Institut Polytechnique de Grenoble

ENSIMAG SCHOOL

Master of Science in Informatics at Grenoble

MASTER THESIS

to obtain the title of

Master of Science

of the Institut Polytechnique de Grenoble

Specialty : GRAPHICS, VISION AND ROBOTICS

Defended by

Varun Raj KOMPELLA

Image-based Detection of Semi-transparent Objects

Thesis Advisor: Peter STURM

prepared at INRIA Rhône Alpes, PERCEPTION Team

defended on June 25, 2009

Jury :

Permanent Panel Members :	Augustin LUX	-	INRIA Rhône-Alpes PRIMA Team
	Michel BURLET	-	Université Joseph Fourier
	Christine VERDIER	-	Université Joseph Fourier
Supervisor :	Peter STURM	-	INRIA Rhône-Alpes PERCEPTION Team
External Expert :	Augustin LUX	-	INRIA Rhône-Alpes PRIMA Team
Optional Responsibility :	Frank HETROY	-	INRIA Rhône-Alpes EVASION Team

Dedicated to my beloved parents

Acknowledgments

I would like to express my deep-felt gratitude towards my advisor, Professor Peter Sturm, PERCEPTION Project at INRIA-Rhône Alpes, who advised, encouraged and supported my ideas right from the beginning of my masters. My research work in the area of transparency burgeoned after I hit myself against a glass door unknowingly while appraoching a girl at Alliance Française de Bangalore. This amusing incident, although quite painful, kept me thinking about how I missed noticing the glass door. Apparently on the same day after narrating the incident to my friends Ishrat and Reetika, who later had asked me whether it is possible to detect transparent objects in an image, I promptly answered saying it is impossible, although I ended up saying that nothing is impossible. My thanks to Ishrat and Reetika as the idea got stuck in my unconscious mind. The idea took a re-birth when I first spoke to my guide, Peter Sturm, with whom I shared my motivational story and then I realized about its usefulness in the field of robotics. He was positive and encouraged me to look into the idea. The problem had two big issues, one to first come out with a computer-vision solution and next a robotic-vision version and its working implementation on a robot. Although it was quite an intimidating task because the maximum duration I had was only 5 months, I took up the challenge and along with the expert guidance from my advisor I could reach to a successful completion of all the goals I had kept during my masters.

It was an immense pleasure working as a member of the PERCEPTION project team at INRIA and would like to thank the members for providing a great time in the office with barbeque parties, coffee-break talks, etc. I am grateful to my friend Visesh Chari, who is currently carrying out his PhD in the same office, for interesting technical and non-technical discussions during several night-outs working at the office. It helped in clearing out many of my doubts in the area of computer vision. I also thank my friend Avinash Sharma who provided with lots of entertainment in the office with his humorous short anecdotes.

I also wish to show my indebtedness to Professor James Crowley, INPG, for selecting me for the masters program and with whom I had several interesting technical discussions in computer-vision and robotics. It is only after these discussions that I had found the area in which I would like to carry out research during my PhD.

Additionally, I want to thank the MOSIG, Institut Polytechnique de Grenoble professors and staff for their hard work and dedication, providing me the means to complete my masters and prepare for a career in research.

And finally, I must thank my parents and my brothers for their loving support and inspiring me to approach and achieve my goals that I had kept during the masters program. I couldn't have achieved anything without the continued love of my parents.

Image-based Detection of Semi-transparent Objects

Abstract:

Most computer and robot vision algorithms, be it for object detection, recognition, or reconstruction, are designed for opaque objects. Non-opaque objects have received less attention, although various special cases have been the subject of research efforts, especially the case of specular objects. The main objective of this thesis is to provide a seminal work in the case of semi-transparent objects, i.e. objects that are transparent but also reflect light, typically objects made of glass. They are rather omnipresent in man-made environments (especially, windows and doors). Detection of these objects provides vital information that can be used in a robot's localization and path planning. Also, several other important applications are discussed in the report. In order to achieve the detection of semi-transparent objects we developed a novel approach using a collective-reward based technique on an image captured by an uncalibrated camera. We also present a robotic-vision version of the approach in the form of a semi-transparent obstacle avoidance algorithm for a wheeled mobile robot. Several experiments were conducted over several scenarios to test the efficacy of the algorithm.

Keywords: Object detection, semi transparent, transparency, obstacle avoidance.

Contents

Chapter 1
Introduction

Every opaque object has specific features which make it visually distinguishable from the rest. On the other hand, an ideal transparent substance would have no such features of its own, therefore making it visually impossible to recognize. Several objects like glass do give us a perception of transparency by passing most of the light through it and posing very few deterministic cues to the observer. Roughly speaking in the context of object recognition, transparency can be defined as an inverse-measure of the number of deterministic features specific to an object. So, a semi-transparent object would have fewer distinguishable features compared to an opaque object (Figure 1.1). In this work we discuss a collective-reward based approach for detecting such semi-transparent objects from a single image by a computer-vision system consisting of a camera. We also present a robotic-vision version of the algorithm to solve several robotic applications like obstacle avoidance, etc.

In the following, we review some of the related research work that was carried out in the area of transparency and its detection. Transparency has been a subject of research in the fields of psychology, vision and graphics. Among the earlier

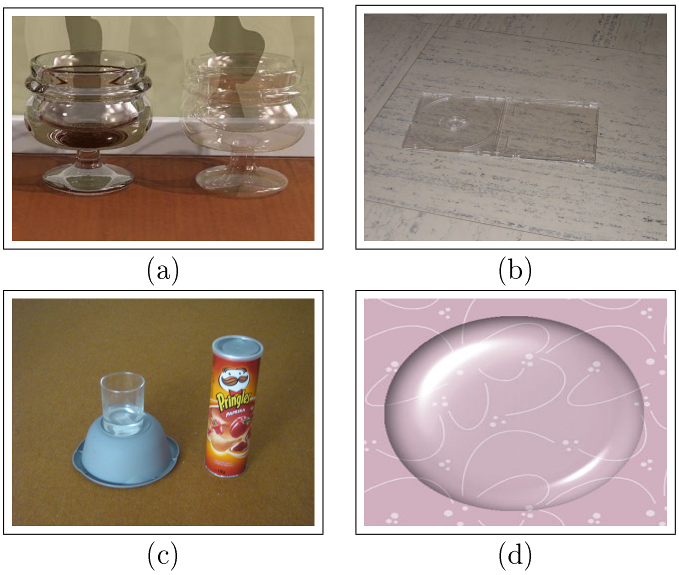


Figure 1.1: Figures show images containing semi-transparent objects. (a) Pair of glasses. (b) A plastic-case (c) Glass with opaque objects (d) Water drop.

researchers studying the phenomenon of transparency, gestalt psychologist Metelli is credited for making important and influential contributions to the theory of perceptual transparency [Singh 2002]. Perceptual transparency is the phenomenon of seeing one surface behind another. Metelli's model of transparency was based on a rotating episcotister, i.e. a rotating disk with reflectance t and an open sector of relative area α (Figure 1.2). When rotated in front of a bi-partite background whose two halves have different reflectance-values a and b, it would lead to a percept of transparent layer with a reflectance p and q overlying the opaque background.

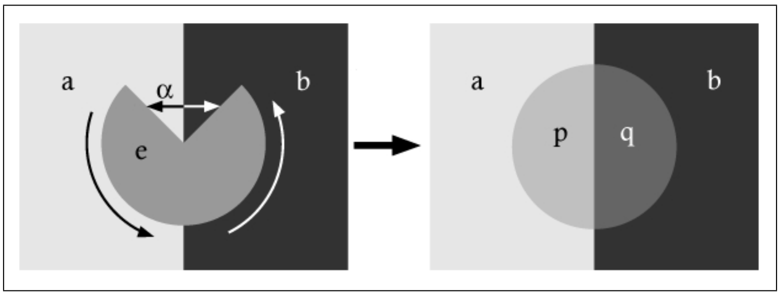


Figure 1.2: Figures illustrate Metelli's model of transparency using a rotating episcotister.

The color mixing in the region where the episcotister rotates over the background is given by Talbot's law:

$$p = \alpha a + (1 - \alpha)t \quad (1.1)$$
$$q = \alpha b + (1 - \alpha)t \quad (1.2)$$

Metelli derived two "qualitative constraints" for predicting the percept of transparency and they are:

- Polarity constraint: $sign(p - q) = sign(a - b)$.
- Magnitude constraint: $|p - q| \leq |a - b|$.

Metelli's magnitude constraint was later found out by Singh and Anderson [Singh 2006] to be inadequate in predicting the percept of transparency. The locus of transition between transparency and non-transparency was approximated instead by a constraint based on Michelson contrast (i.e. $\frac{p-q}{p+q} \leq \frac{a-b}{a+b}$).

Adelson and Anandan [Adelson 1990] used a linear model for the intensity of a transparent surface to achieve relationships between the X junction at the boundary of transparent objects. These relationships categorize the X junctions leading to interpretations that support or oppose transparency. Figure 1.3 illustrates several types of X junctions. Let p,q,r,s be the luminance values in the four regions surrounding the X junction, as indicated in Figure 1.3(d). The vertical edge retains the same sign in both halves of the X junction if p<q and r<s. Similarly, if p<r and q<s, the horizontal edge retains the same sign in both halves of the X junction.

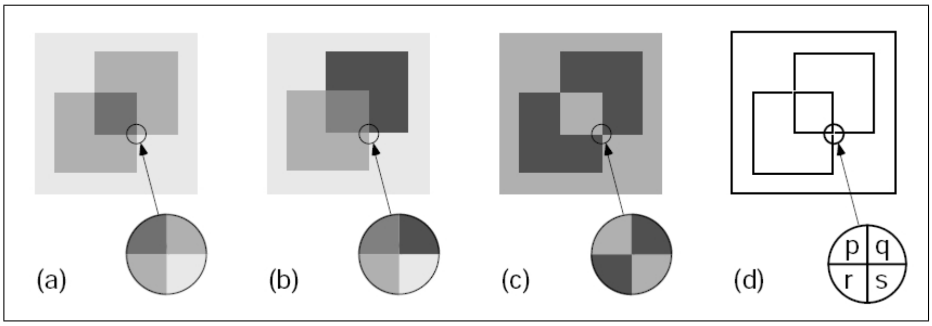


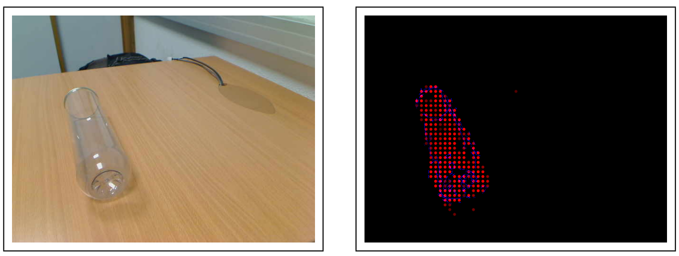
Figure 1.3: Figures(a)-(c) illustrate different types of X-junction. (a) Non-reversing X-junction (b) Single-reversing X-junction (c) Double-reversing X-junction. The conditions on luminance values shown in (d) are used to categorize the X-junction as one among the three types.

This is called a "non-reversing" junction because both edges retain their sign (Figure 1.3(a)). Figure 1.3(b) shows another X junction where the vertical edge changes sign and the horizontal edge retains its sign. This is called a "single-reversing" junction. And when both the edges change their sign then it is called a "double-reversing" junction. Non-reversing and single-reversing junctions support transparency and double-reversing junction does not support transparency.

The perception of transparency has received relatively lesser attention in the computer vision research community. Detection of transparent objects is a fairly difficult task as they do not have any distinguishable features of their own. Singh and Huang [Singh 2003] discuss about separation of transparent overlays from the background surfaces by making use of polarities of X junctions along the boundaries of objects. Transparent overlays are generally formed because of the presence of a transparent surface in front of an opaque object. Schechner *et al.* [Schechner 1998] have used the concept of depth from focus along with reconstruction to separate such overlays. Wexler *et al.* [Wexler 2002] have developed an approach for modeling transparent objects without the need of any specialized calibration by making use of multiple images of the same object over the same background with a relative motion between them. Ben-Ezra and Nayar [Ben-Ezra 2003] developed a model-based algorithm to recover the shape and pose of a transparent object in the scene from motion. They made use of the fact that changing the viewpoint changes the apparent background visible within the confines of a transparent object. Although, this requires for the background to be far away behind the transparent object. Hata *et al.* [Hata 1996] have proposed another approach in extracting the shape of transparent objects using a genetic algorithm. A comparison is performed over a real and a simulated image by tracing a slit light-line along an ordinary board on which the transparent object is placed. The error evaluation is then used to modify the model by making use of a genetic algorithm until the error falls below a threshold.

