Collective-Reward Based Approach for Detection of Semi-Transparent Objects in Single Images

Varun Raj Kompella, Peter Sturm

PERCEPTION Project
INRIA Grenoble Rhône-Alpes and Laboratoire Jean Kuntzmann, Grenoble, France

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 paper is to provide a research 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 important information that can be used in a robot's navigational strategies such as obstacle avoidance, detection of oil/water spills on the floor, localization, etc. 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. Several experiments were conducted over different scenarios to test the efficacy of the algorithm.

Key words: Collective-reward, Object detection, semi-transparency, transparency, glass.

The final version of this article has been published in Computer Vision and Image Understanding, 116(4):484–499 (2012) published by The Academic Press. This version does not differ significantly from the final version.

Email addresses: varunrajk@gmail.com (Varun Raj Kompella), peter.sturm@inrialpes.fr (Peter Sturm)

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. However, several objects like glass do give us a perception of transparency by posing a few deterministic cues such as highlights, texture distortion, intensity variation etc. to the observer and passing most of the light through it. 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). In this paper we discuss a collective-reward based approach for detecting such semi-transparent objects from a single image. The method is thoroughly tested for its efficacy by a 50 image-dataset containing several scenarios of semi-transparent objects made of glass, plastic etc.

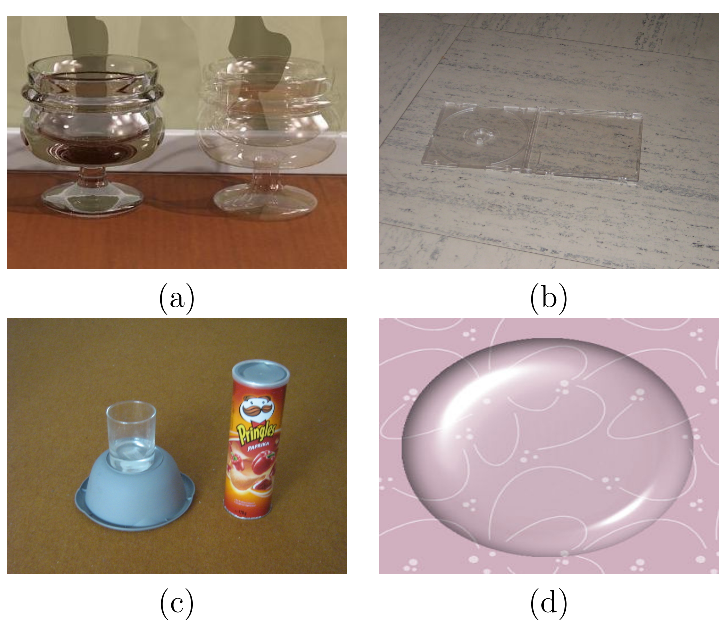


Figure 1: The figures show images containing semi-transparent objects. (a) Pair of glasses. (b) A plastic-case (c) Glass with opaque objects (d) Synthetic water drop.

1.1. Related Work

In this section, 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 researchers studying the phenomenon of transparency, gestalt psychologist Metelli has been credited for making important and influential contributions to the theory of perceptual transparency [14]. 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 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.

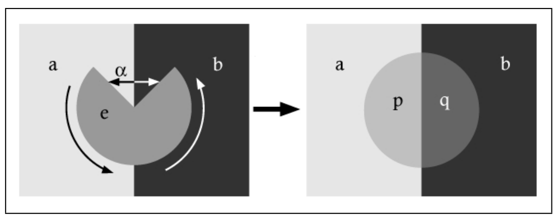


Figure 2: The figures illustrate Metelli's model of transparency using a rotating episcotister.

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

$$p = \alpha a + (1 - \alpha)t \tag{1}$$
$$q = \alpha b + (1 - \alpha)t \tag{2}$$

Two "qualitative constraints" were proposed for predicting the percept of transparency and they are:

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

The magnitude constraint was later found out by Singh and Anderson [15] to be inadequate in predicting the percept of transparency. The locus of

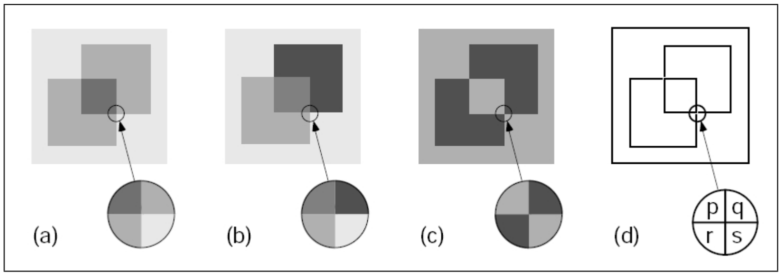


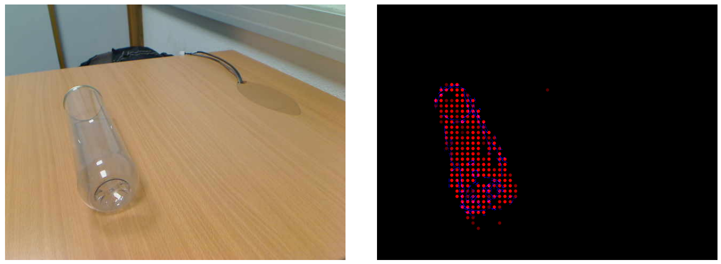
Figure 3: The figures(a)-(c) illustrate different types of X-junction. (a) Non-reversing X-junction (b) Single-reversing X-junction (c) Double-reversing X-junction. The conditions on luminance values shown in (d) are used to categorize the X-junction as one among the three types.

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 [1] used a linear model for the intensity of a transparent surface to achieve relationships between the X junctions at the boundary of transparent objects. These relationships categorize the X junctions leading to interpretations that support or oppose transparency. Figure 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 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. This is called a "non-reversing" junction because both edges retain their sign (Figure 3(a)). Figure 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 a double-reversing junction does not support transparency.

Transparency and its related problems have received relatively less attention in the computer vision research. Singh and Huang [16] discussed about the 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.* [13] have used the concept of depth from focus along with reconstruction to separate such overlays. Wexler *et al.* [17] 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 [3] 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.* [5] have proposed another approach in extracting the shape of transparent objects. A comparison was performed over a real and a simulated image by tracing a slit light-line along an ordinary board on which the transparent object was placed. The error evaluation was 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 [10] 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 [9] with a region-based approach along with the edge information to classify regions as 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 was used to indicate how close a region looks like a glass-covered region of the other. A region-based segmented image was used as an input to the algorithm. One of the issues reported by the authors was that an initial segmentation may merge some parts of the transparent object with parts of the 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



(a) (b)

Figure 4: (a) The figure shows a sample image with a semi-transparent object. (b) The final result of the algorithm.

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.

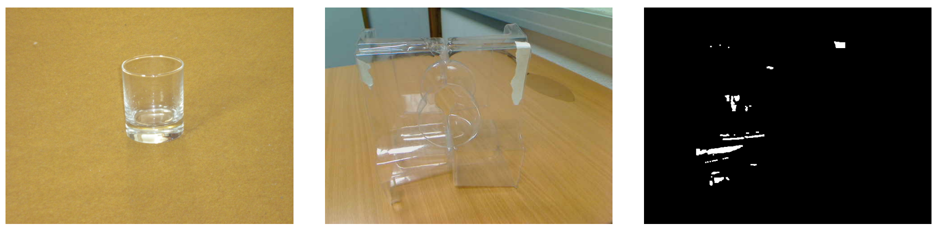
In this paper, we present an algorithm for the automatic detection of semi-transparent objects using the information available from a single image. In this regard we propose a method called the collective-reward based approach to achieve the detection and localization of the object's position in an image. The underlying principle is the fact that the pixels corresponding to a semi-transparent object have features which are similar to the surrounding pixels due to refraction and reflection of light on the object's surface. Figure 4(a) shows an example image which when fed to the algorithm has the result as shown in the Figure 4(b). The collective- reward based approach can be extended to fit onto a robotic system to detect glass doors and transparent obstacles present in the scene for a better mapping and navigation. It may also be used to detect and avoid water or oil spills on the floor.

The paper is organized as follows. In Section 2, we discuss the feature-cues used in the algorithm that are related to the transparent objects. Section 3 discusses details about how the feature-cues are quantified either by offline trained or hand-set reward functions. The tuning parameters used are then fixed for generating experimental results. The collective-reward based approach is presented in-detail in the Section 4. Section 5 discusses about the results of the experiments conducted over several sets of images captured with a web cam and from the Internet. Finally, we conclude the paper by providing some insights on the various applications of the algorithm and also discuss some of our future work in Section 6.

