020833 | 0.48  | 1.6161       |
| Knife1   | 8  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Knife2   | 10 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife3   | 9  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife4   | 8  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Plate1   | 8  | 1 | 0.01042 | 0.005208 | 1.92  | 6.8210       |
| Plate2   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate3   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate4   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate5   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate6   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate7   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate8   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Image 89 |    |   |         |          |       |              |
| Glass    | 1  | 0 | 1       | 1        | 0.01  | 1.0101       |
| Mug1     | 8  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Mug2     | 7  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Mug3     | 7  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Mug4     | 7  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Spoon1   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Spoon2   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Spoon3   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Spoon4   | 1  | 0 | 0.02083 | 0.020833 | 0.48  | 1.6161       |
| Fork1    | 1  | 0 | 0.02083 | 0.020833 | 0.48  | 1.6161       |

|          |    |   |         |          |       |              |
|----------|----|---|---------|----------|-------|--------------|
| Fork2    | 50 | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Fork3    | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Fork4    | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife1   | 8  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Knife2   | 8  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife3   | 8  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife4   | 8  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Plate1   | 8  | 1 | 0.01042 | 0.005208 | 1.92  | 6.8210       |
| Plate2   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate3   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate4   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate5   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate6   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate7   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate8   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Image 90 |    |   |         |          |       |              |
| Glass    | 1  | 0 | 1       | 1        | 0.01  | 1.0101       |
| Mug1     | 8  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Mug2     | 7  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Mug3     | 7  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Mug4     | 6  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Spoon1   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Spoon2   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Spoon3   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Spoon4   | 7  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Fork1    | 1  | 0 | 0.02083 | 0.020833 | 0.48  | 1.6161       |
| Fork2    | 50 | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Fork3    | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Fork4    | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife1   | 9  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Knife2   | 8  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife3   | 8  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife4   | 8  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117       |
| Plate1   | 8  | 1 | 0.01042 | 0.005208 | 1.92  | 6.8210       |
| Plate2   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate3   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate4   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate5   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate6   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate7   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |

|          |    |   |         |          |       |               |
|----------|----|---|---------|----------|-------|---------------|
| Plate8   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567  |
| Image 91 |    |   |         |          |       |               |
| Glass1   | 50 | 4 | 0.03846 | 0.002404 | 4.16  | 64.0715       |
| Glass2   | 1  | 0 | 0.5     | 0.5      | 0.02  | 1.0202        |
| Fork1    | 1  | 0 | 0.02083 | 0.020833 | 0.48  | 1.6161        |
| Fork2    | 50 | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117        |
| Fork3    | 50 | 4 | 0.01923 | 0.001202 | 8.32  | 4105.1600     |
| Fork4    | 50 | 4 | 0.01923 | 0.001202 | 8.32  | 4105.1600     |
| Mug1     | 5  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117        |
| Mug2     | 7  | 1 | 0.01923 | 0.009615 | 1.04  | 2.8292        |
| Mug3     | 7  | 4 | 0.01923 | 0.001202 | 8.32  | 4105.1600     |
| Mug4     | 5  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117        |
| Spoon1   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198     |
| Spoon2   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198     |
| Spoon3   | 50 | 4 | 0.01923 | 0.001202 | 8.32  | 4105.1600     |
| Spoon4   | 7  | 4 | 0.01923 | 0.001202 | 8.32  | 4105.1600     |
| Knife1   | 7  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117        |
| Knife2   | 8  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198     |
| Knife3   | 8  | 4 | 0.01923 | 0.001202 | 8.32  | 4105.1600     |
| Knife4   | 8  | 1 | 0.01923 | 0.009615 | 1.04  | 2.8292        |
| Plate1   | 8  | 1 | 0.01042 | 0.005208 | 1.92  | 6.8210        |
| Plate2   | 50 | 4 | 0.00962 | 0.000601 | 16.64 | 16852338.6935 |
| Plate3   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567  |
| Plate4   | 50 | 4 | 0.00962 | 0.000601 | 16.64 | 16852338.6935 |
| Plate5   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567  |
| Plate6   | 50 | 4 | 0.00962 | 0.000601 | 16.64 | 16852338.6935 |
| Plate7   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567  |
| Plate8   | 50 | 4 | 0.00962 | 0.000601 | 16.64 | 16852338.6935 |
| Image 92 |    |   |         |          |       |               |
| Glass1   | 7  | 4 | 0.25    | 0.015625 | 0.64  | 1.8965        |
| Glass2   | 9  | 4 | 0.25    | 0.015625 | 0.64  | 1.8965        |
| Mug1     | 8  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117        |
| Mug2     | 5  | 4 | 0.02273 | 0.00142  | 7.04  | 1141.3876     |
| Mug3     | 3  | 4 | 0.25    | 0.015625 | 0.64  | 1.8965        |
| Mug4     | 5  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198     |
| Spoon1   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198     |
| Spoon2   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198     |
| Spoon3   | 50 | 4 | 0.02273 | 0.00142  | 7.04  | 1141.3876     |
| Spoon4   | 8  | 4 | 0.02273 | 0.00142  | 7.04  | 1141.3876     |
| Fork1    | 1  | 0 | 0.02083 | 0.020833 | 0.48  | 1.6161        |