The main focus of the computer vision community remained on the problems concerning the detection of overlays and the reconstruction of the 3D shape structure of transparent surfaces. Relatively little work was carried out in the actual automatic detection of transparent objects in a scene. McHenry and Forsyth [McHenry 2005] used the edge information determined by a Canny edge detector to capture cues relating to transparent objects across their boundaries. These edges were combined using an active contour method to identify a single glass region. This method was later extended by McHenry and Ponce [McHenry 2006] with a region-based approach along with the edge information to classify regions to be transparent or not. They proposed two measures called the discrepancy measure and the affinity measure. The affinity measure provides an indication whether the regions belong to the same material and the discrepancy measure is used to indicate how close a region looks like a glass-covered region of the other. A region-based segmented image is used as an input to the algorithm. One of the issues reported by the authors is that initial segmentation may merge some parts of the transparent object into background and this cannot be recovered later in the process. Transparent objects with low refractive indices may face this issue. As the algorithm is dependent on the edge cues for connecting regions, it might lead to problems if the object has weak edges or if the background edges intersect the glass object. This is possible with lower refractive transparent objects like a plastic sheet. It has been suggested that an over segmented image would be preferred as an input to the system. An over-segmented image with a noisy background containing lots of edges might make it computationally intensive because the discrepancy measure is calculated for region samples about edge snippets. But on the other hand considering fewer snippets could be erroneous in a low resolution image.

In this thesis, we present an algorithm for the automatic detection of semi-transparent objects using the information available from a single image. In this regard we propose a method called the collective-reward based approach to achieve the detection and localization of the object's position in an image. The underlying principle behind this method is the fact that the pixels corresponding to a semi-transparent object have features which are similar to the surrounding pixels due



(a) (b)

Figure 1.4: (a) Figure shows a sample image with a semi-transparent object. (b) The final result of the algorithm.

to refraction and reflection of light on the object's surface. Figure 1.4(a) shows an example image which when fed to the algorithm has the result as shown in the Figure 1.4(b). The collective-reward based approach is then extended to fit onto a robotic vision system to detect glass doors and transparent obstacles present in the scene for a better improved mapping and navigation. It can also be used to detect and avoid water or oil spills on the floor. Several experimental results relating to the approach and its robotic application were carried out in order to prove the efficacy of the algorithm.

The report is organized as follows. In Chapter 2, we discuss the feature-cues used in the algorithm that are related to the transparent objects. The collective-reward based approach is presented in-detail in the Chapter 3. It also covers a description about how the feature-cues are used to generate rewards for classifying the pixels. The algorithm is trained offline to set the algorithmic parameters which are then fixed for generating the results. Chapter 4 discusses about the results of the experiments conducted over several sets of images captured from a web cam and also from the Internet. The collective-reward based approach is modified to suit to robotic applications of which obstacle-avoidance of transparent objects is discussed in Chapter 5. Snapshots of videos taken from several experiments conducted on a mobile robot are also presented. Finally, we conclude the report by providing some insights on the various other applications of the algorithm and also discuss some of our future work in Chapter 6.

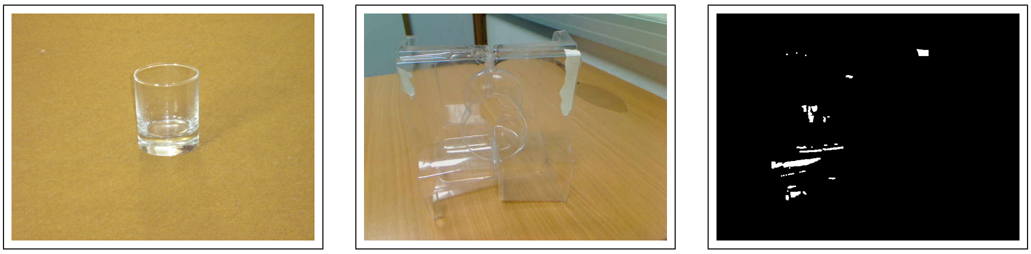
Chapter 2

Feature-Cues

Contents

This chapter presents a description of the features-cues used in our algorithm that are usually present with semi-transparent objects. The following cues are quantified via feature-reward functions, details of which are later discussed in Chapter 3.

2.1 Highlights and Caustics

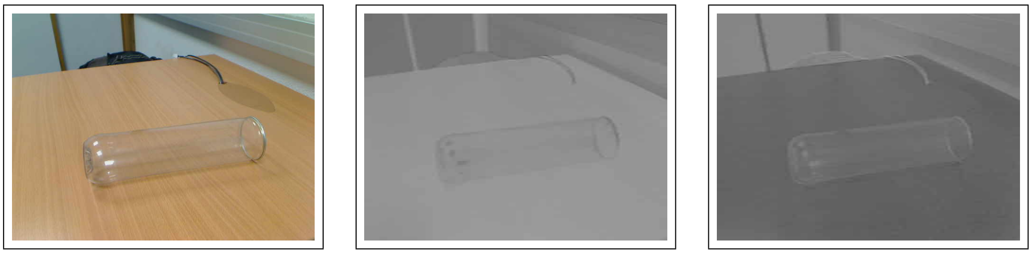


(a) (b) (c)

Figure 2.1: (a) Figure shows the presence of caustics on a semi-transparent object (glass). (b) Figure shows the presence of highlights on a semi-transparent object (plastic case), and Figure (c) shows the segmented highlight pixels.

Highlights are strongly illuminated regions in the image formed due to the specular nature of a surface. Transparent objects are usually highly specular, therefore the presence of highlights increases the probability of a possible transparent material around. Transparent objects like glass are also known to be refractive resulting in caustics which also serve as a cause for the highlights in the image (Figure 2.1(a)). Several methods exist in the literature discussing the detection of highlights [Klinker 1990]. It has been known from the literature that *value* and *saturation* quantities of the HSV color space [Ford 1998] could be used to determine the pixels belonging to achromatic regions in the image. Highlights are bright white pixels that can be found in an image using *value* $> 75\%$ and *saturation* $< 20\%$ [Androutsos 1999]. Figure 2.1(b) shows a sample image of a transparent object with highlights. The pixels corresponding to the highlights are segmented out as shown in the Figure 2.1(b). Although highlights are a valuable cue in detecting transparent objects, there is always a possibility that they could belong to an opaque object or a white paper. In addition to that, they do not carry much other information thereby making them points of high uncertainty. But the presence of these points definitely increases the probability of the surrounding non-highlight points to belong to a transparent object as they can be further tested for other reward functions. After the detection of transparent points, the highlights are then classified as points belonging to transparent or opaque objects.

2.2 Color



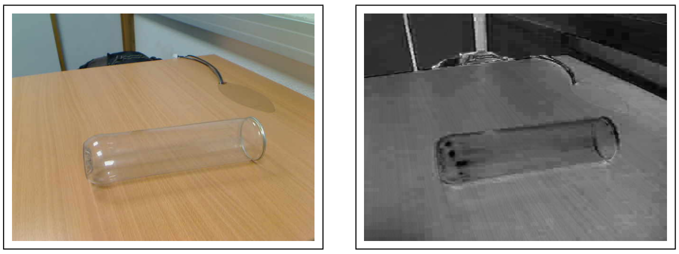
(a) (b) (c)

Figure 2.2: (a) Figure shows an image containing a semi-transparent object. (b) Its corresponding Cr color channel and (c) Cb color channel image.

A highly transparent object would produce almost negligible distortion in the color of the background. On the other hand, semi-transparent objects like glass, plastic, etc. generally have impurities and also due to the presence of specular reflections, the background color is slightly distorted. We made use of the $YCrCb$ color model [Ford 1998] to encode the color information as the 'luma' (Y) component can be separated out making Cr and Cb components to robustly indicate the color attribute invariant to intensity (Figure 2.2).

2.3 Saturation

Saturation serves as another valuable cue in detecting transparent objects. Transparent objects have a slight blurring effect on the background. The pixels belonging to these blurred regions tend to have less vivid colors than pixels corresponding to the unblurred region [Liu 2008]. Therefore these pixels have relatively lower *saturation* values (Figure 2.3).



(a) (b)

Figure 2.3: (a) Figure shows an image containing a semi-transparent object and (b) its corresponding *saturation* channel image. Semi-transparent objects tend to lower the *saturation* values.

2.4 Reflections and Intensity

(a) (b)

Figure 2.4: (a) Figure shows the specular reflection of the side wall about the surface of the semi-transparent object. (b) Figure shows a gray scale image of a semi-transparent object present on a textured floor. There is a slight reduction in the contrast of the texture appearing on the object.

As previously discussed transparent objects are usually highly specular, the light rays coming from the objects around could bounce off the surface of the semi-transparent object and reach the camera. Therefore, the apparent texture present on the transparent objects may not necessarily be the same as that of the background. Figure 2.4(a) shows an image containing a semi-transparent object. We can see the reflection of the white wall on the semi-transparent object.

Intensity plays a major role for backgrounds with texture. We made use of Michelson's contrast constraint as it has been shown by Singh and Anderson [Singh 2002] that transparency lowers its value. Figure 2.4(b) shows a gray scale image of a semi-transparent object. We can observe the slight reduction in the contrast of the texture appearing on the semi-transparent object.

2.5 Cross-Correlation Measure

Cross-correlation is a measure of how well two signals match with each other. A small window is used as a template to be slided over a small rectangular region as shown in the Figure 2.5. The window slides across the region and the normalized cross-correlation is calculated at each point. The maximum and minimum of the result is found and reported. Normalized cross-correlation values will be higher for points that belong to similar regions. As glass produces a slight distortion effect on the background, the measure will be relatively lower. On the other hand two non-similar patches would report for an even lower measure. In order to reduce the effect of noise and improve the result, $YCrCb$ color space channels of the image are fed as an input.

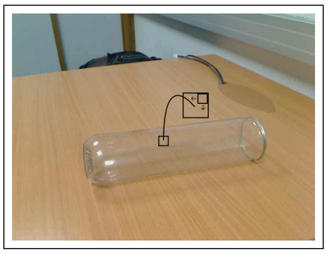


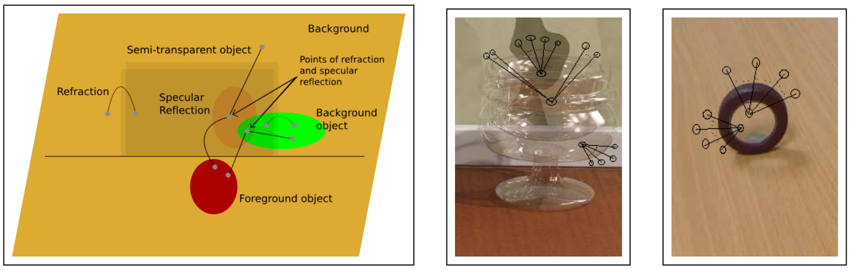
Figure 2.5: Figure shows an illustration of how the cross-correlation measure is determined in an image containing a transparent object.

CHAPTER 3

Collective-Reward Based Approach

Contents

Semi-transparent objects like glass, plastic, etc. not only transmit light but also reflect the light coming from the surrounding objects, typically from the foreground. Therefore, the pixels corresponding to the semi-transparent objects have features similar to that of the surrounding pixels. This is because these pixels actually contain the distorted features of what lies behind the semi-transparent object [Murase 1992]. Figure 3.1(a) illustrates the transmitted and reflected light coming from an object in the background and the foreground respectively. Since from a single image we do not have access to the actual features of the pixels behind the semi-transparent object, we therefore use the surrounding pixels to judge whether a pixel belongs to a semi-transparent object, an opaque object or a point of the background. Figures 3.1(b) and 3.1(c) show two sample images with correspondences drawn between a point inside and outside the semi-transparent object. As the boundary corresponding to the semi-transparent object is not known, a random hypothetical region R is selected in an image. Several connections are established between each point (p_i) inside the region and points (p_e) outside the region. Each of these connections are tested with the feature functions to generate a reward which is received at p_i. A reward is a decimal-point value ranging from 0 to 1. The reward will be high if p_i belongs to a semi-transparent object having features similar to p_e and low if the point p_i belongs



(a) (b) (c)

Figure 3.1: (a) The figure illustrates the contribution of features via refraction and specular reflection. Figures (b) and (c) illustrate the collective reward received via several connections established (with good fitness values) between the points of a semi-transparent object and the surrounding. Figures should be viewed in color.

to an opaque object or the background itself. In order to avoid rewards due to noisy points in the image and to provide more emphasis on the strongly co-related points, each connection is given an appropriate weight to indicate its fitness value. This also accounts for the specularity feature of the glass. These weights are calculated using support fitness functions, discussed in-detail in section 3.1.