2. Feature-Cues

This section 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 Section 3.

2.1. Highlights and Caustics



(a) (b) (c)

Figure 5: (a) The figure shows the presence of caustics on a semi-transparent object (glass). (b) The figure shows the presence of highlights on a semi-transparent object (plastic case), and the 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 like glass are usually highly specular and refractive, therefore the presence of highlights and caustics increases the probability of a possible transparent material around (Figure 5(a)). Several methods exist in the literature discussing the detection of highlights [6]. We have used HSV color space [4] and a method similar to the one discussed in [2]. Figure 5(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 5(c).

2.2. Color

A near transparent object would produce an almost negligible distortion to the color of the background. On the other hand, semi-transparent objects like glass, plastic, etc. generally have impurities and due to specular reflections, the background color is slightly distorted (Figure 6(a)). We made use of the $YCrCb$ color model [4] as the distortion was relatively higher in Cr (Figure 6(b)) and Cb (Figure 6(c)) color channels in comparison to the

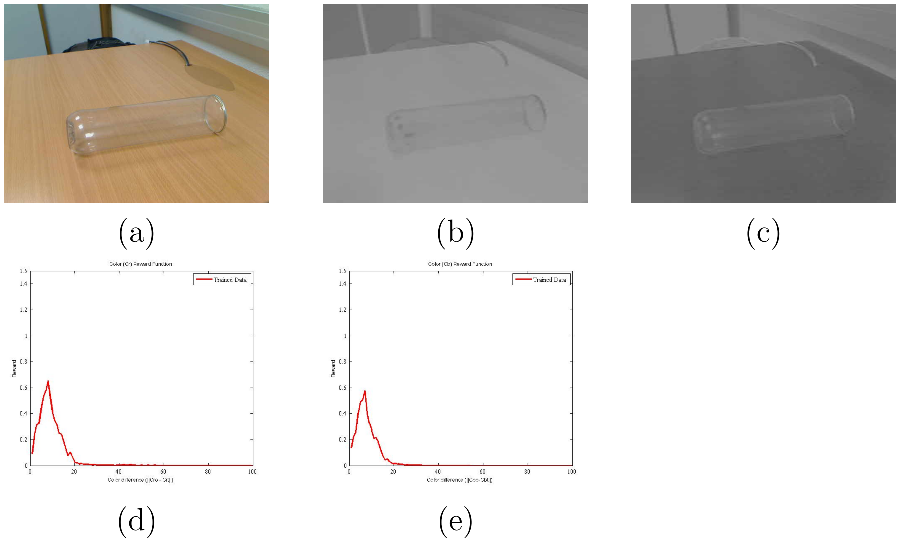


Figure 6: (a) The figure shows an image containing a semi-transparent object. (b) Its corresponding Cr color channel and (c) Cb color channel image. (d) Offline trained function for Cr color channel and (e) Offline trained function for Cb color channel (see text)

Hue channel of the HSV color model. Figures 6(d)-(e) show offline trained functions for both the color (Cr and Cb) components. They relate color differences between a point and a close-by-point on the background to the probability that the first point belongs to a semi-transparent object. The distortion can be seen in the graphs where the reward (discussed in the later sections) or the probability is higher for the color differences that are neither large nor very close to zero. This corresponds to the intuition that semi-transparent objects *slightly* alter the color of the background behind them. Details about the training are discussed in the Section 3.

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 [7]. Therefore these pixels have relatively lower *saturation* values (Figure 7). This can be observed in the offline trained function as shown in the Figure 7(c). The probability (reward) is higher for smaller positive differences between the points on transparent objects and surroundings.

(a) (b) (c)

Figure 7: (a) The 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. (c) Offline trained function for saturation channel.

2.4. Intensity

(a) (b)

Figure 8: (a) The 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. (b) The figure shows an illustration of how the cross-correlation measure is determined in an image containing a transparent object.

Intensity plays a major role for backgrounds with texture. We made use of Michelson's contrast constraint, which uses the information of the maximum and the minimum intensities in the neighborhood (Eqn (3)), as it has been shown by Singh and Anderson [14] that transparency lowers its value:

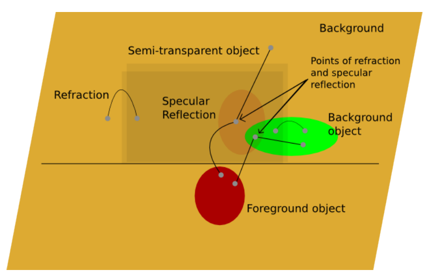
$$C = \frac{I_{max} - I_{min}}{I_{max} + I_{min}} \tag{3}$$

Figure 8(a) shows a gray scale image of a semi-transparent object. We can observe the slight reduction in the difference in the intensities within a small neighborhood of the texture appearing on the semi-transparent object.

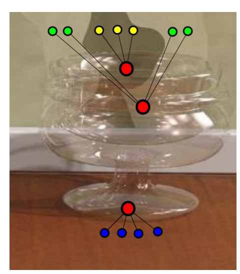
2.5. Cross-Correlation Measure

Cross-correlation is a measure of how well two signals match with each other. All the feature cues discussed till now were used via pixel comparisons, but the distortion produced by a semi-transparent object can also be captured by a region analysis. A small window used as a template was slided over a small rectangular region as shown in the Figure 8(b). The normalized cross-correlation score was calculated at each point. The maximum and minimum of the result were found. These values were used as another cue to determine the probability of the presence of a transparent object. Normalized cross-correlation values (maximum and minimum) are higher and the difference between them is smaller for the points that belong to similar regions. As glass produces a slight distortion effect on the background, the values are relatively lower and the difference between the maximum and minimum is slightly larger. On the other hand two non-similar patches report for even lower values for both the maximum and the minimum. In order to reduce the effect of noise and improve the result, $YCrCb$ color space channels of the image were fed as an input.

3. Feature Reward Functions



(a)



(b)

Figure 9: (a) The figure illustrates the contribution of features via refraction and specular reflection. The figures (b) shows a sample image illustrating the correspondences between a point inside and outside the semi-transparent object.

Semi-transparent objects typically contain the distorted features of what lies behind the object [11]. However we know that 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 the pixels corresponding to the background in addition to those of the foreground. Figure 9(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 regions 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. Figure 9(b) shows a sample image with correspondences drawn between points inside and outside the semi-transparent object.

Feature reward functions are probability density functions of the semi-transparent points for a given feature distortion value. Feature distortion is used because all the feature cues, discussed in the Section 2, are based on the distortion produced by the semi-transparent objects over the background. The feature distortion is either calculated as a difference d in the feature-values or a difference-measure in other attributes of the points belonging to the semi-transparent object and their counterpart background points. Therefore, the reward functions emphasize the difference between the semi-transparent object vs background over background vs background or opaque vs background regions.

This section discusses about the feature reward functions used and their construction details. An offline training was carried out to construct reward functions for the following feature-cues: Cr, Cb and $saturation$. As we are not aware of any definite form for these feature-cues, we constructed the functions from the population density of the points in the sample space. While, for the remaining features such as $intensity$, $cross-correlation$ and $highlights$, we used hand-set models for their reward functions. These are discussed later in this section.

Offline-trained feature reward functions: To construct reward-functions for features f belonging to (Cr, Cb and $saturation$), we calculated the population of points belonging to the semi-transparent objects from a sample-set for a given feature difference d. So, the reward function Rw is given by

$$Rw(d) = \left(\frac{n_d^{tr}}{n_d^{tr} + n_d^{bg}} \right), \ d \in (0, G) \tag{4}$$

where, n_d^{tr} and n_d^{bg} are the number of points belonging to the semi-transparent object and background respectively for a given feature difference d. The sample-set is equal to the sum of n_d^{tr} and n_d^{bg}. The interval $(0, G)$ is the range-interval of the difference d for a given feature f.

(a) (b)

Figure 10: (a) The figure shows an example of how the rectangular samples were collected for offline training of the feature cues. (b) The figure shows an example of how the negative training was carried out. ∼tr indicates a negative sample set.