|           |    |   |         |          |       |              |
|-----------|----|---|---------|----------|-------|--------------|
| Fork2     | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Fork3     | 50 | 4 | 0.02273 | 0.00142  | 7.04  | 1141.3876    |
| Fork4     | 50 | 4 | 0.02273 | 0.00142  | 7.04  | 1141.3876    |
| Knife1    | 8  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife2    | 11 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife3    | 7  | 4 | 0.02273 | 0.00142  | 7.04  | 1141.3876    |
| Knife4    | 8  | 4 | 0.02273 | 0.00142  | 7.04  | 1141.3876    |
| Plate1    | 8  | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate2    | 50 | 4 | 0.01136 | 0.00071  | 14.08 | 1302765.6686 |
| Plate3    | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate4    | 50 | 4 | 0.01136 | 0.00071  | 14.08 | 1302765.6686 |
| Plate5    | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate6    | 50 | 4 | 0.01136 | 0.00071  | 14.08 | 1302765.6686 |
| Plate7    | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate8    | 50 | 4 | 0.01136 | 0.00071  | 14.08 | 1302765.6686 |
| Image 93  |    |   |         |          |       |              |
| Glass1    | 7  | 4 | 0.25    | 0.015625 | 0.64  | 1.8965       |
| Glass2    | 7  | 4 | 0.25    | 0.015625 | 0.64  | 1.8965       |
| Mug1      | 6  | 4 | 0.025   | 0.001563 | 6.4   | 601.8450     |
| Mug2      | 5  | 4 | 0.025   | 0.001563 | 6.4   | 601.8450     |
| Mug3      | 4  | 4 | 0.08333 | 0.005208 | 1.92  | 6.8210       |
| Mug4      | 50 | 4 | 0.02778 | 0.001736 | 5.76  | 317.3483     |
| Plate1    | 50 | 4 | 0.0125  | 0.000781 | 12.8  | 362217.4496  |
| Plate2    | 50 | 4 | 0.01389 | 0.000868 | 11.52 | 100709.9619  |
| Plate3    | 50 | 4 | 0.0125  | 0.000781 | 12.8  | 362217.4496  |
| Plate4    | 50 | 4 | 0.01389 | 0.000868 | 11.52 | 100709.9619  |
| Plate5    | 50 | 4 | 0.0125  | 0.000781 | 12.8  | 362217.4496  |
| Plate6    | 50 | 4 | 0.01389 | 0.000868 | 11.52 | 100709.9619  |
| Plate7    | 50 | 4 | 0.0125  | 0.000781 | 12.8  | 362217.4496  |
| Plate8    | 50 | 4 | 0.01389 | 0.000868 | 11.52 | 100709.9619  |
| Utensil1  | 50 | 4 | 0.03333 | 0.002083 | 4.8   | 121.5104     |
| Utensil2  | 50 | 4 | 0.03333 | 0.002083 | 4.8   | 121.5104     |
| Utensil3  | 50 | 4 | 0.01111 | 0.000694 | 14.4  | 1794074.7726 |
| Utensil4  | 50 | 4 | 0.01111 | 0.000694 | 14.4  | 1794074.7726 |
| Utensil5  | 50 | 4 | 0.01    | 0.000625 | 16    | 8886110.5205 |
| Utensil6  | 50 | 4 | 0.01    | 0.000625 | 16    | 8886110.5205 |
| Utensil7  | 50 | 4 | 0.01    | 0.000625 | 16    | 8886110.5205 |
| Utensil8  | 50 | 4 | 0.01    | 0.000625 | 16    | 8886110.5205 |
| Utensil9  | 6  | 4 | 0.01111 | 0.000694 | 14.4  | 1794074.7726 |
| Utensil10 | 7  | 4 | 0.01111 | 0.000694 | 14.4  | 1794074.7726 |