Let R_I and R_E denote the region interior and exterior to R respectively. Let (I^i) denote the reward (refer to section 3.3.2 for a detailed description about how the rewards are calculated) received at each pixel belonging to R_I. Let (I_e^j) denote the reward given by each of the pixels that belong to R_E. A one-to-many relationship is established between each point of R_I to every point in R_E. Let there be k such connections, the reward at each point p_i is given by the equation (3.1).

$$W_1^{'} I^i = W_1 I_e^1 + W_2 I_e^2 + ... + W_k I_e^k \quad \forall p_i \in R_I \tag{3.1}$$

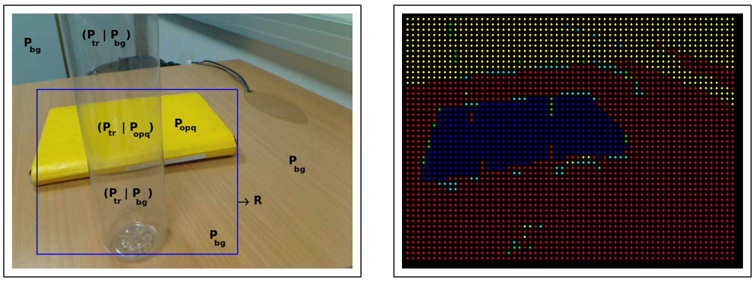
Where $\{W_1, W_2, ..., W_k\}$ are the weights denoting the strength of each connection $(p_i \mapsto p_e^1, p_i \mapsto p_e^2, ..., p_i \mapsto p_e^k)$, $\forall p_i \in R_I$. $W_1^{'}$ is a normalization factor equal to $(W_1 + W_2 + ... + W_k)$. The weight $W_j, \forall j \in \{1, ..., k\}$ denotes the fitness value of the j^{th} connection. Let P_{tr} denote points corresponding to semi-transparent objects in R_I (Equation (3.2)). Let P_{opq} denote all the points in R_I that have features different from the features of points in R_E (Equation (3.3)). These points belong to the opaque objects and background textures that lie only in R_I. We will henceforth refer to these points as opaque points. Let P_{bg} denote points corresponding to the background that lie in $\{R_I \cup R_E\}$ (Equation (3.4)).

$$P_{tr} = \{p \mid p \in \{\text{semi-transparent objects}\} \cap R_I\} \tag{3.2}$$

$$P_{opq} = \{p \mid p \in \{\text{opaque objects or patches}\},\ p \cap R_E = \phi\} \tag{3.3}$$

$$P_{bg} = \{p \mid p \in \{P_{total} \setminus (P_{tr} \cup P_{opq})\}\} \tag{3.4}$$

Let $(P_{tr}|P_{bg})$ denote all those points of semi-transparent objects with features similar to the points in P_{bg}. Without any loss of generality all the points in R_E are



(a) (b)

Figure 3.2: (a) The figure shows the regions corresponding to semi-transparent, opaque and background points in the image. A hypothetical region R indicated by the blue rectangle is used for the illustration. (b) The figure shows a coarse clustering output over the entire image based on color and intensity gradients.

considered as background. Let $(P_{tr}|P_{opq})$ denote all those points of semi-transparent objects with features similar to the points in P_{opq}. As the opaque points are not similar to the background, we have

$$(P_{opq}|P_{bg}) = \emptyset \tag{3.5}$$

Therefore, the point-set corresponding to the semi-transparent objects in R_I is given by the Equation (3.6).

$$P_{tr} = (P_{tr}|P_{opq}) \cup (P_{tr}|P_{bg}) \tag{3.6}$$

Figure 3.2(a) shows a sample image illustrating different point sets (opaque points, semi-transparent points and background points). Let P_{total} denote the set of all points in the image. We have,

$$P_{total} = P_{opq} \cup (P_{tr}|P_{opq}) \cup (P_{tr}|P_{bg}) \cup P_{bg} \tag{3.7}$$

The problem now ramifies down to finding sets $(P_{tr}|P_{opq})$ and $(P_{tr}|P_{bg})$. Our approach is to first segment the set P_{total} into three clusters (P_{bg}), $(P_{tr}|P_{bg})$, $(P_{opq} \cup (P_{tr}|P_{opq}))$ out of which the cluster $T_1 = (P_{tr}|P_{bg})$ is extracted and the cluster $(P_{opq} \cup (P_{tr}|P_{opq}))$ is further processed to extract the cluster $T_2 = (P_{tr}|P_{opq})$. The clusters T_1 and T_2 are reported as a final result.

To begin with, a coarse spatial clustering of the point set P_{total} is carried out to separate similar regions based on color (Cr and Cb) and intensity gradients (I_x and I_y) (Figure 3.2(b)). The clustering is done using the K-means algorithm [MacKay 2003]. The number of clusters is computed automatically based on inter-cluster mean distances and other thresholds. Let C denote the index-set of all the clusters. Let the clusters present in R_I and R_E be equal to C_I and C_E respectively. These clusters are used to calculate the fitness values of the connections made between points in R_I and R_E.

3.1 Support Fitness Functions

3.1.1 Clusters Fitness Function

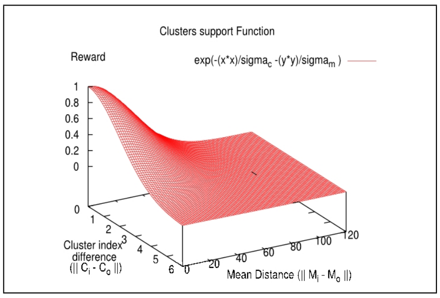


Figure 3.3: Clusters fitness function

Since a coarse clustering is carried out, the points on the semi-transparent object would generally either belong to a cluster of the background, or a cluster that belongs to opaque points, or a cluster of points corresponding to highlights (Figure 3.2(b)). The reward received through a connection would be higher if points of the same cluster are compared. On the contrary, since clustering is performed coarsely, there are chances for the following possibilities:

- $C_I(p_i) \neq C_E(p_e)$, $\mu(C_I(p_i)) \approx \mu(C_E(p_e))$
- $C_I(p_i) = C_E(p_e)$, $\mu(C_I(p_i)) \gg (or \ll)\ \mu(C_E(p_e))$

where, $\mu(C)$ denotes the distance between the cluster gravity centers. The first listed possibility occurs when the points of the semi-transparent objects that are quite similar to the background fall into different clusters. These connections with non-equal cluster index should not be neglected. The second possibility occurs when cluster indices are identical but the distance between the clusters is large. The fitness value of this connection should be relatively lower. In order to account for these possibilities a 2D Gaussian fitness function is used with cluster-index distance (cid) on one axis and the mean-distance (cd) on the other axis. The cluster-index distance (cid) is defined as the difference in the positions of clusters-indices in a sorted cluster-index set. Sorting was done based on mean-distances between the cluster centers. The combination of index and μ provides separability among closely similar clusters but also provides connectivity among clusters with wide mean separation. The Clusters fitness function as shown in the Figure 3.3 is given by:

$$W_{Cj} = e^{-\left(\frac{cid^2}{2\sigma_c^2} + \frac{cd^2}{2\sigma_d^2}\right)} \tag{3.8}$$

Where, σ_c is the standard deviation with respect to cluster-index distance and σ_d is the standard deviation with respect to absolute mean-difference.

3.1.2 Distance Fitness Function

Euclidean distance forms an important fitness function determining the strength of a connection. The chance of finding a similarity in features between a semi-transparent object and the background is higher if they are closely situated. On the contrary, in situations where a semi-transparent object is placed to the side of an opaque object, there are chances that the distance function might localize on the connections between the points of semi-transparent object and the opaque object producing a poor result. Introspecting at another possibility, we have, if the region R is large, a point selected at its bottom-left part will generally have low correlation with the exterior points closer to the top-right corner of R. On the other hand, if the region R is small then the point will have correlations with all the surrounding exterior points. In addition to the above possibilities we also have the effects of perspective, distance blur, focus and radial distortion in the camera. In order to account for these possibilities we made use of the fitness function (3.9). Where, D is equal to the euclidean distance $||p_i - p_e||$, $\sigma_1 = \frac{2}{3}min(Rwidth, Rheight)$ and $\sigma_2 = \frac{2}{3}max(Rwidth, Rheight)$, where $Rwidth$ and $Rheight$ are the dimensions of the region R.

$$W_{Dj} = 0.7e^{-\frac{D^2}{2\sigma_1^2}} + 0.3e^{-\frac{D^2}{2\sigma_2^2}} \tag{3.9}$$

The resultant fitness function for each link is given by equation (3.10)

$$W_j = W_{Cj} * W_{Dj} \tag{3.10}$$

3.2 Post Processing Functions

3.2.1 Nearest Neighbor Transparency (NNT)

The detection process may leave some gaps in the semi-transparent object. Some of these gaps can be filled with the help of the NNT function. There is a high probability for a point to be semi-transparent if the points surrounding it are semi-transparent. So at each point, the weighted average of reward values in the neighborhood is found and added to the existing value. NNT can also help in removing noise points. By making the reward values negative that fall below a threshold, the average could turn out negative if there are more neighboring non-transparent points, thereby reducing the reward value.

3.2.2 Edge Cues

Semi-transparent objects do show up edge cues because of higher refractive indices, grounded edges, caustics or opaque boundaries. A logical conjunction (And) of the total reward function is performed with an edge map, thereby retaining only those edges that belong to transparencies.

3.3 Total Reward and Classification

3.3.1 Offline Training

An offline training was carried out over a set of images to construct the reward functions. A rectangular region of a semi-transparent object is extracted manually from the image. Let T_I represent the set of all points that belong to the region. Two more different rectangular regions of the similar background are extracted manually from the image. Let T_{E1} and T_{E2} denote the sets of all points that belong to the two regions. For each point p_b taken from the set T_I a difference set $T_D^t(p_b)$ is generated as shown in equation (3.11).

$$T_D^t(p_b) = \{|f(p_a) - f(p_b)|, \forall p_a \in T_{E1}\} \tag{3.11}$$

Where $f(x)$ denotes the corresponding feature-value at x. A histogram H^t of the difference set is generated and averaged over each point p_b in T_I.

$$H^t(p_b) = Hist\{T_D^t(p_b)\} \tag{3.12}$$

$$H_{Avg}^t = \frac{1}{\aleph(T_I)} \sum_{\forall p_b \in T_I} H^t(p_b) \tag{3.13}$$

where,$\aleph(T_I)$ represents cardinality of the set T_I. Similarly, a histogram H^o is generated and averaged over each point p_b in T_{E2}.

$$T_D^o(p_b) = \{|f(p_a) - f(p_b)|, \forall p_a \in T_{E1}\} \tag{3.14}$$

$$H^o(p_b) = Hist\{T_D^o(p_b)\} \tag{3.15}$$

$$H_{Avg}^o = \frac{1}{\aleph(T_{E2})} \sum_{\forall p_b \in T_{E2}} H^o(p_b) \tag{3.16}$$

Let $(0, G)$ denote the domain of the histograms generated. The reward function Rw is found using equation (3.19).

$$N_t(l) = H_{Avg}^t(l) \tag{3.17}$$

$$N_o(l) = H_{Avg}^o(l) \tag{3.18}$$

$$Rw(l) = \frac{N_t(l)}{N_t(l) + N_o(l)}, \forall l \in (0, G) \tag{3.19}$$

The reward function Rw is found for Cr, Cb and saturation channels using the above approach with a slight modification for saturation, where a positive difference-set is generated using the equation (3.20). This is done in order to account for the fact that saturation values of semi-transparent objects are lower compared to the background.

$$T_D(p_b) = \{|f(p_a) - f(p_b)|\ H[p_a - p_b], \forall p_a \in T_{E1}\} \tag{3.20}$$

Where, $H[n]$ denotes a Heaviside step function.

The training is performed using several such rectangular regions of different semi-transparent objects and different backgrounds and a function (Rw_i^+) is generated.