To construct the sample set, we collected points from two arbitrarily selected rectangular regions one each belonging to the semi-transparent object (T_{tr}) and the background (T_{bg}) (Figure 10(a)). The sample set is the union of both the regions, that is, $T_{tr} \cup T_{bg}$. Separate regions are selected only for the sake of convenience. The quantities n_d^{tr} and n_d^{bg} in (4) can be found from the histograms of the feature-difference values for the points belonging to the semi-transparent object and the background respectively. To construct such histograms, we selected another similar region (T_d) from the similar background (Figure 10(a)). We compute the feature differences between each point in the sample set (semi-transparent object and background) and the points in the region (T_d) (similar background). Therefore, for an interior point $p_i \in (T_{tr} \cup T_{bg})$ and an exterior point $p_e \in T_d$, we have the following histograms

$$H_d^{tr,p_i} = \{n_d \mid d = |f(p_i) - f(p_e)|, p_i \in T_{tr}, \forall p_e \in T_d\} \tag{5}$$
$$H_d^{bg,p_i} = \{n_d \mid d = |f(p_i) - f(p_e)|, p_i \in T_{bg}, \forall p_e \in T_d\} \tag{6}$$

These histograms are averaged over all the points p_i in the sample set to get a better representative of individual histograms. Therefore, we have

$$H_d^{tr} = \frac{1}{\aleph(T_{tr})} \sum_{\forall p_i \in T_{tr}} H_d^{tr,p_i} \tag{7}$$

$$H_d^{bg} = \frac{1}{\aleph(T_{bg})} \sum_{\forall p_i \in T_{bg}} H_d^{bg,p_i} \tag{8}$$

where, $\aleph(x)$ denotes the cardinality of the set x. Substituting (7) and (8) in (4), we get,

$$Rw(d) = \left(\frac{H^{tr}(d)}{H^{tr}(d) + H^{bg}(d)} \right), \; d \in (0, G) \tag{9}$$

The reward function from (9) is updated and averaged over several positive sample-sets of different semi-transparent objects for each feature $f \in \{Cr, Cb$ and $saturation\}$ and is denoted by (Rw_f^+). A positive sample set is defined as a set that contains at least a few points belonging to a semi-transparent object. While a negative sample set is defined as a set that does not contain points of semi-transparent objects. The training was also done with negative samples by considering regions taken from distinguishable opaque objects giving a negative reward function $(-Rw_f^-)$ (Figure 10(b)). We used a total of 30 sample sets from different images to carry out the complete training. The final reward function used for prediction for each feature $f \in \{Cr, Cb$ and $saturation\}$ is given by (10).

$$Rw_f(d) = \frac{Rw_{f1}^+(d) + Rw_{f2}^+(d) + ... - Rw_{fk}^-(d) - ... - Rw_{fN}^-(d)}{N}, \; d \in (0, G) \tag{10}$$

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 robustly indicate the color attribute invariant to intensity. The absolute difference between the median of a 3x3 neighborhood centered at two points is used as an argument that is passed to the reward function (10). Rewards for each color component Cr and Cb are found from the individual reward functions and the product of both the rewards is reported. Figures 11(c) and 11(d) show the reward functions for Cr and Cb features respectively.

Saturation: The above training was carried out for the saturation channel with a slight modification, where a positive feature difference $d = |f(p_i) - f(p_e)| \, H[f(p_e) - f(p_i)]$ was used instead in (5) and (6). Where, $H[n]$ denotes a Heaviside step function. This was done in order to account for the fact that *saturation* values of semi-transparent objects are lower compared to the background. The *saturation* reward function (see Figure 11(b)) is found from (10) for a feature-difference $d = (Sat(p_e) - Sat(p_i))$.

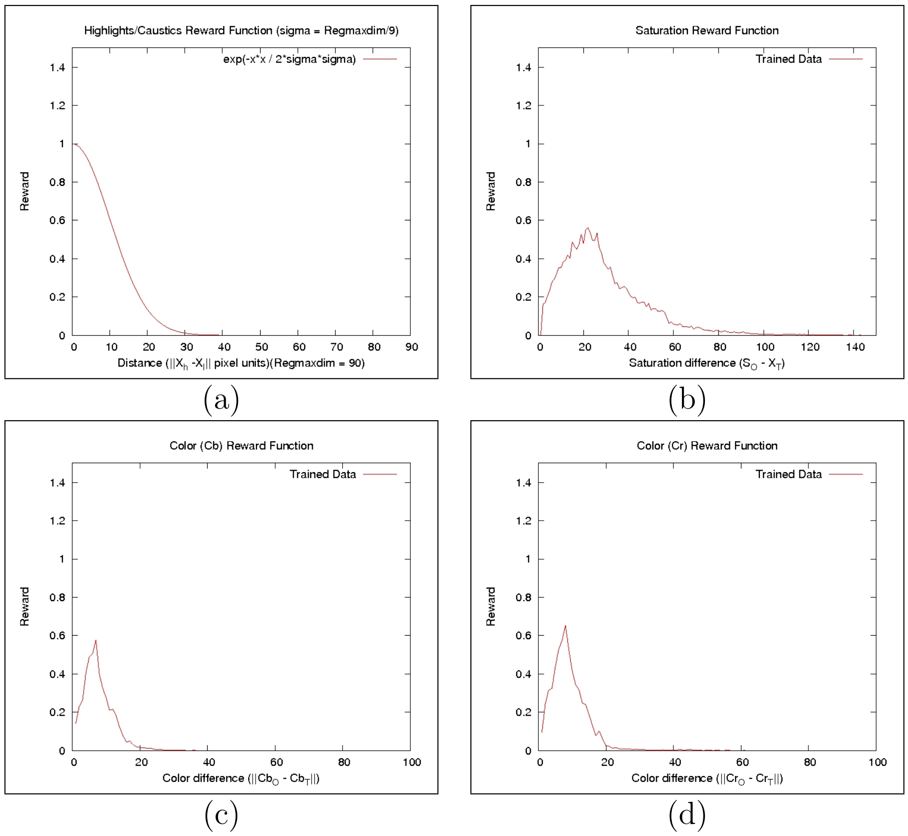


Figure 11: The figures show graphs of (a) Highlights reward function (b) Saturation-reward function (c) Cb-reward function and (d) Cr-reward function.

Hand-set feature reward functions: While the reward functions for the features Cr, Cb and *saturation* were trained using the offline training method, a hand-set model is used for the features *highlights*, *intensity* and $cross - correlation$.

Highlights and Caustics: This is a hand-set reward function $Rw_{high}(d)$ given by the Gaussian function as shown in the Figure 11(a), with the euclidean distance between a point and the closest highlight-point as an argument.

Intensity: The maximum and minimum intensities are calculated in a $K \times K$ neighborhood centered at different points, say p_i and p_e . These values are used to calculate Michelson's contrast. Michelson's contrast is defined as $C = \frac{I_{max}-I_{min}}{I_{max}+I_{min}}$. The difference $d = (I^{avg}_{p_e} - I^{avg}_{p_i})$ in the average intensities calculated in the neighborhood centered at p_i and p_e, is checked if it falls below a threshold ϵ and a reward is generated using the reward-function given by

$$Rw_{Int}(d) = \begin{cases} \frac{C_{p_e}-C_{p_i}}{C_{p_e}+C_{p_i}} & C_{p_e} \geq C_{p_e} > 0, |d| < \epsilon \\ 0 & otherwise \end{cases} \tag{11}$$

Cross-Correlation Measure: A small window of size $K \times K$ centered at a point p_i is used as a template to be slided over a rectangular region MxM centered at another point p_e with $(M > K)$. The window slides across the region and the normalized cross-correlation values (Maximum and Minimum) are calculated at each point. In order to reduce the effect of noise and improve the result, $YCrCb$ color space channels of the image are fed as an input.

4. Collective-Reward Based Approach

This section discusses details about the algorithm presented in the paper, which makes use of the feature reward functions as discussed in Section 3.

Feature reward functions give higher reward outputs for the points belonging to a semi-transparent object as they emphasize on the distortion created by the semi-transparent object over the background. Since, the boundary corresponding to the semi-transparent object is not known, an arbitrarily selected hypothetical region R is used to calculate the feature distortion values for each interior (p_i) and several exterior (p_e) points (with respect to R) in the image. The automatic selection of the hypothetical region is later discussed in the section 4.6.

