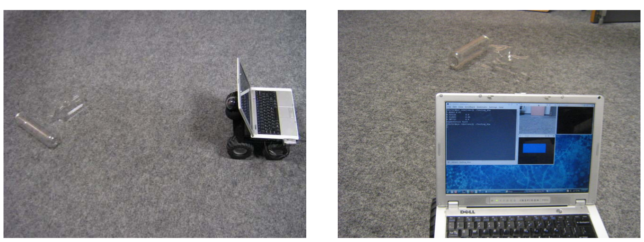
2011 IEEE International Conference on Robotics and Automation
Shanghai International Conference Center
May 9-13, 2011, Shanghai, China

Detection and Avoidance of Semi-Transparent Obstacles using a Collective-Reward Based Approach

Varun Raj Kompella and Peter Sturm

***Abstract*— Most of the computer and robot-vision algorithms are designed mainly for opaque objects and non-opaque objects have received less attention, in spite of them being omnipresent in man-made environments. With an increasing usage of such objects, especially those made of glass, plastic etc., it becomes necessarily important to detect this class of objects while building a robot navigation system. Obstacle avoidance forms a primary yet challenging task in mobile robot navigation. The main objective of this paper is to present an algorithm to detect and avoid obstacles that are made of semi-transparent materials, such as plastic or glass. The algorithm makes use of a technique called the collective-reward based approach to detect such objects from single images captured by an uncalibrated camera in a live video stream. Random selection techniques are incorporated in the method to make the algorithm run in real-time. A mobile robot then uses the information after detection to perform an obstacle avoidance maneuver. Experiments were conducted on a real robot to test the efficacy of the algorithm.**

I. Introduction and Related Work



(a) (b)

Fig. 1. (a) A navigating robot with semi-transparent obstacles in its path. (b) Another view of the robot with a sample output shown on its desktop screen.

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 [1]. Perceptual transparency is the phenomenon of seeing one surface behind another. Adelson and Anandan [2] 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.

Transparency and its related problems have received relatively less attention in the computer vision research. Singh and Huang [3] discussed about the separation of transparent overlays from the background surfaces by making use of polarities of X junctions along the boundaries of objects. Schechner *et al.* [4] have used the concept of depth from focus along with reconstruction to separate such overlays. McHenry and Forsyth [5] used the edge information determined by a Canny edge detector to capture cues relating to transparent objects across their boundaries. This method was later extended by McHenry and Ponce [6] with a region-based approach along with the edge information to classify regions as transparent or not. 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. Also, 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.

Lately, with an increasing usage of objects made of glass in man-made environments, a mobile robot navigating in an office would need to avoid colliding with such obstacles on its path. Figure 1(a) 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. The algorithm presented in this paper would enable a mobile robot to perform successful collision avoidance against such class of obstacles. Figure 1(b) shows a sample output of the detection process. Also, the technique can be further extended to carry out several other strategies such as self-localization of a mobile robot in the presence of glass doors etc. 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. We used random selection techniques to reduce the computational time.

II. 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 III-A.

Highlights and Caustics: Transparent objects are usually highly specular and refractive, therefore the presence of highlights and caustics increases the probability of a possible transparent material around. These highlights are found in an image using the method discussed in [9].

Color: Semi-transparent objects like glass, plastic, etc. generally have impurities and also due to the presence of specu-

V.R. Kompella is with IDSIA, Lugano, Switzerland varun@idsia.ch
P. Sturm is with INRIA Rhône-Alpes, Grenoble, France peter.sturm@inrialpes.fr

lar reflections, the background color is slightly distorted. Cr and Cb components of the $YCrCb$ color model [10] is used in this regard.

Saturation: 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 [11]. Therefore these pixels have relatively lower $saturation$ values.

Intensity: Intensity plays a major role for backgrounds with texture. Michelson's contrast constraint is used as it has been shown in [1] that transparency lowers its value.