The training is also done with negative samples by considering regions taken from distinguishable opaque objects giving a reward function (Rw_i^-). The average reward function generated is given by (3.21).

$$Rw_{avg} = \frac{Rw_1^+ + Rw_2^+ + ... - Rw_k^- - Rw_{k+1}^- - ... - Rw_N^-}{N} \tag{3.21}$$

3.3.2 Reward Functions

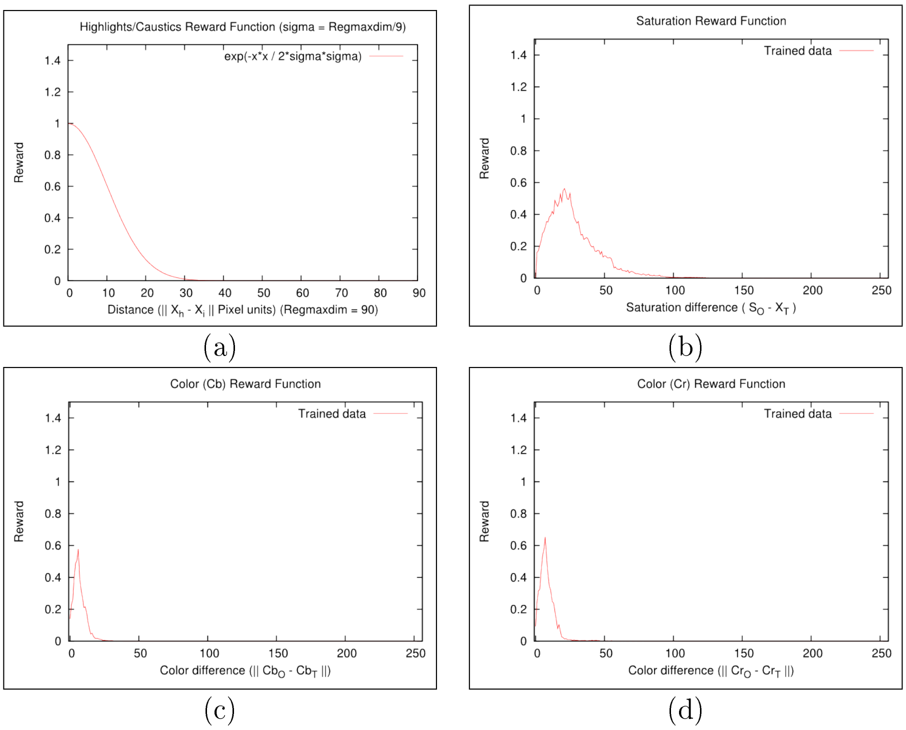


Figure 3.4: The figures show graphs of (a) Highlights reward function (b) Saturation-reward function (c) Cb-reward function and (d) Cr-reward function.

For each point inside the region R, k best points outside the region are found using the support fitness functions as discussed in the previous sections. For each of these connections let p_i denote the point inside and let p_e denote the point outside the region R. The following are the feature-reward functions used to generate a reward for each of these connections based on the cues discussed in the Chapter 2.

3.3.2.1 Highlights and Caustics

The reward function for these points is given by the Gaussian function as shown in the Figure 3.4(a), with euclidean distance between the point and the closest highlight-point as an argument. After the detection of semi-transparent points, the highlights which were rewarded low are now classified as points belonging to semi-transparent or opaque objects based on the euclidean distance from a closest semi-transparent point using the same reward function.

3.3.2.2 Saturation

The pixels corresponding to the semi-transparent objects have relatively lower *saturation* values compared to the pixels corresponding to the background. The difference $(Sat(p_e) - Sat(p_i))$ is used as an argument that is passed to the reward function. We had carried out offline training to determine the reward function as shown in the Figure 3.4(b).

3.3.2.3 Color

We made use of the $YCrCb$ color model to encode the color information. This is because the 'luma' (Y) component can be separated out making Cr and Cb components to robustly indicate the color attribute invariant to intensity. The absolute difference between the median of a 3x3 neighborhood centered at p_i and p_e is used as an argument that is passed to the reward function. Rewards for each color component Cr and Cb are found from the individual reward functions and the product of both the rewards is reported. We had carried out offline training to determine the Cr and Cb reward functions as shown in the Figures 3.4(c) and 3.4(d) respectively.

3.3.2.4 Intensity

The maximum and minimum intensities are calculated in a KxK neighborhood centered at p_i and p_e. These values are used to calculate Michelson's contrast. Michelson's contrast is defined as

$$C = \frac{I_{max} - I_{min}}{I_{max} + I_{min}} \tag{3.22}$$

The difference in the average intensities calculated in the neighborhood centered at p_i and p_e, is checked if it falls below a threshold T and a reward is generated using the reward-function given by

$$\text{Reward} = \begin{cases} \frac{C_{p_e} - C_{p_i}}{C_{p_e} + C_{p_i}} : & C_{p_e} \geq C_{p_i} > 0,\ |(I_{p_e}^{avg} - I_{p_i}^{avg})| < T \\ 0 \quad : & otherwise \end{cases}$$

3.3.2.5 Cross-Correlation Measure

A small window of size KxK centered at p_i is used as a template to be slided over a rectangular region MxM centered at p_o with $(M > K)$. The window slides across the region and the normalized cross-correlation is calculated at each point. The maximum and minimum of the result are found and reported. Normalized cross-correlation values will be higher for points that belong to similar regions. Thresholds used were found based on an offline training over several images. In order to reduce the effect of noise and improve the result, $YCrCb$ color space channels of the image are fed as an input.

3.3.3 Classification

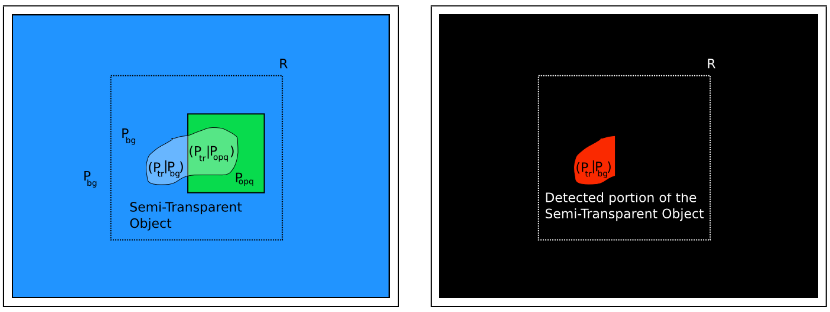
The reward value corresponding to each connection $I_e^j, j \in (1, ..., k)$ (Refer to the Equation (3.1)) connecting a point $p_i \in R_i$ to a point $p_e \in R_E$ is found from the feature functions:

$$I_e^j = Rw(g(p_i, p_e)) \tag{3.23}$$

$$I^i = \frac{1}{W_1'} \left(W_1 I_e^1 + W_2 I_e^2 + ... + W_k I_e^k \right) \tag{3.24}$$

Where, $g(p_i, p_e)$ denotes the functional argument passed to each of the feature-reward functions (highlights, Cr, Cb, Saturation, Intensity and Cross-correlation). The reward value I^i received at the point $p_i \in R_I$ through all the k connections is given by equation (3.24) and the weights W_j are computed as shown in Section 3.1. The total reward $fm(p_i)$ is then calculated as a combination of individual feature reward values. As individual feature functions are weak classifiers, an ensemble of classifiers is formed to generate a strong classifier. The total reward is modified by the post-processing functions to get a normalized result.

3.4 Intra-Region Clustering

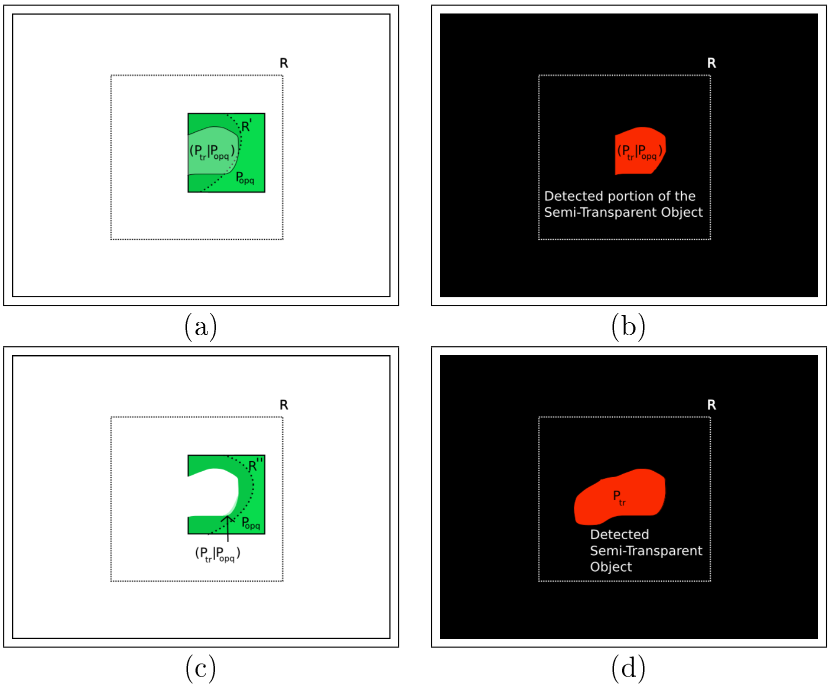


(a) (b)

Figure 3.5: Figure (a) illustrates the regions relating to the semi-transparent object that have features similar to the background and the opaque patch (object). (b) The final outcome on carrying out the collective-reward based approach using the region R.

Summarizing the process carried out till now, we perform a coarse clustering of all the pixels in the image into several clusters based on color and intensity gradients. These points are then divided into two regions based on whether they lie interior or exterior to the hypothetical region R. A point node taken from the inner-region is compared with points taken from the outer-region. Each of these connections are analyzed for their strengths through which rewards are collected by the inner points. Finally, the total reward is calculated after passing through the post-processing functions.

On performing a comparison between the points inside and outside the region R , the set $(P_{tr}|P_{bg})$, i.e. the points of semi-transparent objects that have features similar to the background receive high reward while the points P_{bg} (background points), P_{opq} (opaque points) and $(P_{tr}|P_{opq})$ (semi-transparent points that have features similar to the opaque points) receive a low or zero reward. The result is thresholded to extract $(P_{tr}|P_{bg})$. But in the process the points $(P_{tr}|P_{opq})$ that also belong to semi-transparent objects are lost because they do not have features similar to P_{bg}. Figure 3.5(a) shows an illustration with a region R encompassing a semi-transparent object and an opaque object. The regions of the semi-transparent object are labeled accordingly. Figure 3.5(b) shows the result of the collective-reward based approach with the region R. Only the portion $(P_{tr}|P_{bg})$ is segmented and the rest of the semi-transparent object is filtered out. In order to recover this portion we need to perform an additional operation called the intra-region clustering.



(a) (b) (c) (d)

Figure 3.6: (a) Figure shows an opaque patch and a semi-transparent object in-front of it. It also shows the new hypothetical region R'. (b) The final outcome on carrying out the collective-reward based approach using the region R'. (c) Intra-region clustering is carried out once again to find a new region R'' to improve the final outcome. (d) The normalized appended result found after carrying out the collective-reward based approach using the regions R, R' and R''.

We first find out all those color clusters that belong only to R_I.

$$P_{opq} \cup (P_{tr}|P_{opq}) = \{p_i | P(C_I(p_i)) \cap P(C_E(p_e)) = D, \aleph(D) < \delta, \ \forall p_i \in R_I, \ \forall p_e \in R_E\} \tag{3.25}$$

Where $P(C_I(p_i))$ and $P(C_E(p_e))$ denote the set of all points that belong to the cluster-index of the point p_i and p_e respectively. δ is a small number to account for those objects whose majority of the portion lies inside the region but may have some points in the exterior. Figure 3.6(a) shows the extracted region $P_{opq} \cup (P_{tr}|P_{opq})$. A Mahalanobis distance set for each of the color clusters is generated between each point that is extracted using the equation (3.25) and the gravity center of the detected transparent points within a close range. The median of the distance set is used to partition the point-set $(P_{opq} \cup (P_{tr}|P_{opq}))$ into two clusters. Figure 3.6(a) shows the partition R' which separates the region into two clusters. The cluster that is closer to the detected portion of semi-transparent object is denoted as interior of the new region R' and the rest is exterior. The collective-reward based approach is carried out between these clusters to extract most of the points that belong to $(P_{tr}|P_{opq})$. Figure 3.6(b) shows the segmented portion $(P_{tr}|P_{opq})$. Although using the region R' we may not be able to extract all the points corresponding to the semi-transparent object. In the Figure 3.6(b) we can see that a small portion of the semi-transparent object on the bottom-right corner is lost. This is because the new region R' did not encompass the whole of the semi-transparent object. To get a better result, the intra-region clustering can be performed another time to get a new region R'' which will encompass the remaining portion of the semi-transparent object (Figure 3.6(c)). Finally, the result is normalized and combined to give a final result P_{tr} as shown in the Figure 3.6(d).