Generally there may be only fewer exterior points similar to an interior point that are useful in characterizing whether the interior point belongs to a semi-transparent object or not. Therefore, a general aggregation scheme for rewards in such situations would be less fruitful as the negative rewards produced by non-similar exterior points would dominate over the positive rewards produced by similar ones. Besides, it is also computationally expensive to compute rewards over all the possible pairs of interior and exterior points. Therefore, we make use of *Support Fitness Functions* (discussed in Section 4.2) to find a limited k suitably-fit exterior points for each interior point, which can be used for collective reward generation and classification.

points lying on the similar background. Only these few exterior points, which are similar to the actual inaccessible point behind the glass, are useful in characterizing whether the corresponding interior point belongs to a semi-transparent object or not. Usually, we find situations where $(N-M-n_{bg}) >> n_{bg}$ and therefore, a general aggregation scheme for rewards in such situations would be less fruitful as the negative rewards produced by $(N - M - n_{bg})$ points would dominate over the positive rewards produced by n_{bg} points. Besides, it is also computationally expensive to compute rewards over all the possible pairs of interior and exterior points. Therefore, we make use of *Support Fitness Functions* (discussed in Section 4.2) to find a limited k suitably-fit exterior points for each interior point that can be used for collective reward generation and classification.

A *Collective-reward based approach* is the process of classifying a point by aggregation of the results found from a reward-generation scheme where, the point and its corresponding suitably-fit points participate. The suitably-fit points are found using *Support Fitness Functions* and the reward-generation scheme is a collection of complementary functions called the *Feature Reward Functions* that act on the features related to the semi-transparent object. Figure 12 shows the block diagram of the proposed algorithm discussed in this paper. Given an input image, a hypothetical region R is selected to set up connections between interior and exterior points. The term "connection" is used to denote an association made between the points for transporting information along it. The next block in the block diagram 12, which has two *Support Fitness Functions*, namely clusters function and distance function (refer Section 4.2 for details), generates fitness values to all possible pairs of connections between each interior point and all the exterior points. These fitness values are an input to the next block which selects the k best neighbors for each interior point based on the magnitude of the fitness values.

The k connections thus formed for each interior point are then tested further with the feature reward functions (see Section 3), as shown by the blocks numbered 1 to 5, to generate a reward. These reward values for all the connections between the interior and the exterior points are an input to the *Collective Reward and Classification* block. This block calculates a total reward called the collective reward at each interior point for each feature cue. The collective reward is equal to the weighted average of the individual rewards received through the connections with the corresponding connection fitness-values as weights. An ensemble of all the individual feature classifiers (reward functions) is formed to generate a strong classifier that outputs a final reward (refer Section 4.4 for details). The result is then passed through a test condition and then to post-processing functions, which are discussed

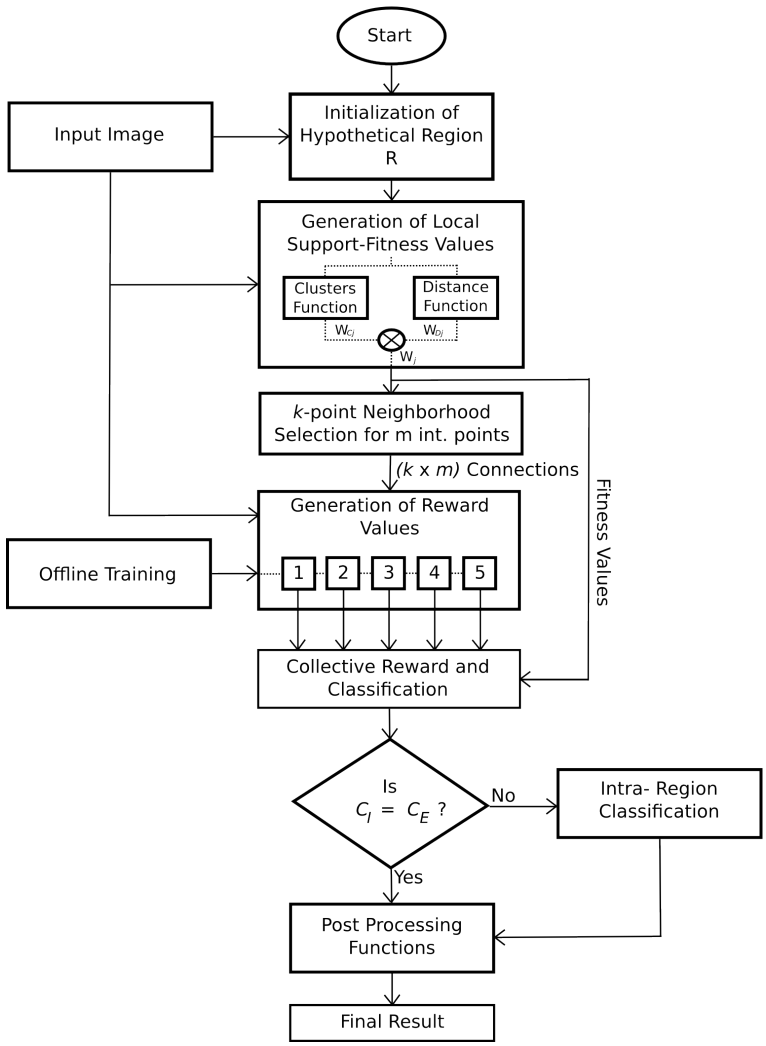


Figure 12: Block Diagram of the Algorithm

in Sections 4.5 in more detail, to generate a final output.

4.1. Point Sets and Problem Formulation

This section will present a formulation of the detection problem and also discusses the notations that will be used henceforth. Figure 13 shows a sample image illustrating the point sets. Let R_I and R_E denote the set of points lying interior and exterior to the region R respectively. Let P_{tr}

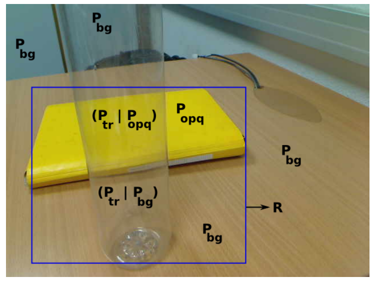


Figure 13: The figure shows the regions corresponding to semi-transparent, opaque and background points in the image. A hypothetical region R indicated by the blue rectangle is used for the illustration.

denote the point set corresponding to the semi-transparent objects in R_I. Let P_{opq} denote the set of points in R_I that have features different from the features of points in R_E and do not belong to the point set corresponding to semi-transparent object(s). These points belong to the opaque objects and texture patches that lie only in R_I . We will henceforth refer to these points as opaque points. Let P_{bg} denote the set of all points that do not belong to either the set of points corresponding to the semi-transparent object or the set of opaque points. So, this point set would include all the points lying in R_E and the points that are not a part of semi-transparent or opaque objects in R_I. Let $(P_{tr}|P_{bg})$ denote a set of all those points of semi-transparent objects with features similar to some points in P_{bg}. 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 do not have features similar to points in P_{bg}, we have $(P_{opq}|P_{bg}) = \emptyset$. Therefore, the point-set corresponding to the semi-transparent objects in R_I is given by (12).

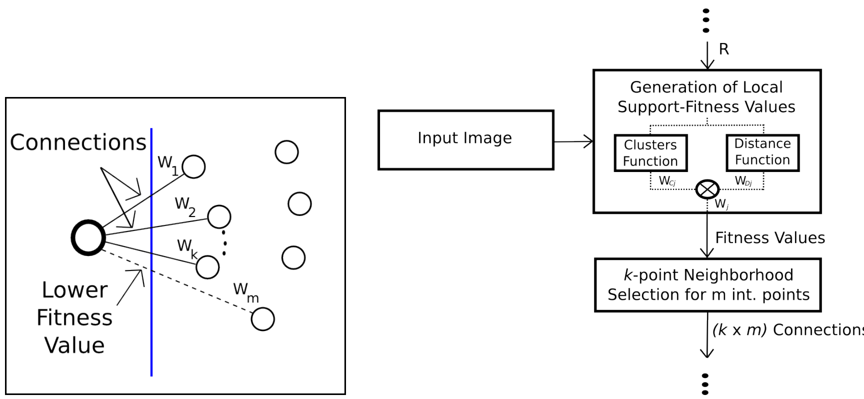
$$P_{tr} = (P_{tr}|P_{opq}) \cup (P_{tr}|P_{bg}) \tag{12}$$

Let P_{total} denote the set of all points in the image. We have,

$$P_{total} = P_{opq} \cup (P_{tr}|P_{opq}) \cup (P_{tr}|P_{bg}) \cup P_{bg} \tag{13}$$

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 two point sets $(P_{tr}|P_{bg})$ and $(P_{bg} \cup P_{opq} \cup (P_{tr}|P_{opq}))$ out of which the set $T_1 = (P_{tr}|P_{bg})$ is extracted based on the feature-reward output. The subset $(P_{opq} \cup (P_{tr}|P_{opq}))$ is further processed only if $P_{opq} \neq \emptyset$, to extract the set $T_2 = (P_{tr}|P_{opq})$ (which is carried out by Intra-Region Classification, discussed in the Section 4.5). The point sets T_1 and T_2 are reported as a final result.