|          |    |   |         |          |       |               |
|----------|----|---|---------|----------|-------|---------------|
| Image 94 |    |   |         |          |       |               |
| Glass1   | 6  | 4 | 0.25    | 0.015625 | 0.64  | 1.8965        |
| Glass2   | 3  | 4 | 0.25    | 0.015625 | 0.64  | 1.8965        |
| Mug1     | 8  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198     |
| Mug2     | 5  | 1 | 0.02083 | 0.010417 | 0.96  | 2.6117        |
| Mug3     | 8  | 4 | 0.02273 | 0.00142  | 7.04  | 1141.3876     |
| Mug4     | 7  | 4 | 0.02273 | 0.00142  | 7.04  | 1141.3876     |
| Plate1   | 7  | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567  |
| Plate2   | 50 | 4 | 0.01136 | 0.00071  | 14.08 | 1302765.6686  |
| Plate3   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567  |
| Plate4   | 50 | 4 | 0.01136 | 0.00071  | 14.08 | 1302765.6686  |
| Plate5   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567  |
| Plate6   | 50 | 4 | 0.01136 | 0.00071  | 14.08 | 1302765.6686  |
| Plate7   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567  |
| Plate8   | 50 | 4 | 0.01136 | 0.00071  | 14.08 | 1302765.6686  |
| Fork     | 7  | 1 | 0.09091 | 0.045455 | 0.22  | 1.2461        |
| Knife    | 6  | 4 | 0.09091 | 0.005682 | 1.76  | 5.8124        |
| Utensil1 | 7  | 4 | 0.00926 | 0.000579 | 17.28 | 31960138.1039 |
| Utensil2 | 50 | 4 | 0.00926 | 0.000579 | 17.28 | 31960138.1039 |
| Utensil3 | 50 | 4 | 0.00926 | 0.000579 | 17.28 | 31960138.1039 |
| Utensil4 | 50 | 4 | 0.00926 | 0.000579 | 17.28 | 31960138.1039 |
| Utensil5 | 50 | 4 | 0.00926 | 0.000579 | 17.28 | 31960138.1039 |
| Utensil6 | 50 | 4 | 0.00926 | 0.000579 | 17.28 | 31960138.1039 |
| Utensil7 | 50 | 4 | 0.0101  | 0.000631 | 15.84 | 7572243.8881  |
| Utensil8 | 50 | 4 | 0.0101  | 0.000631 | 15.84 | 7572243.8881  |
| Utensil9 | 50 | 4 | 0.0101  | 0.000631 | 15.84 | 7572243.8881  |
| Image 95 |    |   |         |          |       |               |
| Glass1   | 3  | 4 | 0.25    | 0.015625 | 0.64  | 1.8965        |
| Glass2   | 4  | 4 | 0.25    | 0.015625 | 0.64  | 1.8965        |
| Knife1   | 8  | 4 | 0.04167 | 0.002604 | 3.84  | 46.5255       |
| Knife2   | 8  | 4 | 0.05    | 0.003125 | 3.2   | 24.5325       |
| Mug1     | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198     |
| Mug2     | 1  | 0 | 0.02083 | 0.020833 | 0.48  | 1.6161        |
| Mug3     | 50 | 4 | 0.025   | 0.001563 | 6.4   | 601.8450      |
| Mug4     | 3  | 4 | 0.25    | 0.015625 | 0.64  | 1.8965        |
| Plate1   | 8  | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567  |
| Plate2   | 50 | 4 | 0.0125  | 0.000781 | 12.8  | 362217.4496   |
| Plate3   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567  |
| Plate4   | 50 | 4 | 0.0125  | 0.000781 | 12.8  | 362217.4496   |
| Plate5   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567  |