3.5 Automatic Region Selection

Region selection is an important factor which affects the outcome of the algorithm. As the comparisons are made between the points inside and outside the region, a single region may not be sufficient to locate the semi-transparent objects in the entire image. We could either have multiple regions scanned across the image or a hierarchical set of regions with varied dimensions. Several such methods exist which can be used depending on the requirements of the end user. One approach we followed to automate the result is by using a two step region selection. The algorithm is executed initially using a large region encompassing most of the image as shown in the Figures 3.7(a) and 3.7(b). The resulting reward for the points situated at the central part of the region may be erroneous as the distance between the point comparison is large (Figure 3.7(c)). In order to improve the result, we use the outcome of the first iteration as a region for the second iteration. This will re-check all the points that have been detected as semi-transparent points by executing the algorithm with the close-by points in the second iteration. Therefore, the erroneous points get eliminated as shown in the figure 3.7(d). If the computational time is not a constraint, the algorithm can be run via this two step method for several large

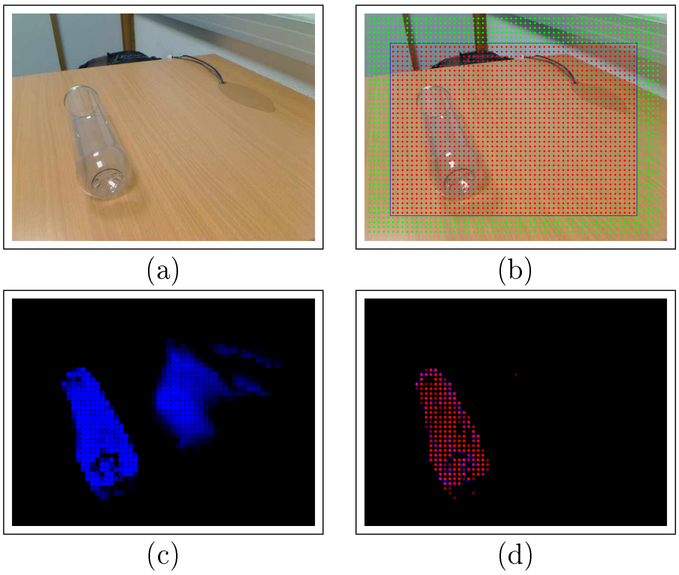
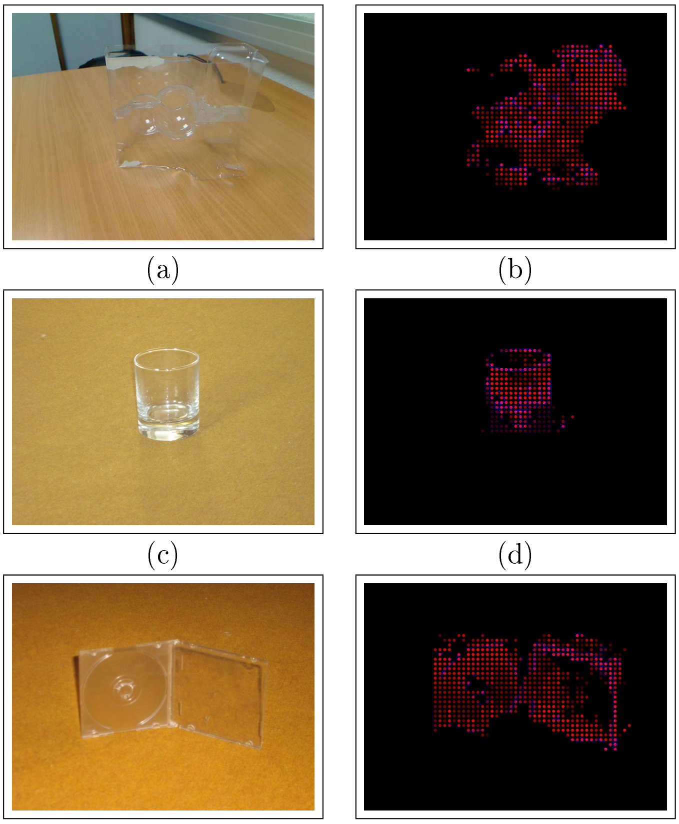


Figure 3.7: (a) Figure shows a sample image with a semi-transparent object. (b) A large region denoted by red-points is selected for the first iteration. (c) The outcome of the first iteration with erroneous result for the points at the central part of the region. (d) The outcome of the second iteration after using the output of the first iteration as the region R itself.

regions in the image to get a best possible result.

Chapter 4

Experimental Results



(a) (b)

(c) (d)

(e) (f)

Figure 4.1: Figures (a)-(f) show a few semi-transparent objects and the corresponding detection results. The images should be viewed in color. For all results we used the same parameters that were learnt via offline training and a two-step region selection method.

In this chapter we will present some of the results of several experiments conducted to test our algorithm over several images taken from a webcam and also from the Internet. We made use of a Logitech web camera to capture images of resolution equal to 640x480. One of the main reasons behind using the webcam is to extend the approach to a robotics platform accounting for the effects of perspective and noise due to distance, focus and radial blur.

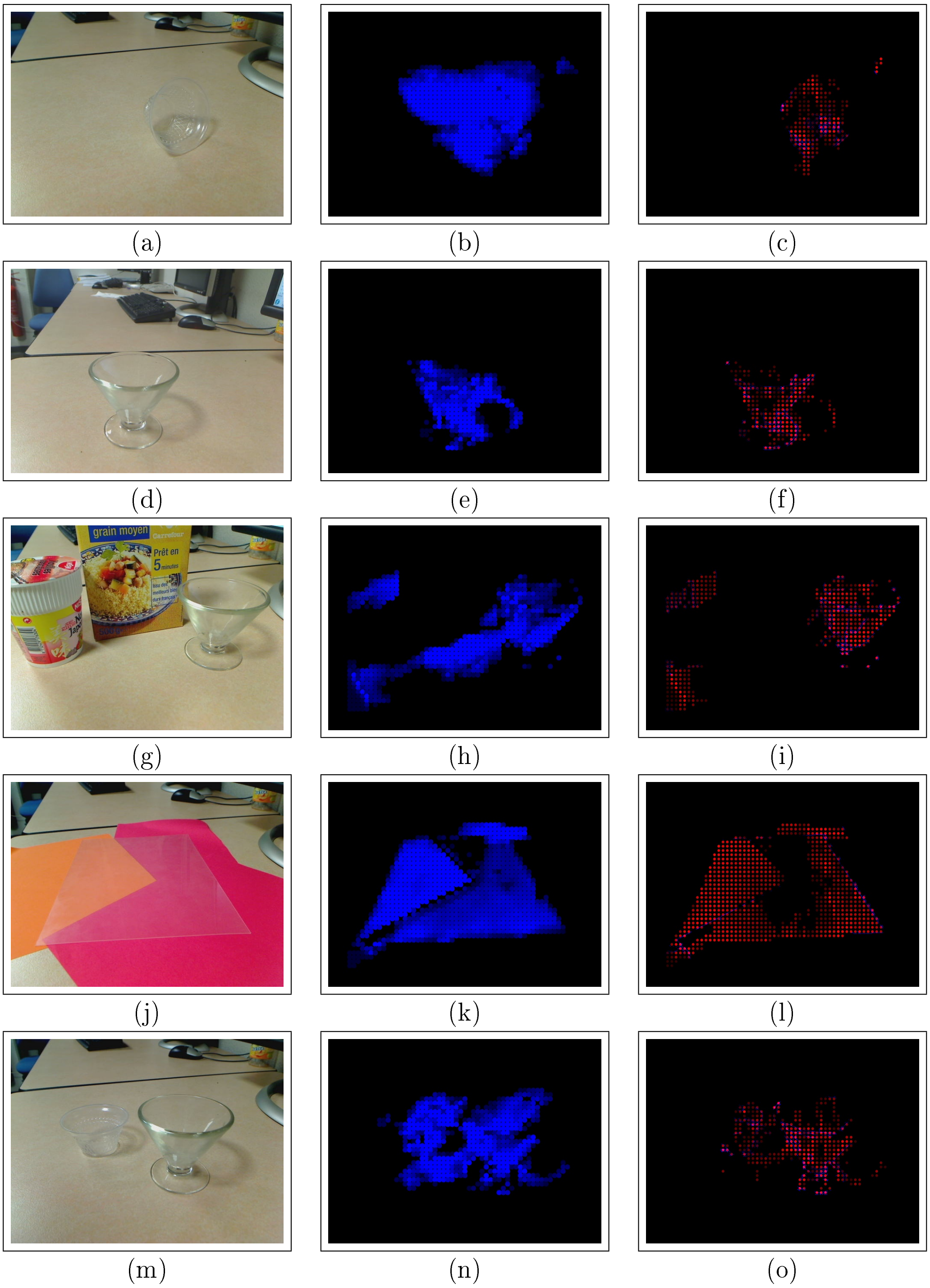
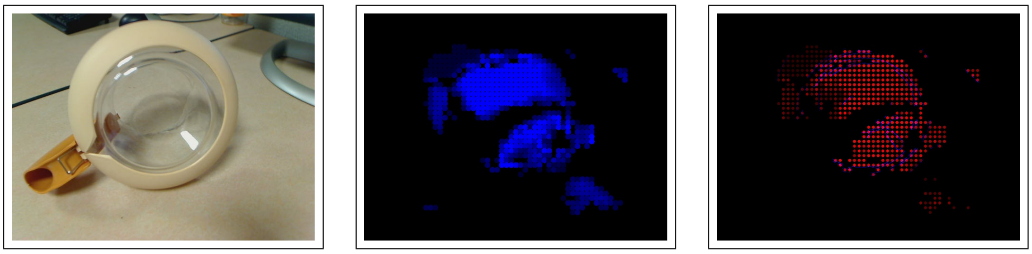


Figure 4.2: Figures (a)-(o) show a few semi-transparent objects and the corresponding detection results using the two-step region selection method. Second column shows the result of the first iteration and third column shows the result of the second iteration. The images should be viewed in color.

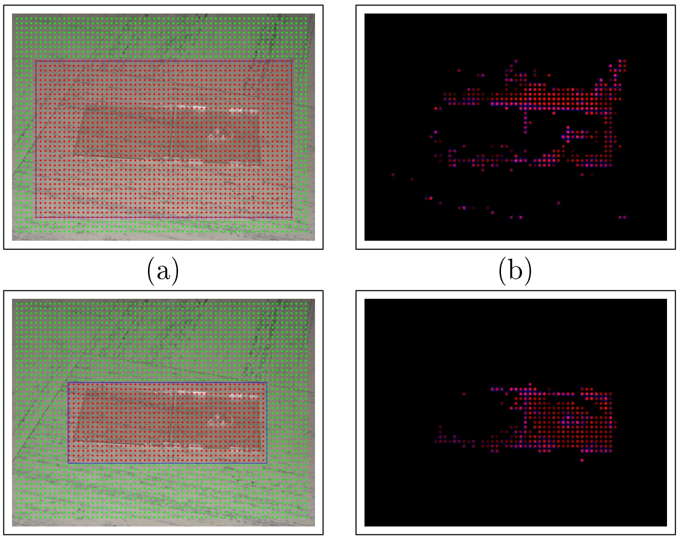
To make the algorithm computationally tractable, the point nodes considered are sampled as 1 node to 10 pixels in the image. The exterior points used for comparison are restricted to the 40 best connections found out by the fitness values. To train the reward functions we collected 35 sample-regions of transparent objects and the corresponding close-by regions of the background. Figures 4.1-4.2 show results of a few images containing semi-transparent objects over different backgrounds. As discussed in the section 3.5, we evaluated our approach using the two-step region selection. The algorithm is executed initially using a large region encompassing most of the image. We then use the outcome of the first iteration as a region R for the second iteration. Figures 4.2 show results of the two-step region selection method. The second column shows the result of the first iteration of the algorithm. We can see that the first-iteration results are noisy at the central part of the image. This is due to the fact that the comparisons for the central points are made with distant points outside the region. The third column of the Figure 4.2 shows the result of the second iteration. We can clearly see that the precision of the result has improved. Figures 4.2(a)-(c) and 4.2(d)-(f) shows experiments conducted using a thin plastic cup and a refractive glass respectively. Figure 4.2(g) contains a semi-transparent object made of glass along with two opaque objects. The algorithm detects the glass as shown in the Figure 4.2(i). We can notice that there are few false positives detected on the left. This is because the algorithm had picked up the points belonging to the dim-shadow and classified them as points belonging to semi-transparent regions. This could be explained by the fact that light shadows also appear as transparencies over the background as they are very similar and hold most features of a semi-transparent object. From a single image it remains a challenging problem to filter out the dim-shadows from the result. Although, as the variation in the shadow is gradual, a nearby-point comparisons will eliminate most part of it. Figures 4.2(j) shows another scenario with a transparent sheet placed on sheets of different color. The algorithm successfully detects the transparent sheet in the first iteration itself. We can see a small undetected patch on the top portion of the transparent sheet. On performing the second iteration, many exterior points are taken from this undetected patch. Therefore we see that the more points of the transparent sheet get filtered due to the point-point comparisons within the transparent sheet. Figures 4.2(m)-(o) show the result of another experiment with two transparent objects placed next to each other.