4.2. Support Fitness Functions



(a) (b)

Figure 14: (a) The figure shows an illustration of connections between one interior point and few exterior points. (b) The figure shows the block diagram corresponding to the generation of support fitness values.

Recollecting what has been discussed earlier, we have connections set up between each point p_i inside the region and points p_e outside the region (denoted by black lines as shown in the Figure 14(a)). A *Support Fitness Function* is a weighting function that provides a fitness score to each of the connections depending on whether an interior point p_i is a suitable semi-transparent counterpart of an exterior point p_e. Among the existing connections between the interior and exterior points, the best k connections are selected based on their fitness values (W_i) and they form an input to the reward generation block. Given m interior points, we have a total of $k \times m$ connections as an output of the block (Figure 14(b)). We have used two support fitness functions, which are discussed in the sections below.

4.2.1. Clusters Fitness Function

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). The clustering is done using the K-means algorithm [8]. The number of clusters is computed automatically based on inter-cluster mean distances and other thresholds. Since, points belonging to semi-transparent objects are not easily discernible, they would generally either belong to a cluster of the background (or the foreground points), or a cluster that belongs to opaque points, or a cluster of points corresponding to highlights. Therefore, the points belonging to the similar clusters are emphasized and given larger weights compared to the distinct ones. We used a combination of euclidean and topology distance metric between the clusters, similar to the one discussed by Y. Peng et al. in [12], to assign weights to the individual points.

$$W_{Cj} = e^{-(\frac{c_{ind}^2}{2\sigma_c^2} + \frac{md^2}{2\sigma_d^2})} \tag{14}$$

Where, σ_c and σ_d are the standard deviations with respect to topological distance (c_{ind}) and euclidean distance (md) respectively.

). This is useful as the points of transparent objects have a definite relationship with the surrounding points. Therefore, the combination of c_{ind} and md provides separability among closely similar clusters but also provides connectivity among clusters with wide mean separation. Note that the term c_{ind} could be removed if the mean distances among all the clusters are normalized. Although, normalization removes the relative fitness-value levels between multiple support fitness functions, hence cluster index is used.

4.2.2. Distance Fitness Function

Euclidean distance forms an important fitness function. Similarity in features between a semi-transparent object and the background is higher if they are closely situated. This also avoids the variation in perspective, focus and radial distortion in the camera. We made use of two bell-shaped exponential functions (15), centered at the interior point p_i and each varying across the dimensions of the region R. The coefficients are hand-selected ($a = 0.7$ and $b = 0.3$) to give more emphasis to the smaller dimension of the region R. D is equal to the euclidean distance between the interior and the exterior point, i.e. $||p_i - p_e||$. $\sigma_1 = \frac{2}{3}min(Rwidth, Rheight)$ and $\sigma_2 = \frac{2}{3}max(Rwidth, Rheight)$ are the variance parameters, where $Rwidth$ and $Rheight$ are the dimensions of the region R.

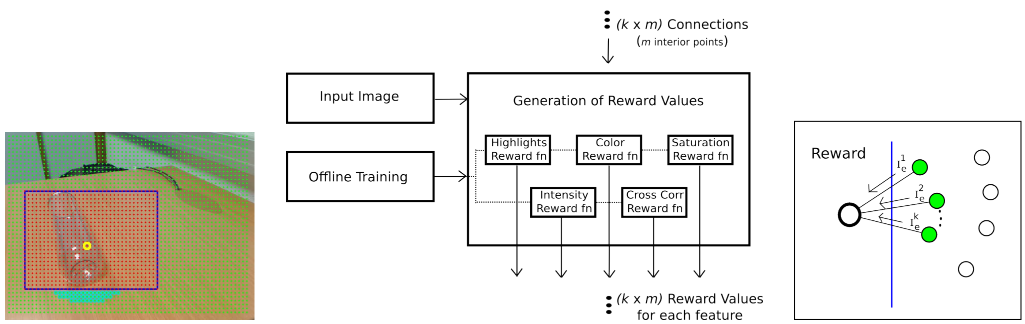
$$W_{Dj} = ae^{-\frac{D^2}{2\sigma_1^2}} + be^{-\frac{D^2}{2\sigma_2^2}} \tag{15}$$

The resultant fitness value for each connection (p_i, p_e) is equal to the product of the fitness values of the individual fitness functions (16)

$$W_j = W_{Cj} * W_{Dj} \tag{16}$$

of the region R. The coefficients are hand-selected ($a = 0.7$ and $b = 0.3$) to give more emphasis to the smaller dimension of the region R for the reasons explained above. D is equal to the euclidean distance between the interior and the exterior point, i.e. $||p_i - p_e||$. $\sigma_1 = \frac{2}{3}min(Rwidth, Rheight)$ and $\sigma_2 = \frac{2}{3}max(Rwidth, Rheight)$ are the variance parameters, where $Rwidth$ and $Rheight$ are the dimensions of the region R.

4.3. Generating Rewards



(a) (b) (c)

Figure 15: (a) The figure shows a sample image with an arbitrarily selected region R (blue rectangle). Points colored green are exterior points and colored red are interior points. An interior point highlighted by yellow color has $k = 40$ suitably-fit exterior points highlighted by cyan color. (b) The figure shows the block diagram corresponding to the generation of reward values. (c) The figure illustrates the feature rewards generated for the best k connections. Figures should be viewed in color

The best k connections for each interior point (see Figure 15(a)) (a total of $k \times m$ connections for m interior points) based on the fitness values (W_j) are input to this Reward Generation block (see Figure 15(b)). Each of these connections are tested further with the feature reward functions (see Section 3) to generate a reward which is aggregated for each p_i (Figure 15(c)). 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 to an opaque object or the background itself.

4.4. Collective Reward and Classification

Collective reward is the aggregated result of a feature-reward function acting on all the connections between an interior point and the corresponding suitably-fit k exterior points. For each feature-cue $f \in$ {Highlights, Cr, Cb, Saturation, Intensity and Cross-correlation}, a collective reward is found for every interior-point $p_i \in R_I$. Let $I^f_{i,j},\ j \in (1, ..., k)$ denote the reward generated by a feature reward function (see Section 3) of a feature f, for the connection pair $(p_i,\ p_e)$, where p_e belongs to the suitably-fit k points found via support fitness functions (see Section 4.2). From the reward functions of each feature f discussed in Section 3 and with the calculated feature distortion d as an argument, the reward for each connection given by

$$I^f_{i,j} = Rw_f(d) \tag{17}$$

Let I^f_i denote the collective reward for each point $p_i \in R_I$ and for each feature $f \in$ {Highlights, Cr, Cb, Saturation, Intensity and Cross-correlation}. It is calculated using (18)

$$I^f_i = \frac{1}{W'_1}\left(W_1 I^f_{i,1} + W_2 I^f_{i,2} + ... + W_k I^f_{i,k}\right) \tag{18}$$

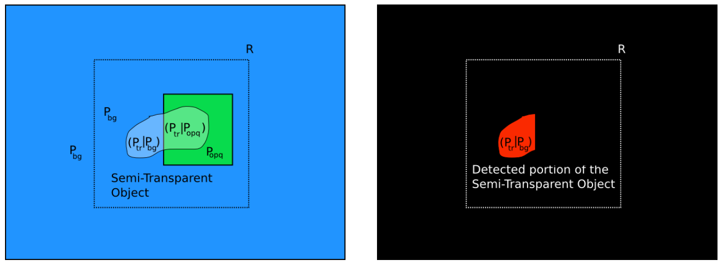
Where $\{W_1, W_2, ..., W_k\}$ are the weights denoting the fitness values (computed using (16) for each connection $(p_i \mapsto p^1_e, p_i \mapsto p^2_e, ..., p_i \mapsto p^k_e),\ \forall p_i \in R_I$. W'_1 is a normalization factor equal to $(W_1 + W_2 + ... + W_k)$.

Collective rewards (I^f_i) for each feature $f \in$ {Highlights, Cr, Cb, Saturation, Intensity and Cross-correlation} are determined. As each of the individual feature functions turn out to be weak classifiers for semi-transparent object detection, an ensemble of classifiers is formed to build a strong classifier. The total collective reward I_i for each point $p_i \in R_I$ is then found as an output to the strong classifier.