|          |    |   |         |          |       |              |
|----------|----|---|---------|----------|-------|--------------|
| Plate6   | 50 | 4 | 0.0125  | 0.000781 | 12.8  | 362217.4496  |
| Plate7   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate8   | 50 | 4 | 0.0125  | 0.000781 | 12.8  | 362217.4496  |
| Spoon    | 7  | 4 | 0.1     | 0.00625  | 1.6   | 4.9530       |
| Fork     | 8  | 4 | 0.1     | 0.00625  | 1.6   | 4.9530       |
| Utensil1 | 7  | 4 | 0.0119  | 0.000744 | 13.44 | 686938.4673  |
| Utensil2 | 50 | 4 | 0.0119  | 0.000744 | 13.44 | 686938.4673  |
| Utensil3 | 50 | 4 | 0.0119  | 0.000744 | 13.44 | 686938.4673  |
| Utensil4 | 50 | 4 | 0.0119  | 0.000744 | 13.44 | 686938.4673  |
| Utensil5 | 50 | 4 | 0.0119  | 0.000744 | 13.44 | 686938.4673  |
| Utensil6 | 50 | 4 | 0.01429 | 0.000893 | 11.2  | 73130.4418   |
| Utensil7 | 50 | 4 | 0.01429 | 0.000893 | 11.2  | 73130.4418   |
| Image 98 |    |   |         |          |       |              |
| Glass    | 2  | 4 | 1       | 0.0625   | 0.16  | 1.1735       |
| Mug1     | 50 | 4 | 0.03125 | 0.001953 | 5.12  | 167.3354     |
| Mug2     | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Mug3     | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Mug4     | 50 | 4 | 0.0625  | 0.003906 | 2.56  | 12.9358      |
| Spoon1   | 50 | 4 | 0.03125 | 0.001953 | 5.12  | 167.3354     |
| Spoon2   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Spoon3   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Spoon4   | 50 | 4 | 0.0625  | 0.003906 | 2.56  | 12.9358      |
| Fork1    | 50 | 4 | 0.03125 | 0.001953 | 5.12  | 167.3354     |
| Fork2    | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Fork3    | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Fork4    | 50 | 4 | 0.0625  | 0.003906 | 2.56  | 12.9358      |
| Knife1   | 50 | 4 | 0.03125 | 0.001953 | 5.12  | 167.3354     |
| Knife2   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife3   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife4   | 50 | 4 | 0.0625  | 0.003906 | 2.56  | 12.9358      |
| Plate1   | 50 | 4 | 0.01563 | 0.000977 | 10.24 | 28001.1259   |
| Plate2   | 50 | 4 | 0.01563 | 0.000977 | 10.24 | 28001.1259   |
| Plate3   | 50 | 4 | 0.01563 | 0.000977 | 10.24 | 28001.1259   |
| Plate4   | 1  | 0 | 0.01563 | 0.015625 | 0.64  | 1.8965       |
| Plate5   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate6   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate7   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate8   | 50 | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |

|          |    |   |         |          |       |              |
|----------|----|---|---------|----------|-------|--------------|
| Image 99 |    |   |         |          |       |              |
| Mug1     | 50 | 4 | 0.04167 | 0.002604 | 3.84  | 46.5255      |
| Mug2     | 50 | 4 | 0.04167 | 0.002604 | 3.84  | 46.5255      |
| Mug3     | 9  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Mug4     | 7  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Spoon1   | 50 | 4 | 0.04167 | 0.002604 | 3.84  | 46.5255      |
| Spoon2   | 50 | 4 | 0.04167 | 0.002604 | 3.84  | 46.5255      |
| Spoon3   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Spoon4   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Fork1    | 50 | 4 | 0.04167 | 0.002604 | 3.84  | 46.5255      |
| Fork2    | 50 | 4 | 0.04167 | 0.002604 | 3.84  | 46.5255      |
| Fork3    | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Fork4    | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife1   | 50 | 4 | 0.04167 | 0.002604 | 3.84  | 46.5255      |
| Knife2   | 50 | 4 | 0.04167 | 0.002604 | 3.84  | 46.5255      |
| Knife3   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Knife4   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Plate1   | 10 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Plate2   | 8  | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Plate3   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Plate4   | 50 | 4 | 0.02083 | 0.001302 | 7.68  | 2164.6198    |
| Plate5   | 8  | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate6   | 8  | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate7   | 9  | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |
| Plate8   | 8  | 4 | 0.01042 | 0.000651 | 15.36 | 4685578.7567 |