To quantify the results, we carefully marked the boundaries of the semi-transparent objects in an image and performed experiments by measuring the precision and the recall rate. We collected 50 images with different objects and background scenarios. As the final result of the algorithm is probabilistic, the precision rate is calculated taking this into account. True positives are measured as the number of detected points, with the corresponding probabilities, lying within the boundary. Similarly, false positives are measured as the number of detected points along with the probabilities lying outside the marked boundary. We achieved a precision rate of 77.19% and a recall rate of 65.84%. We consider the rates to be pretty good due to the fact that detection of the transparent objects is a fairly difficult prob-



(a) (b) (c)

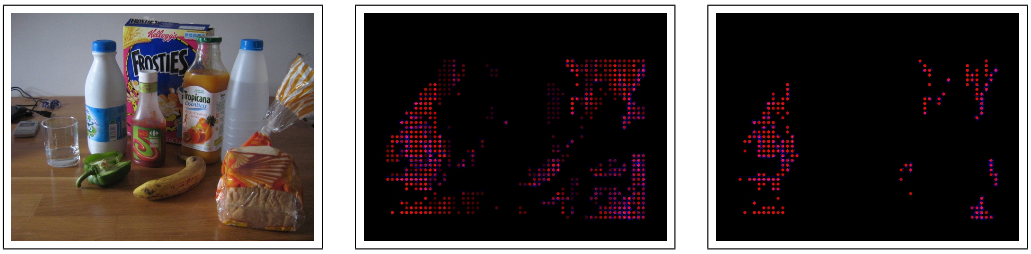
Figure 4.3: Figure (a) shows a sample image with a semi-transparent object. (b) The result of the algorithm after the first iteration. (C) The result of the algorithm after the second iteration.



(a) (b) (c) (d)

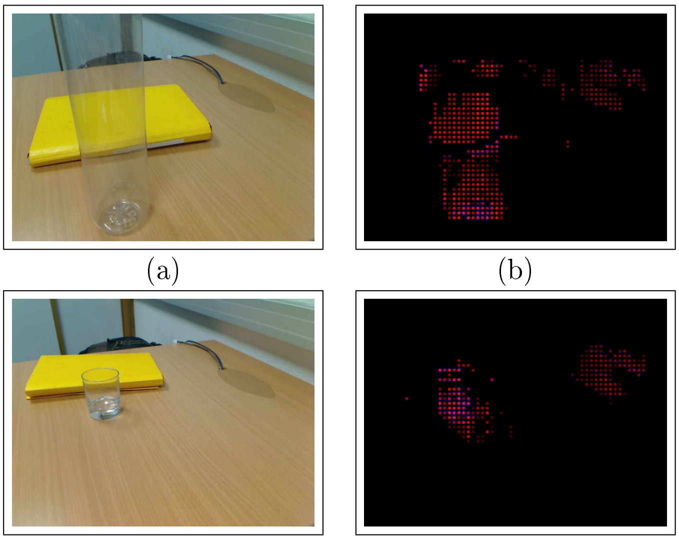
Figure 4.4: Figure (a) shows a sample image with a larger region R used for the detection. (b) The corresponding result of the algorithm. (c) The sample image with a smaller region R used for detection. (d) The corresponding result of the algorithm.

lem to solve. Our precision is better than the precision reported by McHenry and Ponce [McHenry 2006], i.e. 77.03%. We requested the author of [McHenry 2006] to provide us with the image dataset they have used, but we did not get any reply. So, we made our own dataset with images containing several object scenarios and some of which may fail to get detected by the algorithm presented in [McHenry 2006]. As our algorithm is based on points, the information at each point is independent from the structure of the other points situated around. Therefore, if a semi-transparent object has a few opaque regions on it, the corresponding opaque points will get filtered out by the algorithm. This is one of the reasons for the recorded recall rate



(a) (b) (c)

Figure 4.5: Figure (a) shows a sample image with several objects. (b) The result of the algorithm with a lower threshold. (C) The result of the algorithm with a higher threshold.



(a) (b) (c) (d)

Figure 4.6: Figures (a)-(d) show a few semi-transparent objects and the corresponding detection results. The images should be viewed in color.

to be slightly lower. This in a way acts as an advantage to filter out opaque parts of the transparent objects.

Figures 4.3 show the result of another experiment conducted on a semi-transparent object. We see that some points of the glass are not detected. This is due to the presence of a slightly darker shadow behind the glass. As mentioned earlier, we used the two-step region selection to evaluate our approach. Although it works quite well, it may not be the optimal method to get the best precision and recall rates. Figure 4.4(a) shows a sample image of a semi-transparent object with a region R selected by the red points. We see that the result (Figure 4.4(b)) is quite noisy. On the other hand when a region R as shown is the Figure 4.4(c) is used, we get a much better output (Figure 4.4(d)). Therefore, we find that the 2-step region

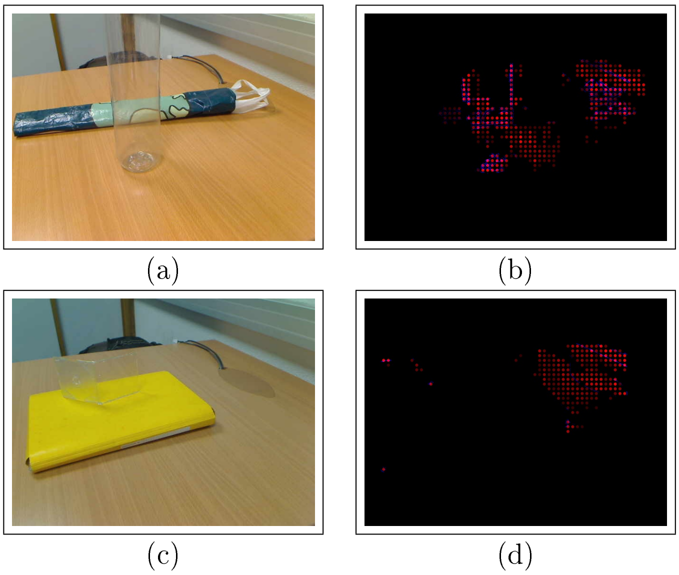


Figure 4.7: Figures (a)-(b) show an example of a false detection by the algorithm. Figures (c)-(d) show an example of a miss-detection of the semi-transparent object present in the image.

selection may not be the optimal solution for all cases. We will look into this aspect as a future improvement to the algorithm.

Figure 4.5(a) shows an image with several types of objects. We can find in Figure 4.5(b) that the semi-transparent glass on the left and the transparent region of the glass bottle on the right are detected with a good recall rate but with several false positives. As the result is probabilistic, on increasing the threshold, points with lower probability are filtered out and we find the result to be much better than earlier (Figure 4.5(c)). Figures 4.6 show results of a few more experiments conducted on semi-transparent objects. Figures 4.6(a) shows a thin plastic bottle placed in front of an opaque object. This is an example where intra-region clustering takes place. Similarly, a glass placed in front of the obstacle is also detected as shown in Figures 4.6(c)-(d). We can observe from both the results that a small patch on the top right of the image is falsely detected as semi-transparent. This is due to fact that the patch is similar to the background and could be considered as its distorted form. And, as the algorithm is point-based, the patch gets detected as a semi-transparent object. Although we considered these as false positives while quantifying the results, it may entirely not be a disadvantage to pick out such patches. One such situation which would produce a very similar patch is an oil or water spill on the table. As a conclusion, the effectiveness of the algorithm lies in detecting any media that creates a percept of transparency. Figures 4.7(a)-(b) show another example of a false detection where the patch as discussed earlier is aided by the white-plastic close to it which got detected as highlights in the image. Figures 4.7(c)-(d) show an example of a complete miss-detection of the semi-transparent object in the image.

Chapter 5

Semi-Transparent Obstacle Avoidance for a Mobile Robot

Contents

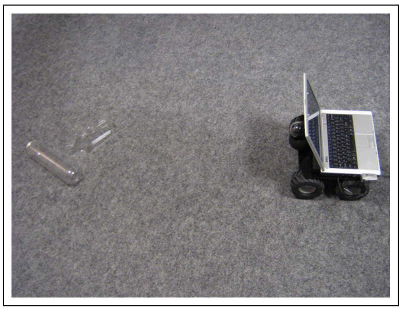


Figure 5.1: A navigating robot with semi-transparent obstacles in its path.

Obstacle avoidance forms the primary yet a challenging task in mobile robot navigation. With the increasing usage of objects made of glass, plastic etc., it becomes necessarily important to detect this class of objects too while building a robotic navigation system. Figure 5.1 shows a scenario where the robot has two semi-transparent obstacles on its path. In addition, water and oil spills on the floor are also some of the important examples that fall under the class. A few of the major constraints when it comes to robotic vision algorithms are the computational time and system cost i.e., the algorithm has to run in real-time with limited resources. The approach discussed so far in the previous chapters for detecting semi-transparent objects, although being efficient, is time consuming. In this chapter we discuss modifications made to the approach to reduce the computational time with a slight compromise on the accuracy of the result.

5.1 Modified Collective-Reward Based Approach

In the collective-reward based approach discussed in the previous chapters, for every point inside the hypothetical region R, k best points outside the region are determined using the support fitness functions. These k connections are used to generate appropriate rewards determined by the feature-reward functions. The rewards collected from all the k connections are weighted with their respective fitness values and the resulting total average-reward is found. So, we see that in-order to determine the best k points, we have to calculate the fitness values for each connection made between a point inside and a point outside the region R. To get a general notion about the computations involved, let us consider a region R of dimension 200x200 in a 640x480 image. The number of all possible connections is equal to $200 * 200 * (640 * 480 - 200 * 200) = 1.0688\text{x}10^{10}$. For each of these connections a fitness-value has to be computed which takes a total processing time in the order of minutes when run on a Pentium IV machine. To make it computationally tractable the point nodes considered in the algorithm are sampled as 1 node to 10 pixels in the image. Therefore, for a 640x480 image we have $20*20*(64*48-20*20) = 1.0688\text{x}10^6$ connections. The computational time after determining the fitness values on a Pentium IV machine was found out to be in the order of a second. Adding on this the time required to run the feature reward functions and the rest of the code takes around tens of seconds to execute. But in a robotics system, the complete execution should ideally take less than a second to make it reactive in real-time.