4.5. Intra-Region Classification

This section discusses about retrieving the point-set $T_2 = (P_{tr}|P_{opq})$ (refer Section 4.1 for the notations used), i.e. the points of semi-transparent objects that have features similar to the objects that are exclusively present inside the hypothetical region R. Although this step is only required if there exists such an object(s).

The discussed process till now extracts only $T_1 = (P_{tr}|P_{bg})$, but the point set $T_2 = (P_{tr}|P_{opq})$ that also belongs to semi-transparent objects are lost



(a) (b)

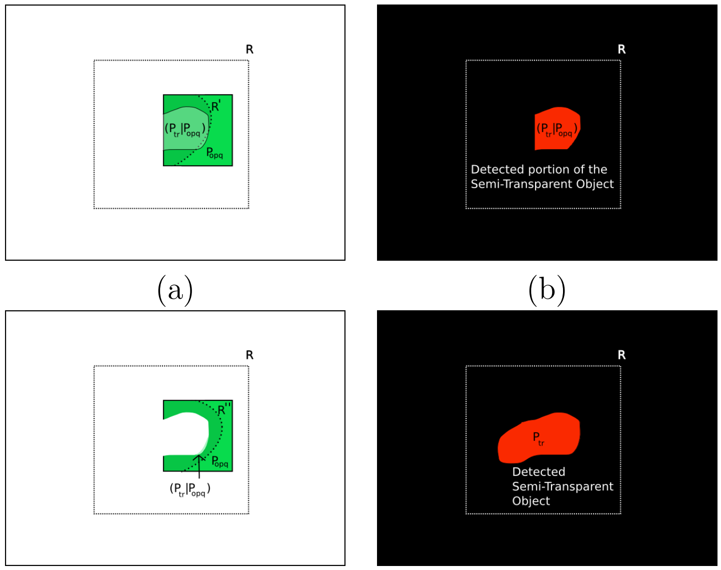
Figure 16: The figure (a) illustrates the regions relating to the semi-transparent object that have features similar to the background and the opaque patch (object). (b) The final outcome on carrying out the collective-reward based approach using the region R.

because they do not have features similar to P_{bg}. Figure 16(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 16(b) shows the result of the collective-reward based approach with the region R. Only the portion $(P_{tr}|P_{bg})$ is detected and the rest of the semi-transparent object is filtered out. In order to recover this portion we perform an additional operation called the intra-region classification. Let C_I and C_E be the set of clusters (refer Section 4.2.1) present in R_I and R_E respectively. A condition $C_I = C_E$ is checked to initiate this intra-region classification step.

We first find out the point-set that belongs exclusive in R_I (Figure 17(a)) which is given by,

$$P_{opq} \cup (P_{tr}|P_{opq}) = \{p_i | P(C_I(p_i)) \cap P(C_E(p_e)) = D, \aleph(D) < \delta,\ \forall p_i \in R_I,\ \forall p_e \in R_E\} \quad (19)$$

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 that lie mostly inside and have a few points outside the region. From the gravity center of the detected transparent points that are in close proximity, Mahalanobis distance (D_M) is determined for each point in the set $(P_{opq} \cup (P_{tr}|P_{opq}))$. The median of the distance set D_M, denoted by d_{med}, is used to partition the point-set $(P_{opq} \cup (P_{tr}|P_{opq}))$ into two sub-sets (or sub-regions) in the image. One region comprising of points that satisfy $D_M \leq d_{med}$ and the remaining points into the other region. Let R' (Figure 17(a)) be the partition which separates these two regions



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

Figure 17: (a) The 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 classification 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''.

in the region R. The region that is closer to the detected portion of semi-transparent object is termed as interior ($R_I^{'}$) of the new region R' and the rest is termed as exterior ($R_E^{'}$). The collective-reward based approach is carried out between $R_I^{'}$ and $R_E^{'}$ to extract most of the points that belong to $(P_{tr}|P_{opq})$. Figure 17(b) shows the segmented portion $(P_{tr}|P_{opq})$. To get a more improved segmentation, the intra-region classification can be performed another time to get a new region R'' which will encompass the remaining portion of the semi-transparent object (Figure 17(c)). Finally, the result is normalized and combined to give $P_{tr} = T_1 \cup T_2$ (see Section 4.1) as shown in the Figure 17(d). For a qualitative evaluation of the intra-region classification refer Figure 25.
Post Processing Functions: Additional post processing we used to improve the result further are morphological hole filling, restoration of semi-transparent edges by performing a logical conjuction (And) of the result with the edge map and re-classification of the highlights close to the detected

semi-transparent points.

4.6. Automatic Region Selection

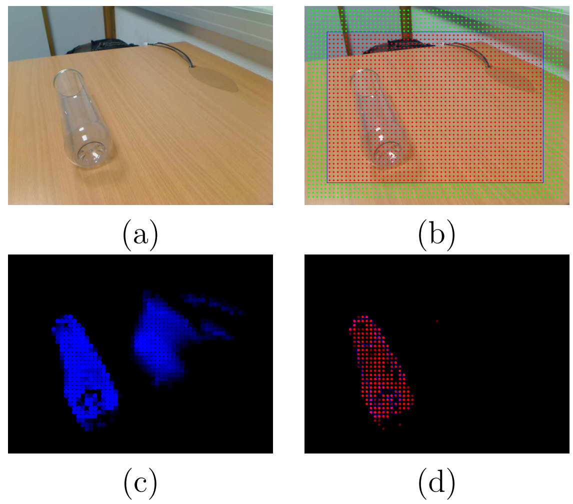


Figure 18: (a) The figure shows a sample image with a semi-transparent object. (b) A large region denoted by red points is selected for the first iteration. (c) The outcome of the first iteration with erroneous results for the points at the central part of the region. (d) The outcome of the second iteration after using the output of the first iteration as the region R itself.

We used a two step region selection method to automatically select the hypothetical region R. The algorithm is executed initially using a large region encompassing most of the image as shown in the Figures 18(a) and 18(b). The output for the points situated at the central part of the region may be erroneous as the distance between the point comparison is large (Figure 18(c)), which makes the noise due to perspective, radial distortion etc. quite comparable. In order to improve the result, we used the outcome of the first iteration as a new hypothetical 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, we get a better result as shown in the figure 18(d). Algorithm 1 presents the pseudocode of the complete automatic collective-reward based approach using 2-step region selection method.

points by executing the algorithm with the close-by points in the second iteration. Therefore, we get a better result as shown in the figure 18(d).

Algorithm 1 presents the pseudocode of the complete automatic collective-reward based approach using 2-step region selection method.

Algorithm 1 Collective-Reward Based Approach using 2-Step Region Selection Method

```
iter = 0
while iter != 2 do
    Initialize the Region R
    for each p_i ∈ R_I do
        for each feature f do
            compute fitness values ∀p_e ∈ R_e
            find the k-point Neighborhood
            compute the feature-reward values
            compute the collective-reward value
        end for
        classify p_i using ensemble of feature-classifiers
    end for
    post-process and perform intra-region classification if required
    generate the result image I
    R = Boundary(I)
    iter++
end while
report the result I
```

5. Experimental Results

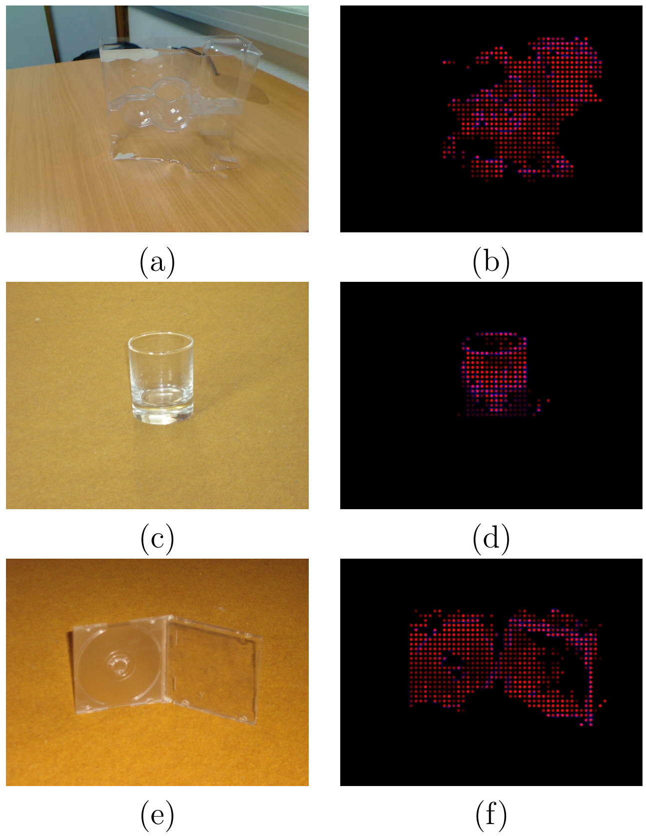


Figure 19: The figures (a)-(f) show a few semi-transparent objects and the corresponding detection results. The images should be viewed in color. For all results we used the same parameters that were learnt via offline training, and a two-step region selection method.