## APPENDIX C

### MATLAB CODE FOR SHAPE RECOGNITION

```
%-----  
--  
% AL-AZHAR UNIVERSITY  
% FACULTY OF ENGINEERING  
% SYSTEMS & COMPUTERS ENGINEERING Department  
%-----  
--  
% Author : Ahmed Samieh Abd El-Wahab  
% Date   : 14 December 2006  
%  
% Modified by : Morenike Ajulo  
% Date      : January 12, 2010  
%-----  
--  
%% To Recognize User-specified shapes in a User-given scene  
function W = Classify_modified2(ImageFile, shape)  
  
if strcmp(shape,'square')||strcmp(shape,'SQUARE')||...  
    strcmp(shape,'Square') == 1  
    shp = 3;  
else if strcmp(shape,'rectangle')||strcmp(shape,'RECTANGLE')||...  
    strcmp(shape,'Rectangle') == 1  
    shp = 2;  
else if strcmp(shape,'circle')||strcmp(shape,'CIRCLE')||...  
    strcmp(shape,'Circle') == 1  
    shp = 1;  
else shp = 0;  
end  
end  
  
if (shp == 0)  
    error('Your spelling may be wrong. Or I may not know this  
shape. Or this shape may not be present in this scene.');
```

```
end  
  
%===Step 1: Read image Read in=====
```

```
RGB = imread(ImageFile);  
size_image = size(RGB); %Extracts the size of the original image  
area_image = size_image(1)*size_image(2); %Calculates area of the  
image  
figure(1),  
imshow(RGB),  
title('Original Image');
```

```
%=====
```

```
%===Step 2: Convert image from rgb to gray=====
```

```
%---First to hsv:  
H = rgb2hsv(RGB);
```

```

GRAY1 = H(:,:,1); figure(2), imshow(GRAY1), title('Gray Image'); %hue
channel
GRAY2 = H(:,:,2); figure(3), imshow(GRAY2); %saturation channel
GRAY3 = H(:,:,3); figure(4), imshow(GRAY3); %value channel
%-----
avggray1 = mean(GRAY1(:));
avggray2 = mean(GRAY2(:));
avggray3 = mean(GRAY3(:));
NewImg = ((GRAY1>avggray1)+(GRAY2>avggray2)+(GRAY3>avggray3))/3;
avggray = mean(NewImg(:)); %Threshold pixel value to create BLANK
%=====

%===Step 3: Create a black and white copy of input image===
BLANK = NewImg > avggray;
figure(5), imshow(BLANK), title('GRAY1 into BLANK');
%=====

%===Step 4: Remove smaller outer (white) blobs=====
labels = bwlabel(BLANK);
s = regionprops(labels,'Area');
area_values = [s.Area];

min_area = (0.03)*area_image;

BW2 = (ismember(labels,find(area_values >= min_area)));
figure(7),
imshow(BW2),
title('Cleaned Image');
%=====

%===Step 5: Invert the Black and White Image=====
Negate = ~ BW2;
figure(8),
imshow(Negate),
title('Inverted Binary Image');
%=====

%===Step 6: Remove smaller inner (white) blobs=====
labels2 = bwlabel(Negate); %Numbers each of the conjoined blobs in
image
s2 = regionprops(labels2,'Area');
area_values2 = [s2.Area];
min_area2 = (0.1)*area_image;

BW3 = (ismember(labels2,find(area_values2 >= min_area2)));
figure(9),
imshow(BW3),
title('Inverted Cleaned Image');
%=====

%===Step 7: Re-invert Black and White Image=====
BW3 = ~BW3;
[B,L] = bwboundaries(BW3, 'noholes');
%=====

```

```

%===Step 8: Determine objects properties=====
LIST = regionprops(L, 'all');
%=====

%===Step 9: Classify Shapes according to properties=====
Indexes = [];
figure(10),
imshow(IMG),
title('Results');
hold on
for i = 1 : length(LIST)

    measure_cir_dim = (LIST(i).Area)/...
        ((LIST(i).BoundingBox(3)*LIST(i).BoundingBox(4)));
    sum_of_sides = LIST(i).BoundingBox(3)+LIST(i).BoundingBox(4);
    measure_rect_dim = abs(LIST(i).Perimeter - 2*(sum_of_sides))/...
        max(LIST(i).Perimeter,sum_of_sides);
    measure_sqr_dim = abs(LIST(i).BoundingBox(4) - ...
        LIST(i).BoundingBox(3))/max(LIST(i).BoundingBox(4),...
        LIST(i).BoundingBox(3));

    if (measure_cir_dim <= 0.8)
        W_value = 1;
    elseif (measure_rect_dim < 0.25)
        if (measure_sqr_dim < 0.2)
            W_value = 3;
        else
            W_value = 2;
        end
    else
        W_value = -1; %Signifies object is neither circ., rect., nor
sqr.
    end
    W(i) = W_value;

    centroid = LIST(i).Centroid;
    round_centr = round(centroid);

    if (shp == W(i))
        switch W(i)
            case 1
                plot(centroid(1),centroid(2),'kO');
            case 2
                plot(centroid(1),centroid(2),'kX');
            case 3
                plot(centroid(1),centroid(2),'kS');
        end
        Indexes(i,1) = i; %1st col. to identify if shape was
recognized
        Indexes(i,2:4) = impixel(IMG,round_centr(1), round_centr(2));
        %2nd to 4th cols. to identify color of recognized shape
    else