We see from the above discussion that most of the processor time is spent in finding out the best k connections for every point inside the region R. The main motive behind finding out the best connections is to perform a good comparison between the points inside and outside the region. In order to improve the computational speed we resorted for the k points selection based on random-distribution modeled using the clusters fitness function as discussed in the Chapter 3. All the point-nodes are stacked based on the cluster they fall into. The distribution model is generated for each cluster using the cluster fitness function. This model determines how many points (of the k points) have to be selected from each of the cluster stacks. The k exterior points for each interior point are found by collecting the corresponding number of points randomly from each cluster stack. This ensures that each interior point has more connections with the points of the same cluster and lesser for slightly different clusters and even lesser with the points of largely different clusters. Because points in an individual cluster stack are selected uniformly randomly, their spatial positions in the image could be distributed anywhere in the cluster. As the correlations are much better for the points that are close, we made use of the rejection sampling in order to limit the random selection to closer distances. The mask for the rejection sampling for each interior point p_i is given by a rectangle with dimensions equal to the region R and centered at the point $(p_i.x, p_i.y + 40)$, where $p_i.x$ is the x-coordinate and $p_i.y$ is the y-coordinate of the interior point p_i. An offset of 40 is selected so as to avoid the distortion due to the perspective blur by taking more points in the front. Figure 5.2(a) shows an image with a semi-transparent object placed on an opaque object. The blue rectangle is the hypothetical region R and the green rectangle is the mask for the rejection sampling. The points belonging to the same cluster are shown by the same color in the figure. We can see that the interior point which is highlighted by a red circular boundary belongs to the cluster

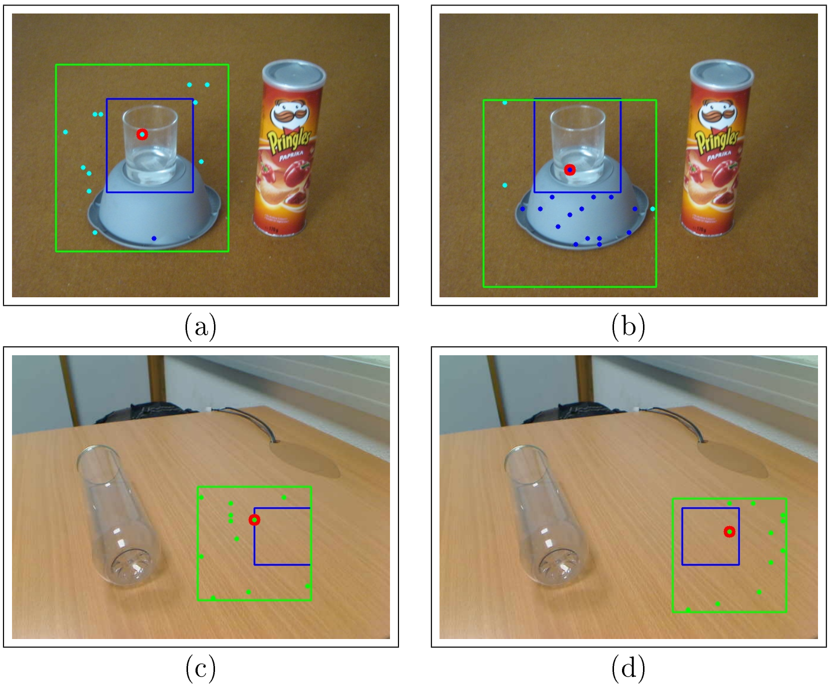
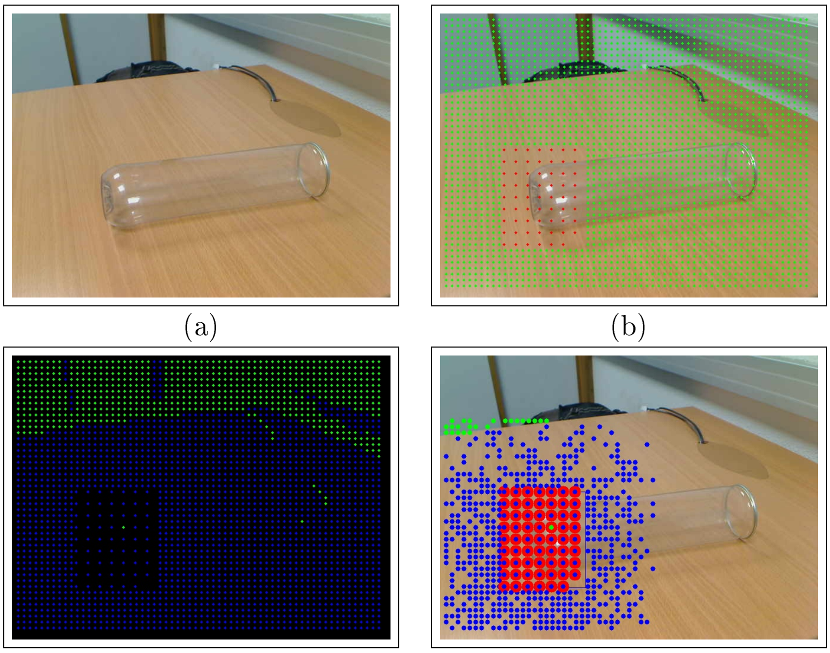


Figure 5.2: Figures (a)-(b) illustrate the selection of k exterior-points based on random-distribution modeled using clusters fitness function. Blue rectangle denotes the region R and green rectangle denotes the rejection sampling mask. The interior point p_i is highlighted by a red boundary for visualization. The color of a point indicates the cluster it belongs to. (a) Interior point belongs to green cluster, therefore majority of the exterior points selected are from green cluster (background). (b) Interior point belongs to blue cluster, therefore majority of the exterior points selected are from blue cluster (opaque object) and few from the green cluster. Figures (c)-(d) illustrate the rejection sampling mask selection.

similar to that of the floor. Therefore we find that the majority of the exterior points selected are of the same color and only few points are selected from the cluster belonging to the opaque object which is indicated by blue color. Figure 5.2(b) shows a similar image with a different interior point p_i. The highlighted interior point now belongs to the cluster of the opaque object, therefore the majority of the exterior points selected are from the opaque object and only few are from the background. Figures 5.2(c) and 5.2(d) illustrate the random selection of the exterior points based on the rejection sampling with respect to the mask (green rectangle). In the Figure 5.2(c) the interior point which is at the top left has the mask situated such that the majority of the points taken are close to it. And in Figure 5.2(d) the mask is at a different location collecting points on the right side of the region R.

The above method of finding k points has definitely reduced the computational overhead of calculating fitness values from a number $1.0688x10^6$ to just $k * 20 * 20 = 400k$ for a region R with dimensions 200x200, where k is around $10 - 50$. As the



(a) (b) (c) (d)

Figure 5.3: (a) Figure shows a sample image with a semi-transparent object. (b) A rectangular region denoted by red points is selected. (c) The outcome of coarse clustering, with a lower point-node sampling ratio inside the region R. (d) k exterior-points selected for each interior point using the rejection mask. The points inside the region R are highlighted by red boundary for the sake of visualization. The color of the points indicate the cluster it belongs to.

main objective of the robot is to detect and approximately locate the object in the image, we may relax on the actual boundary and shape of the body and detect it as a blob. So we sampled the point nodes inside the region R as 1 node to 20 pixels and the point nodes outside the region R is left the same as earlier, i.e., 1 node to 10 pixels. It is important to have a higher sampling rate outside than inside. Figure 5.3(a) shows a sample image with a semi-transparent object. Figure 5.3(b) shows the sampled point nodes, the points colored red are interior points and the points colored green are exterior points of the region R selected in the the image. Red points are sampled 1-20 node-pixels and green points are sampled 1-10 node-pixels. The coarse clustering output of the image is shown in the Figure 5.3(c). Figure 5.3(d) shows the random selection of the points outside the region. We can see that the points are concentrated within a rectangular region by rejection sampling.

The next important task is the selection of the hypothetical region R. The main objective behind its selection is to check for the presence of semi-transparent objects in the entire image and also approximately localize the object's position in the image. The region selected has to be scanned across the image to detect semi-transparent objects in the entire image. For the localization of the object's position

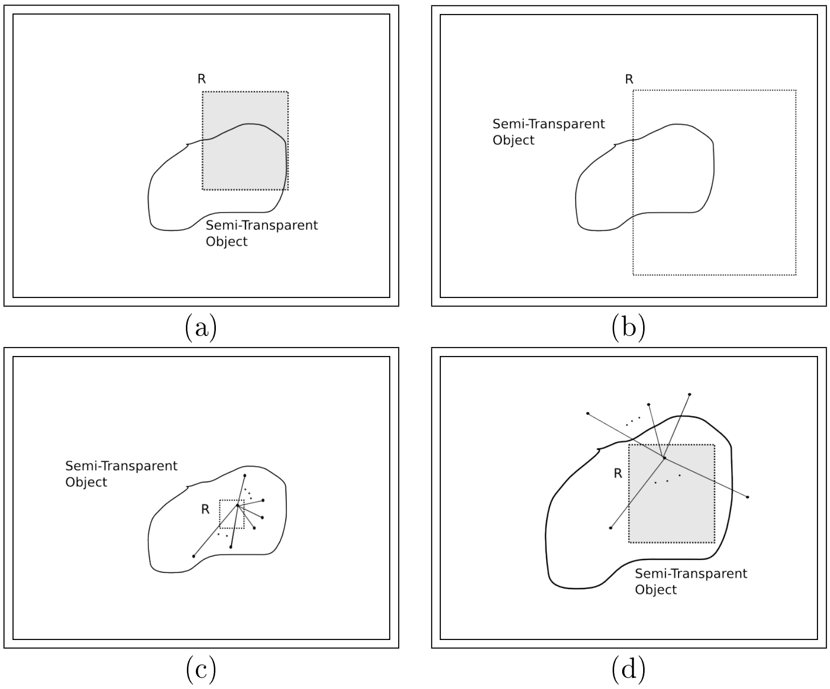
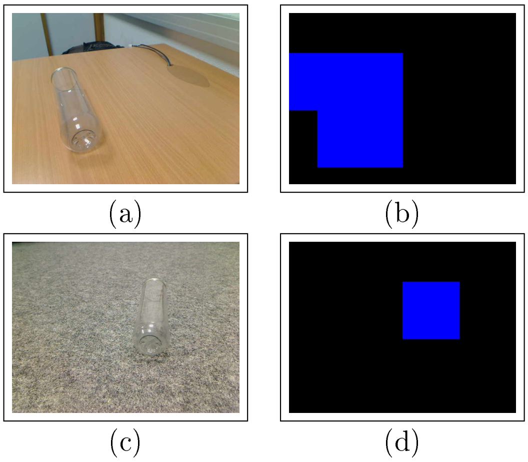


Figure 5.4: (a) Figure illustrates an image containing a semi-transparent object with an optimal sized region R. The region is shaded because the detection ratio crossed the threshold. (b) Figure illustrates the case when R is too large with a possible miss-detection and also leads to a large object localization error in the image. (c) Figure illustrates the case when R is too small. Possible miss-detection due to inter-transparent point comparisons. (d) Figure shows the advantage of random sampling over the deterministic approach for the region R as shown.

in the image, we note down all those regions where the ratio (detection-ratio) of the detected transparent points to the total number of points in the region is greater than 0.25. Figure 5.4(a) shows an illustration of a semi-transparent object and a region selected. The region is shown highlighted as the detection ratio has crossed the threshold of 0.25. Figure 5.4(b) shows an illustration with a larger region used for the detection. For this region there is a chance that the detection ratio might not cross the threshold resulting in a miss detection. It may also be ineffective in localizing the position of the objects in the image. Although a larger region will work well for larger semi-transparent objects as this will avoid the comparison between points within the transparent object. Figure 5.4(c) shows a similar illustration with a smaller region used for the detection. The interior part of an obstacle will not get detected as the comparisons are made between the points of the same object as shown in the figure 5.4(c). Therefore a region which is not too small or not too large is selected for the detection. We used a region of area equal to $\frac{1}{12}$ Image-area. It is also interesting to note that the random-selection approach has a slight advantage over the deterministic approach used in the Chapter 3 for the detection of inner regions of semi-transparent objects using a region as shown in the Figure 5.4(d). The deterministic approach would take the best points surrounding the region which turn out to be the closest points in the example shown in the figure. Therefore, the comparisons between the points of the semi-transparent object will be made resulting in a zero reward leading to a miss detection. But, with the random-selection approach, the points are randomly distributed about the region and therefore there is a better chance of detecting at least a few points inside the region meeting the threshold for the detection ratio. Once the object's position in the image is localized we make use of a heuristic closed loop turn maneuver algorithm for the robot to avoid the detected obstacles.

5.2 Experimental Results



(a) (b) (c) (d)

Figure 5.5: Figures (a)-(d) show a few semi-transparent objects and the corresponding detection results. The location of the object in the image is found as the highlighted rectangular regions in which the detection ratio has crossed the threshold.