Cross-Correlation Measure: Distortion produced by a semi-transparent object can also be captured by a region analysis. The normalized cross-correlation score is calculated at each point and the values of maximum and minimum are used to determine the presence of semi-transparent objects.

III. Collective-Reward Based Approach

This section discusses details of the collective-reward based approach for the detection of semi-transparent objects in an image. A native version of the algorithm, presented in [7], performs a reasonably accurate segmentation of semi-transparent objects in the order of 10-30 secs on a Pentium IV machine. The method presented in this paper is a significant modification, to make the algorithm work in real-time in order to suit to a robotic application.

Images of semi-transparent objects typically contain the distorted features of what lies behind the objects [8]. Although 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. As in a single image there is no access to the actual features of the regions behind the semi-transparent object, the surrounding pixel information is used instead. For a given input image, the boundary corresponding to the semi-transparent object is not known. Therefore, an arbitrarily selected rectangular hypothetical region R is used to calculate the feature distortion values for pairs of points in the image. The selection of R is automated to perform the obstacle detection and avoidance autonomously (refer Section IV). For each point interior in R, feature-distortion is calculated from the pixels exterior to R. These distortion values are then aggregated by using the collective-reward based technique to classify whether the interior point belongs to a semi-transparent object or not.

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. The following sub-sections discuss details of several elements of the algorithm starting from the quantification of the feature cues, weighting functions and finally the collective-reward generation and classification.

A. Feature Reward Functions

The feature cues discussed in Section II are quantified and the distortion is calculated by using a set of feature-reward functions. *Feature reward functions* are probability density functions of the semi-transparent points for a given feature distortion value. 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 on the difference between the semi-transparent object vs background over background vs background or opaque vs background regions, where opaque regions stand for the objects different from background. Feature-reward functions for the feature cues Cr, Cb and $saturation$ are generated by an offline training while for the rest, a handset-model is used.

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

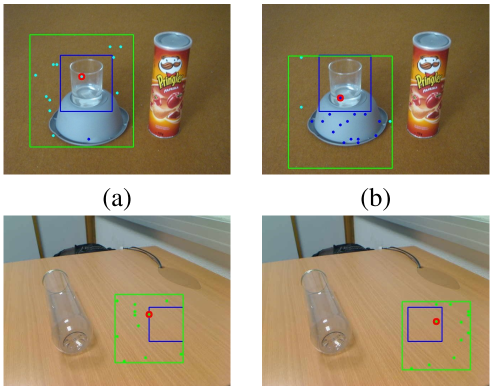
$$Rw(d) = \left(\frac{n_d^{tr}}{n_d^{tr} + n_d^{bg}} \right), \ d \in (0, G) \quad (1)$$

where, n_d^{tr} and n_d^{bg} are the number of points belonging to the semi-transparent object (P_T) and background (P_B) 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. The quantities n_d^{tr} and n_d^{bg} in (1) are found from the histograms of feature-difference values between a set of points belonging to the $P_T \cup P_B$ and a set that contains only points from a similar background. The reward functions of features Cr, Cb and $saturation$ are computed in this manner.

Handset Model: A gaussian function is used for $highlights$, with euclidean distance between a point and the closest highlight-point as an argument. A Threshold for Michelson's contrast and normalized cross-correlation values are used for $intensity$ and $cross-correlation$ feature cues.

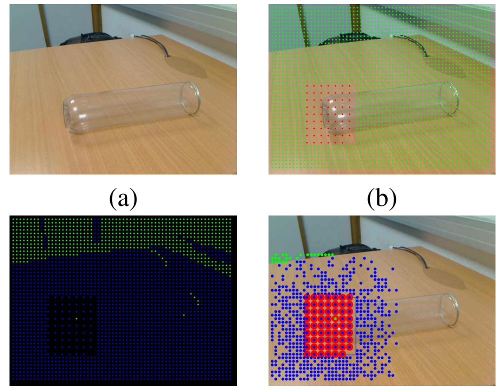
B. Support Fitness Functions

For each point p_i interior to the region R, rewards are computed using the points p_e exterior to R. Usage of all the exterior points for this purpose was found to be not fruitful because only few exterior points, which are similar to the actual inaccessible point behind the transparent surface, are useful in characterizing whether the corresponding interior point belongs to a semi-transparent object or not. Therfore, an aggregation over all the points could lead to an erroneous result. Besides, computing over all the points is computationally expensive. Therfore, a set of weighting functions called the *Support Fitness Functions* are used to



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

Fig. 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.