In this section 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 focus and radial distortion.

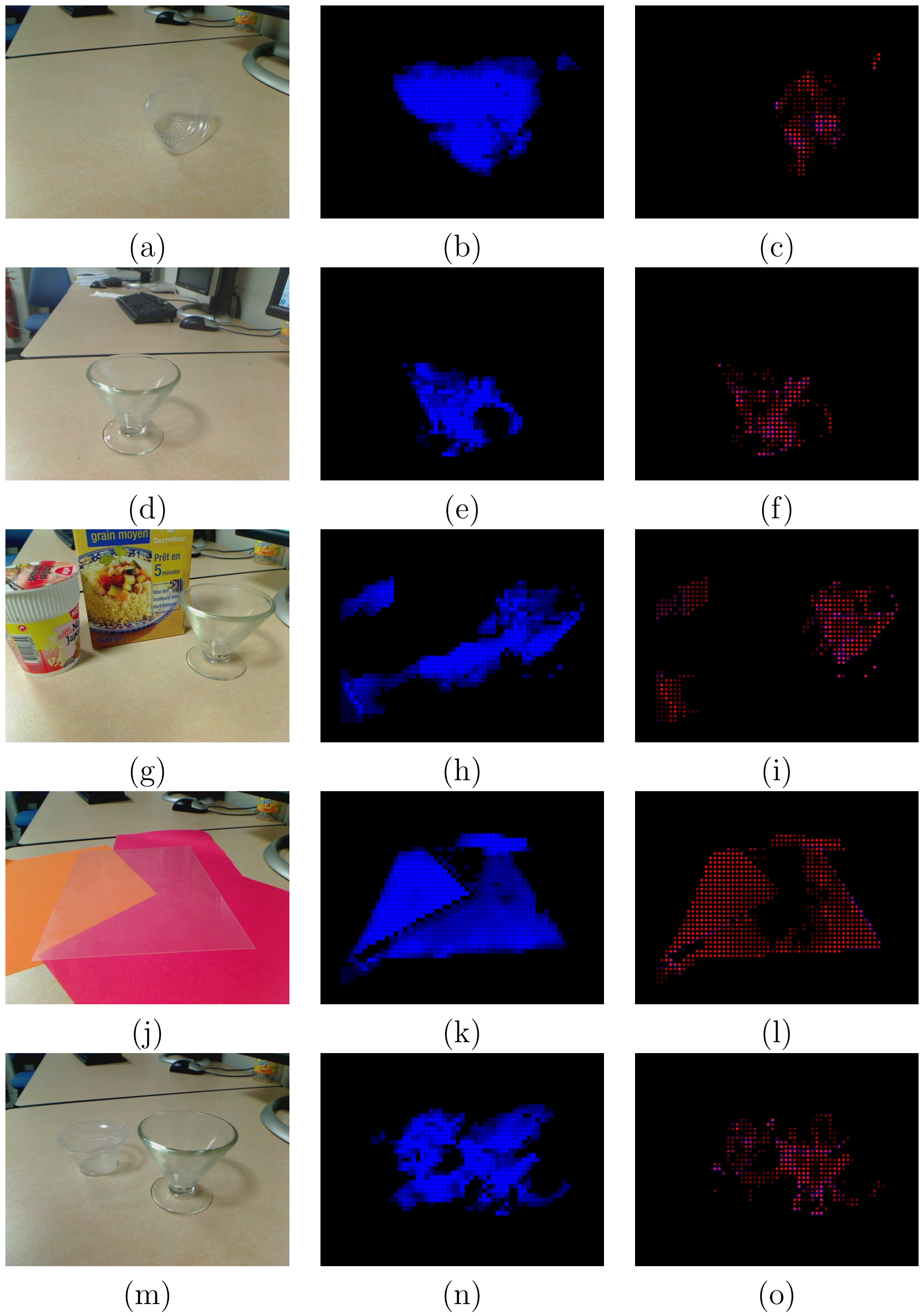


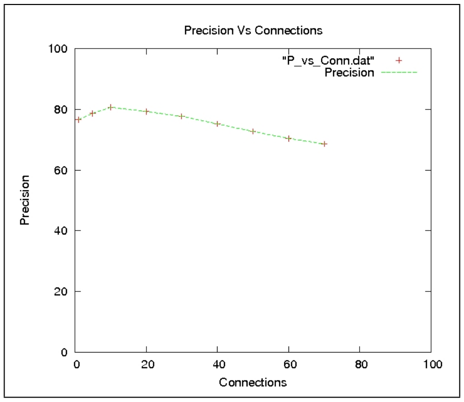
Figure 20: The figures (a)-(o) show a few semi-transparent objects and the corresponding detection results using the two-step region selection method. The Second column shows the result of the first iteration and the third column shows the result of the second iteration. The images should be viewed in color.

To make the algorithm computationally tractable, the pixels considered are sampled as 1 to 10 pixels in the image. The exterior points used for comparison are restricted to the $k = 40$ 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. To quantify the results, we carefully marked the boundaries of the semi-transparent objects in images and performed experiments by measuring the precision and the recall rate. We collected a 50 image dataset (made available in the Internet URL: `http://www.idsia.ch/~kompella/datasets/TransparentDataset.zip`) 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 first discuss here the qualitative results followed with the statistics obtained from the quantitative analysis.

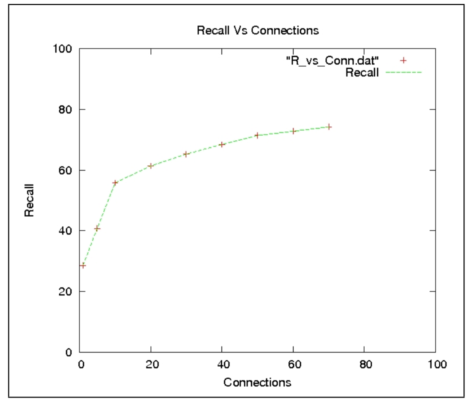
Figures 19-20 show results of a few images containing semi-transparent objects over different backgrounds. As discussed in the section 4.6, we evaluated our approach using the two-step region selection. The algorithm is executed initially using a large region encompassing most of the image. We then use the outcome of the first iteration as a region R for the second iteration. Figures 20 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 20 shows the result of the second iteration. We can clearly see that the precision of the result has improved. Figures 20(a)-(c) and 20 (d)-(f) show experiments conducted using a thin plastic cup and a refractive glass respectively. Figure 20(g) contains a semi-transparent object made of glass along with two opaque objects. The algorithm detects the glass as shown in the Figure 20(i). We can notice that there are a few false positives detected on the left. This is because the algorithm had picked up the points belonging to the dim shadows 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 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 20(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 more points of the transparent sheet get filtered due to the point-point comparisons within the transparent sheet. Figures 20(m)-(o) show the result of another experiment with two transparent objects placed next to each other.

Figure 21 shows graphs of a quantitative study made by varying the number of connections used for the algorithm. Figures 21(a) and 21(b) show the variation of *Precision* and *Recall* rates with an increase in the number of connections. We find that the precision rate slightly reduces with a number of connections larger than 10. This is due to the fact that as we increase the number of connections, many outlier points get involved in generating the rewards and therefore the noisy rewards will increase summing up to an erroneous final reward. Due to this few points outside the semi-transparent objects may get detected as transparent media although with a low probability and hence the slight reduction in the precision. This is the reason why brute force evaluation of all the points in this technique may not work well even if computational time is considered not to be a constraint. We can also find that there is a drop in the precision with a decreasing number of connections less than 10. This can be explained with the help of the recall graph (Figure 21(b)). The recall rate as expected increases with the increase in the number of connections, ie., many points inside the transparent object are now recognized as belonging to the transparent media. With a decrease in connections less than 10, we find that the recall rate becomes very low. Due to this the noise points detected outside affect the precision value and therefore the precision reduces with a decreasing number of connections. Figure 21(c) shows a graph of processing time with increasing number of connections.