In this section, we will present some of the experimental results conducted using the modified collective-reward based approach. We also carried out several experiments on a mobile robotic platform and tested the obstacle avoidance algorithm using the same approach. Figures 5.5 show the algorithmic response for the corresponding sample image. We can see in the Figure 5.5(b) that the semi-transparent object is detected and localized in the image using rectangular blue regions. A total of 12 regions are used for a single image and when the detection ratio crosses the threshold, the corresponding region gets highlighted as shown in the Figures 5.5(b) and (d).

Figure 5.6 shows the outputs of both the native approach and the modified one compared with each other. The native approach is more precise as expected than the modified approach. On the contrary the modified approach is much faster. The

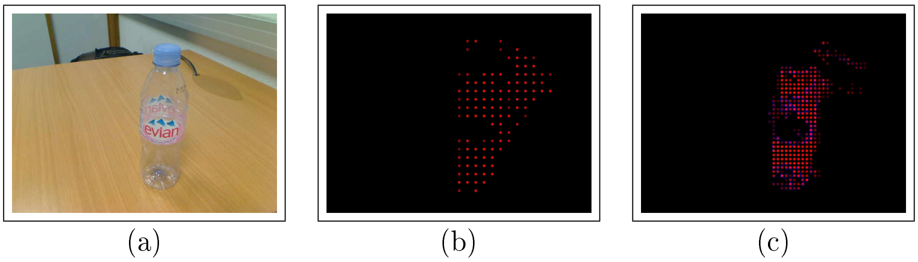


Figure 5.6: Figures (a)-(c) show a comparison between the outputs of both the native approach and the modified approach of the algorithm. (a) Sample input image (b) Result of the modified approach (c) Result of the native approach

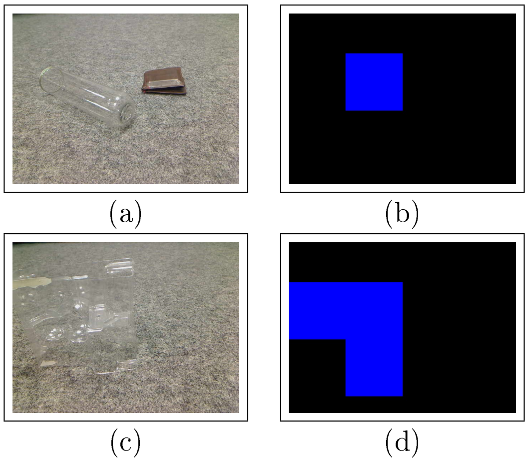
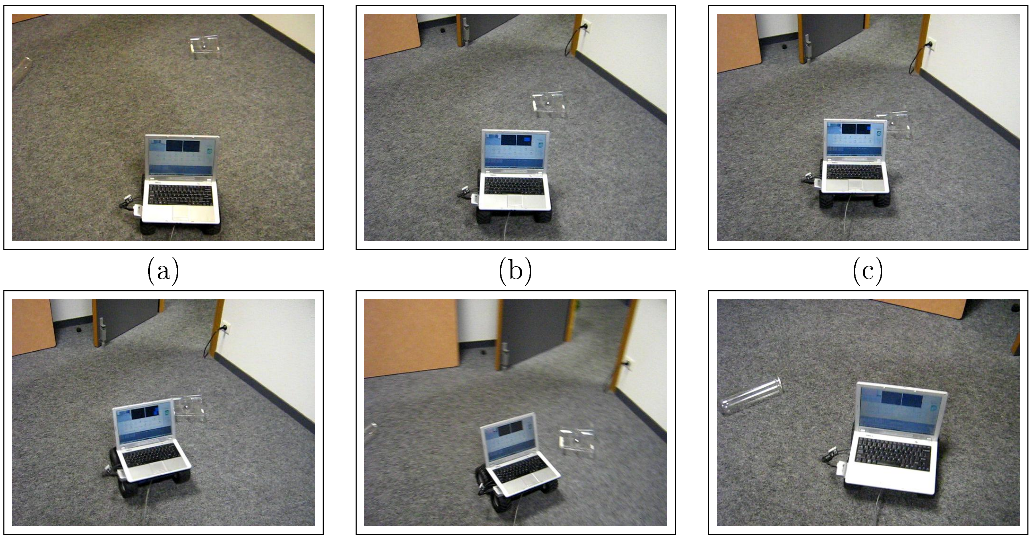


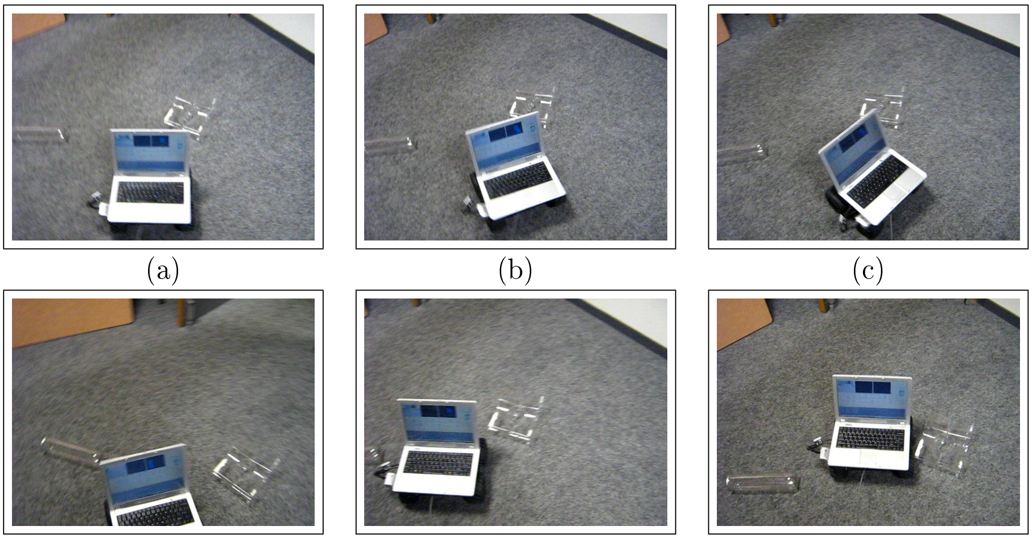
Figure 5.7: Figures (a)-(d) show a few semi-transparent objects and the corresponding detection results. Figure (b) shows that the algorithm only detects semi-transparent objects.

execution time for both the algorithms is found out to be equal to 10.274 seconds for the native approach and 1.671 seconds for the modified approach. Therefore the modified approach is approximately 6-7 times faster than the native approach. Figure 5.7(a) shows another input image with a semi-transparent object along with an opaque object. The result (Figure 5.7(b)) shows that only the semi-transparent object is detected. Figures 5.7(c)-(d) show another example-pair with a different object in the scene. We used a simple avoidance maneuver algorithm. The number of highlighted rectangular regions are counted on each half of the screen to determine the turn direction. The turn-trigger signal to the robot is generated as soon as a region corresponding to the bottom-most row gets highlighted. For example, in the Figure 5.7(d), we can see that 4 rectangular regions are highlighted with 3 in the middle row and 1 in the bottom row. Therefore, the robot receives a control signal to turn right. Figures 5.8 and 5.9 show snapshots taken from a video of the robot while performing the obstacle avoidance with semi-transparent obstacles on its path.



(a) (b) (c)
(d) (e) (f)

Figure 5.8: Figures (a)-(f) show snapshots taken from a video of the robot performing obstacle avoidance using the modified collective-reward based approach. The robot moves towards the obstacle on the right and then avoids it when a bottom rectangle region gets highlighted as shown in the figure (d).



(a) (b) (c)
(d) (e) (f)

Figure 5.9: Figures (a)-(f) show snapshots taken from a video of the robot performing obstacle avoidance using the modified collective-reward based approach. (b) The robot detects the obstacle on the right. (c) It turns left and moves towards the obstacle on the left. (d) The bottom-left rectangular region gets highlighted, it then turns right. (e)-(f) The robot passes through the space between the two obstacles.

Chapter 6

Conclusions and Future Work

We proposed a new approach to detect the presence of transparencies in an image using the collection of rewards via support fitness functions. This approach makes use of the dependency between the points belonging to the transparent object and the points that are situated around. This accounts for both the refracted background and reflected foreground about the semi-transparent object. The method uses a hypothetical region to determine the transparent objects inside it. The hypothetical region can either be manually selected by the user or can be optimally automated. A two-step region selection method was discussed in this regard. An optimal region-selection method forms one of the immediate goals of future work. The algorithm is point-based and the detection at each point is independent from the structure of other detected transparent points of the object (except for the immediate neighbors which are used in the Nearest Neighbor Transparency post-processing function, refer to Chapter 3). A better result may be achieved by combining the information regarding the structure of the detected points. We submitted the above collective-reward based approach for the detection of semi-transparent objects to the British Machine Vision Conference, 2009 (BMVC'09).

The native algorithm was modified to suit to robotic applications of which obstacle avoidance was discussed and experimented in the work leading to the thesis. We look forward to carry out several other important applications such as transparent door detection that helps in localization and mapping of the robot looking through a glass door. Obstacle avoidance along with the localization helps the robot to safely navigate along the corridors made of glass without colliding. The collective-reward based approach could well be used to detect other kinds of transparent media like dim shadows, water-spill or any material that creates a percept of transparency.

Bibliography

[Adelson 1990] Edward H. Adelson and P. An. *Ordinal characteristics of transparency.* In in Proc. AAAI workshop on Qualitative Vision, pages 77–81, 1990. 2

[Androutsos 1999] D. Androutsos, K. N. Plataniotis and A. N. Venetsanopoulos. *A novel vector-based approach to color image retrieval using a vector angular-based distance measure.* Comput. Vis. Image Underst., vol. 75, no. 1-2, pages 46–58, 1999. 8

[Ben-Ezra 2003] Moshe Ben-Ezra and Shree K. Nayar. *What Does Motion Reveal About Transparency?* Computer Vision, IEEE International Conference on, vol. 2, page 1025, 2003. 3

[Ford 1998] Adrian Ford and Alan Roberts. *Colour space conversions*, 1998. 8

[Hata 1996] S. Hata, Y. Saitoh, S. Kumamura and K. Kaida. *Shape Extraction of Transparent Object Using Genetic Algorithm.* Pattern Recognition, International Conference on, vol. 4, page 684, 1996. 3

[Klinker 1990] Gudrun J. Klinker, Steven A. Shafer and Takeo Kanade. *A physical approach to color image understanding.* Int. J. Comput. Vision, vol. 4, no. 1, pages 7–38, 1990. 7

[Liu 2008] Renting Liu, Zhaorong Li and Jiaya Jia. *Image partial blur detection and classification.* In Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, pages 1–8, 2008. 8

[MacKay 2003] David MacKay. *Chapter 20. An Example Inference Task: Clustering*, 2003. 13

[McHenry 2005] Kenton McHenry, Jean Ponce and David Forsyth. *Finding Glass.* Computer Vision and Pattern Recognition, IEEE Computer Society Conference on, vol. 2, pages 973–979, 2005. 4

[McHenry 2006] Kenton McHenry and Jean Ponce. *A Geodesic Active Contour Framework for Finding Glass.* Computer Vision and Pattern Recognition, IEEE Computer Society Conference on, vol. 1, pages 1038–1044, 2006. 4, 26

[Murase 1992] Hiroshi Murase. *Surface shape reconstruction of an undulating transparent object.* pages 213–217, 1992. 11

[Schechner 1998] Yoav Y. Schechner, Nahum Kiryati and Ronen Basri. *Separation of transparent layers using focus.* In Proc. ICCV, pages 1061–1066, 1998. 3

[Singh 2002] Manish Singh and Barton L. Anderson. *Toward a perceptual theory of transparency.* Psychological Review, vol. 109, pages 492–519, 2002. 2, 9

[Singh 2003] Manish Singh and Xiaolei Huang. *Computing Layered Surface Representations: An Algorithm for Detecting and Separating Transparent Overlays.* Computer Vision and Pattern Recognition, IEEE Computer Society Conference on, vol. 2, page 11, 2003. 3

[Singh 2006] Manish Singh and Barton L. Anderson. *Photometric determinants of perceived transparency.* Vision Research, vol. 46, pages 879–894, 2006. 2

[Wexler 2002] Yonatan Wexler, Andrew. W. Fitzgibbon and Andrew. Zisserman. *Image-based environment matting.* In EGRW '02: Proceedings of the 13th Eurographics workshop on Rendering, pages 279–290, Aire-la-Ville, Switzerland, Switzerland, 2002. Eurographics Association. 3