```

```

        Indexes(i,1) = 0; %1st col. to identify if shape wasn't
recognized
        Indexes(i,2:4) = impixel(RGB,round_centr(1), round_centr(2));
        %2nd to 4th cols. to identify color of recognized shape
    end
end
%=====

%===Step 10: Blot out the unrecognized shapes=====
Q = uint8(255*ones(size(RGB))); %Creates a white image

%Call each Shape's function to blot out unrecognized shapes from
original
%image:
if (shp == 3)
    Squares(LIST,Q,Indexes);
else if (shp == 2)
    Rectangles(LIST,Q,Indexes);
    else if (shp == 1)
        Circles(LIST,Q,Indexes);
    end
end
end
%=====

return
end

%% If the user wants squares highlighted:
function [] = Squares(LIST,Q,Indexes)

for k = 1:length(LIST)
    val_sq = (Indexes(k,1) ~= 0);
    if val_sq == 1
        %---To get the boundary of the recognized square:
        xs = LIST(k).Centroid(1) + ...
            (-0.5*LIST(k).BoundingBox(3):0.5*LIST(k).BoundingBox(3));
        ys = LIST(k).Centroid(2) + ...
            (-0.5*LIST(k).BoundingBox(4):0.5*LIST(k).BoundingBox(4));
        %---To round the boundary (x,y) values to the nearest integer:
        xs = round(xs);
        ys = round(ys);
        %---To highlight the portion of the white image (Q) where the
        %---recognized shape is located:
        for i = 1:length(xs)
            for j = 1:length(ys)
                Q(ys(j),xs(i),1) = Indexes(k,2);
                Q(ys(j),xs(i),2) = Indexes(k,3);
                Q(ys(j),xs(i),3) = Indexes(k,4);
            end
        end
    end
end
%-----
end

```

```

end

figure(11),
image(Q);
axis off;
title('Identified Squares');
end
%% If the user wants rectangles highlighted:
function [] = Rectangles(LIST,Q,Indexes)

for k = 1:length(LIST)
    val_rect = (Indexes(k,1) ~= 0);
    if val_rect == 1
        %---To get the boundary of the recognized rectangle:
        xr = LIST(k).Centroid(1) + ...
            (-0.5*LIST(k).BoundingBox(3):0.5*LIST(k).BoundingBox(3));
        yr = LIST(k).Centroid(2) + ...
            (-0.5*LIST(k).BoundingBox(4):0.5*LIST(k).BoundingBox(4));
        %---To round the boundary (x,y) values to the nearest integer:
        xr = round(xr);
        yr = round(yr);
        %---To highlight the portion of the white image (Q) where the
        %---recognized shape is located:
        for j = 1:length(yr)
            for i = 1:length(xr)
                Q(yr(j),xr(i),1) = Indexes(k,2);
                Q(yr(j),xr(i),2) = Indexes(k,3);
                Q(yr(j),xr(i),3) = Indexes(k,4);
            end
        end
        %-----
    end
end

figure(12),
image(Q);
axis off;
title('Identified Rectangles');
end
%% If the user wants circles highlighted:

function [] = Circles(LIST,Q,Indexes)

for k = 1:length(LIST)
    val_circ = (Indexes(k) ~= 0);
    if val_circ == 1
        diameter = max(LIST(k).BoundingBox(3),LIST(k).BoundingBox(4));
        diameter = round(diameter);
        r = diameter/2;
        %---To get the boundary of the recognized circle:
        x_center = LIST(k).Centroid(1);
        y_center = LIST(k).Centroid(2);
        %---To highlight the portion of the white image (Q) where the
        %---recognized shape is located:

```

```

M = size(Q);
for y = 1:M(1)
    for x = 1:M(2)
        if sqrt((x-x_center)^2 + (y-y_center)^2) <= r
            Q(y,x,1) = Indexes(k,2);
            Q(y,x,2) = Indexes(k,3);
            Q(y,x,3) = Indexes(k,4);
        end
    end
end

end

end

figure(13),
image(Q);
axis off;
title('Identified Circles');
end

```

## APPENDIX D

### MATLAB CODE FOR COLOR RECOGNITION

```
%% Colour Segmentation - VIBGYOR Colour Segmentation
%% This function can be used for VIBGYOR Colour segmentation from the
RGB
%% Color images.
%% Function C = VIBGYORsegmentation(img)
%% input    img = Color image (The input image should be a color image)
%% Example: C = VIBGYORsegmentation(img);
%%         Posted date : 14 - 07 - 2008
%%
%%
%% Developed By : K.Kannan & Jeny Rajan
%%               Medical Imaging Research Group (MIRG), NeST,
Trivandrum.
%% Modified By : Morenike Ajulo
%%               On January 26, 2010
%%
function C = VIBGYORmodified2(img)
[row col plane] = size(img);
img = double(img);
C = zeros(row,col,plane);
if plane ~= 3
    disp('Input should be a color image');
    return;
end
GL = 255;    %maximum number of pixels at each RGB channel
%% Color Choice
clc;
disp(' ');
disp('          r / R - for Red Color');
disp('          o / O - for Orange Color');
disp('          y / Y - for Yellow Color');
disp('          g / G - for Green Color');
disp('          b / B - for Blue Color');
disp('          i / I - for Indigo Color');
disp('          v / V - for Violet Color');
color = input('\nYour Color Choice = ','s');
%%
switch color %f1=Rmax,f2=Rmin;f3=Gmax,f4=Gmin;f5=Bmax,f6=Bmin;
    case {'R','r'}
        f1 = (GL * 1);f2 = (GL * 0.545);
        f3 = (GL * 0.42);f4 = (GL * 0);
        f5 = (GL * 0.36);f6 = (GL * 0);
        C = cfilter(img,f1,f2,f3,f4,f5,f6,1,0);
    case {'O','o'}
        f1 = (GL * 1);f2 = (GL * 0.85);
        f3 = (GL * 0.84);f4 = (GL * 0.39);
        f5 = (GL * 0.57);f6 = (GL * 0);
        C = cfilter(img,f1,f2,f3,f4,f5,f6,1,0);
    case {'Y','y'}
        f1 = (GL * 1);f2 = (GL * 0.93);
        f3 = (GL * 1);f4 = (GL * 0.87);
        f5 = (GL * 0.88);f6 = (GL * 0);
```

```

        C = cfilter(img, f1, f2, f3, f4, f5, f6, 1, 1);
    case {'G', 'g'}
        f1 = (GL * 0.68); f2 = (GL * 0);
        f3 = (GL * 1); f4 = (GL * 0.39);
        f5 = (GL * 0.83); f6 = (GL * 0);
        C = cfilter(img, f1, f2, f3, f4, f5, f6, 2, 0);
    case {'B', 'b'}
        f1 = (GL * 0.69); f2 = (GL * 0);
        f3 = (GL * 0.65); f4 = (GL * 0);
        f5 = (GL * 1); f6 = (GL * 0.44);
        C = cfilter(img, f1, f2, f3, f4, f5, f6, 3, 0);
    case {'I', 'i'}
        f1 = (GL * 0.68); f2 = (GL * 0);
        f3 = (GL * 1); f4 = (GL * 0.6);
        f5 = (GL * 1); f6 = (GL * 0.6);
        C = cfilter(img, f1, f2, f3, f4, f5, f6, 1, 1);
    case {'V', 'v'}
        f1 = (GL * 0.9); f2 = (GL * 0.5);
        f3 = (GL * 0.9); f4 = (GL * 0);
        f5 = (GL * 0.98); f6 = (GL * 0.5);
        C = cfilter(img, f1, f2, f3, f4, f5, f6, 3, 1);
    otherwise
        disp('unknown method');
end
%% Display
C = uint8(C);
figure, imshow(uint8(img), []); title('Original Image');
figure, imshow(uint8(C), []); title('Color Segmented Image');

%% Function for Color Filter
function C = cfilter(img, f1, f2, f3, f4, f5, f6, m, flg)
[ row col plane ] = size(img);
C = zeros(row, col, plane);
for i = 1:row
    for j = 1:col
        if flg == 0
            if (img(i, j, 1) <= f1 && img(i, j, 1) >= f2 && ...
                img(i, j, 2) <= f3 && img(i, j, 2) >= f4 && ...
                img(i, j, 3) <= f5 && img(i, j, 3) >= f6 ...
                && img(i, j, m) == max([img(i, j, 1) img(i, j, 2)
img(i, j, 3)]))
                C(i, j, 1:3) = img(i, j, 1:3);
            else
                %---Blot out Unrecognized Colors:
                C(i, j, 1:3) = [253 254 255];
            end
        else
            if (img(i, j, 1) <= f1 && img(i, j, 1) >= f2 && ...
                img(i, j, 2) <= f3 && img(i, j, 2) >= f4 && ...
                img(i, j, 3) <= f5 && img(i, j, 3) >= f6)
                C(i, j, 1:3) = img(i, j, 1:3);
            else
                %---Blot out Unrecognized Colors:
                C(i, j, 1:3) = [253 254 255];
            end
        end
    end
end