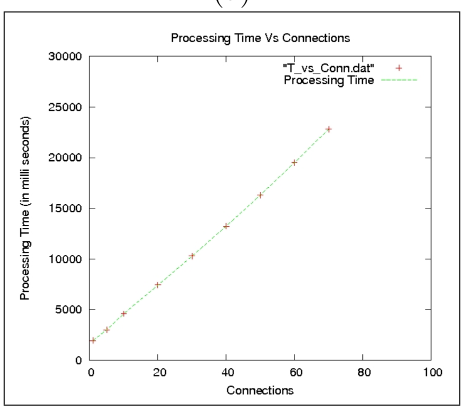
As a good combination in the precision and recall values, we achieved a precision rate of 75.73% and a recall rate of 66.21%. We consider the rates to be pretty good due to the fact that detection of the transparent objects is a fairly difficult problem to solve. Our precision, although slightly less than the precision reported by McHenry and Ponce [9], i.e. 77.03%, is for a challenging dataset. We were not able to obtain the original images used in



(a)



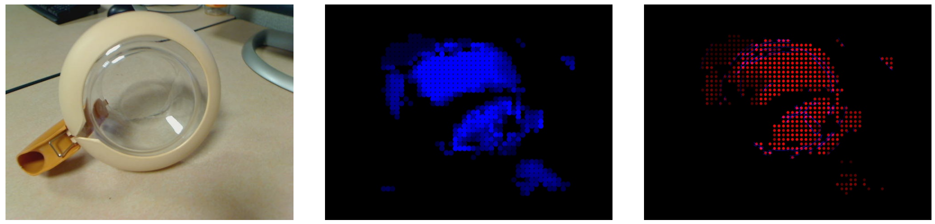
(b)



(c)

Figure 21: The figure shows graphs of (a) *Precision* Vs Number of *Connections*, (b) *Recall* Vs Number of *Connections*, and (c) $Processing{-}time$ Vs Number of *Connections*.

[9]. So, we made our own dataset (URL: `http://www.idsia.ch/~kompella/datasets/TransparentDataset.zip`) with images containing several object scenarios and some of which may fail to get detected by the algorithm presented in [9]. 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 to be slightly lower. This in a way acts as an advantage to filter out opaque parts of the transparent objects.



(a) (b) (c)

Figure 22: The 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.

Figures 22 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 23(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 23(b)) is quite noisy. On the other hand when a region R as shown in the Figure 23(c) is used, we get a much better output (Figure 23(d)). We will look into the aspect of improving the region selection as a future improvement to the algorithm.

Figure 24(a) shows an image with several types of objects. We can find in Figure 24(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 24(c)). Figure 25 shows results of a few more experiments conducted on semi-transparent objects. Figure 25(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 25(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

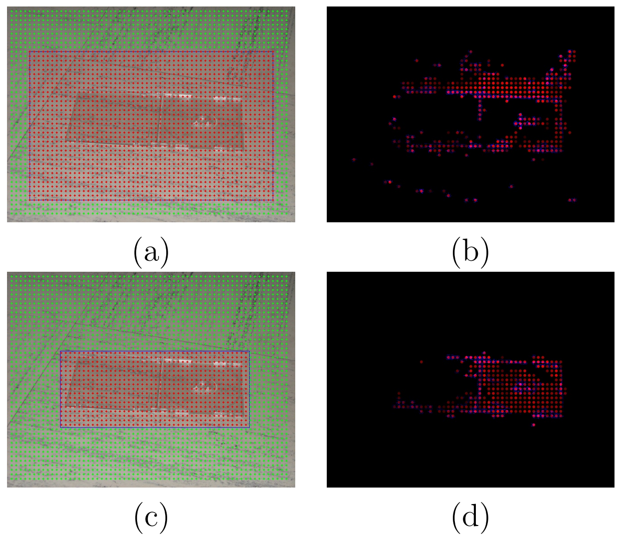


Figure 23: The figure (a) shows a sample image with a larger region R used for the detection. (b) The corresponding result of the algorithm. (c) The sample image with a smaller region R used for detection. (d) The corresponding result of the algorithm.

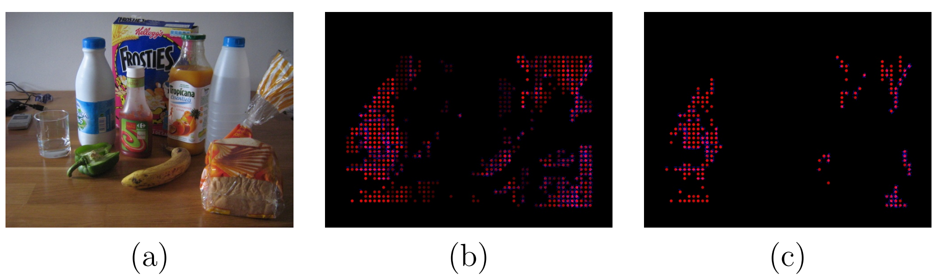
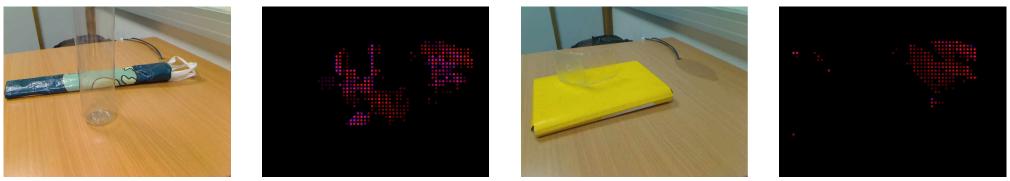


Figure 24: The figure (a) shows a sample image with several objects. (b) The result of the algorithm with a lower threshold. (C) The result of the algorithm with a higher threshold.

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 26(a)-(b) show another example of a false detection where the patch as discussed earlier has been falsely detected with a higher probability due to

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

Figure 25: Figures (a)-(d) show a few semi-transparent objects and the corresponding detection results. The images should be viewed in color.



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

Figure 26: The figures (a)-(b) show an example of a false detection by the algorithm. The figures (c)-(d) show an example of a miss-detection of the semi-transparent object present in the image.

the presence of a white plastic close to it which got detected as highlights in the image. Figures 26(c)-(d) show an example of a complete miss-detection of the semi-transparent object in the image.

6. Conclusions and Future Work

We proposed an 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 automated. A two-step region selection method was discussed in this regard. An improvement in the 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 section 4). A better result may be achieved by combining the information regarding the structure of the detected points.

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.

References

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

[2] 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.*, 75(1-2):46–58, 1999.

[3] M. Ben-Ezra and S. K. Nayar. What does motion reveal about transparency? *Computer Vision, IEEE International Conference on*, 2:1025, 2003.

[4] A. Ford and A. Roberts. Colour space conversions, 1998.

[5] S. Hata, Y. Saitoh, S. Kumamura, and K. Kaida. Shape extraction of transparent object using genetic algorithm. *Pattern Recognition, International Conference on*, 4:684, 1996.

[6] G. J. Klinker, S. A. Shafer, and T. Kanade. A physical approach to color image understanding. *Int. J. Comput. Vision*, 4(1):7–38, 1990.

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

[8] D. MacKay. Chapter 20. an example inference task: Clustering, 2003.

[9] K. McHenry and J. Ponce. A geodesic active contour framework for finding glass. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 1:1038–1044, 2006.

[10] K. McHenry, J. Ponce, and D. Forsyth. Finding glass. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 2:973–979, 2005.

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

[12] Y. Peng, Q. Guo, I-F. Shen, and W. Chen. Topology distance for manifold clustering. 7:V7–197 –V7–201, april 2010.

[13] Y. Y. Schechner, N. Kiryati, and R. Basri. Separation of transparent layers using focus. In *Proc. ICCV*, pages 1061–1066, 1998.

[14] M. Singh and B. L. Anderson. Toward a perceptual theory of transparency. *Psychological Review*, 109:492–519, 2002.

[15] M. Singh and B. L. Anderson. Photometric determinants of perceived transparency. *Vision Research*, 46:879–894, 2006.

[16] M. Singh and X. Huang. Computing layered surface representations: An algorithm for detecting and separating transparent overlays. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 2:11, 2003.

[17] Y Wexler, A. W. Fitzgibbon, and A 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.