```



## REFERENCES

1. [http://son.uth.tmc.edu/coa/FDGN\\_1/RESOURCES/ADLandIADL.pdf](http://son.uth.tmc.edu/coa/FDGN_1/RESOURCES/ADLandIADL.pdf) (Date Accessed: April 2009).
2. Dave Roos, "How Interactive Voice Response (IVR) Works." 20 February 2008. HowStuffWorks.com. <<http://communication.howstuffworks.com/interactive-voice-response.htm>> . Date Accessed: 15 April 2010.
3. Ruth Rosenholtz, Yuanzhen Li, Jonathan Mansfield, Zhenlan Jin, "Feature Congestion: A measure of display clutter." *SIGCHI 2005*, 761-770, 2005.
4. Javi F. Gorostiza, Ramon Barber, Alaa M. Khamis, Rakel Pacheco, Rafael Rivas, Ana Corrales, Elena Delgado, Miguel A. Salichs, "Multimodal Human-Robot Interaction Framework for a Personal Robot," *Robot and Human Interactive Communication, 2006. ROMAN 2006. The 15th IEEE International Symposium on*, vol., no., pp.39-44, 6-8 Sept. 2006.
5. Changyoon Lee, You-Sung Cha, Tae-Yong Kuc, "Implementation of dialogue system for intelligent service robots," *Control, Automation and Systems, 2008. ICCAS 2008. International Conference on*, vol., no., pp.2038-2042, 14-17 Oct. 2008.
6. Hong Liu, Jie Zhou, "Motion planning for Human-Robot Interaction based on stereo vision and SIFT," *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on*, vol., no., pp.830-834, 11-14 Oct. 2009.
7. Hee-Deok Yang, A-Yeon Park, Seong-Whan Lee, "Gesture Spotting and Recognition for Human-Robot Interaction," *Robotics, IEEE Transactions on*, vol.23, no.2, pp.256-270, April 2007.
8. Ming-Shaung Chang, Jung-Hua Chou, "A friendly and intelligent human-robot interface system based on human face and hand gesture," *Advanced Intelligent Mechatronics, 2009. AIM 2009. IEEE/ASME International Conference on*, vol., no., pp.1856-1861, 14-17 July 2009.
9. Guojing He, Jianqi Zhang, Delian Liu, and Honghua Chang, "Clutter metric based on the Cramer-Rao lower bound on automatic target recognition," *Appl. Opt.* **47**, pp.5534-5540, 2008 .
10. Gary J Ewing, Christopher J Woodruff, Douglas Vickers, "Effects of 'local' clutter on human target detection," *Spatial Vision*, vol. 19, issue 1, pp.37-60, 2006.
11. Paul F. Bulakowski, Robert B. Post, David Whitney, "Visuomotor crowding: the resolution of grasping in cluttered scenes," *Frontiers in behavioral neuroscience*, vol. 3, 2009.

12. Laurent Itti, Carl Gold, Christof Koch, "Visual Attention and Target Detection in Cluttered Natural Scenes," *Optical Engineering*, Vol. **40**, No. 9, pp. 1784-1793, Sep 2001.