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

Fig. 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.

find a limited k suitably-fit exterior points for each interior point for collective reward generation and classification.

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. The term "connection" is used to denote an association made between a pair of points (p_i, p_e). Two fitness functions called the Clusters Fitness Function and Distance Fitness Function are used. *Clusters fitness function* gives higher fitness values to the exterior points that fall into the color clusters that are close in terms of centers of gravity (mean-distance (md)) and cluster rank (c_{ind}) with respect to the cluster of the interior point. Clustering of points is carried out in $Cr - Cb$ color space. A gaussian model (2) is used with mean-distance and sorted (Bubble Sort) cluster rank as arguments. Cluster rank is used in order to add separation to closely connected clusters and connect distant clusters that have consecutive ranks. The Clusters fitness function is given by:

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

Where, σ_c and σ_d are the standard deviations with respect to cluster-index (c_{ind}) and absolute mean distance (md) respectively.

Distance fitness function gives more emphasis to the fitness values of the exterior points that are close in terms of euclidean distance to the interior point.

Random Selection using Clusters Fitness Function: Evaluating the fitness values for every connection pair in order to find the best k connections turned out to be time consuming. In order to improve the computational speed, we resorted for the k-points selection based on random distribution modeled using the clusters fitness function. 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 then 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 less for slightly different clusters and even lesser with the points of largely different clusters.

Rejection Sampling using Distance Fitness Function: 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 discussed above, the correlations are much better for the points that are close. Therefore, we made use of rejection sampling based on the distance fitness function 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 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 color cluster similar to that of the floor. Therefore we find that the majority of the exterior points selected are of the same color cluster and only few points are selected from the cluster belonging to the opaque object which is indicated by blue color. Figure 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 2(c) and 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 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 2(d) the mask is at a different location collecting points on the right side of the region R.

To make it computationally tractable the point nodes considered in the algorithm are further sampled. As the main objective of the robot is, to detect and approximately locate the object in the image, we may relax on finding 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 as 1 node to 10 pixels. It is important to have a higher sampling rate outside than inside. Figure 3(a) shows a sample image with a semi-transparent object. Figure 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 3(c). Figure 3(d) shows the random selection of the points outside the region. It can see that the points are concentrated within a rectangular region by rejection sampling.

C. 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 point p_i interior to R. Let $I^f_{i,j}$, $j \in (1, ..., k)$ denote the reward generated by a feature reward function (see Section III-A) 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 III-B). From the reward functions of each feature f discussed in Section III-A and with the calculated feature distortion d as an argument, the reward for each connection given by

$$I^f_{i,j} = Rw(d) \tag{3}$$

Let I^f_i denote the collective reward for each point p_i interior to R and for each feature $f \in$ {Highlights, Cr, Cb, Saturation, Intensity and Cross-correlation}. It is calculated using (4)

$$I^f_i = \frac{1}{W_1'}\left(W_1 I^f_{i,1} + W_2 I^f_{i,2} + ... + W_k I^f_{i,k}\right) \tag{4}$$

Where $\{W_1, W_2, ..., W_k\}$ are the weights denoting the fitness value (computed as explained in Section III-B) of 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. However, we found that each of the individual feature functions turn out to be weak classifiers for semi-transparent object detection and therefore an ensemble of classifiers is formed to generate 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.

IV. Region Selection and Obstacle Avoidance

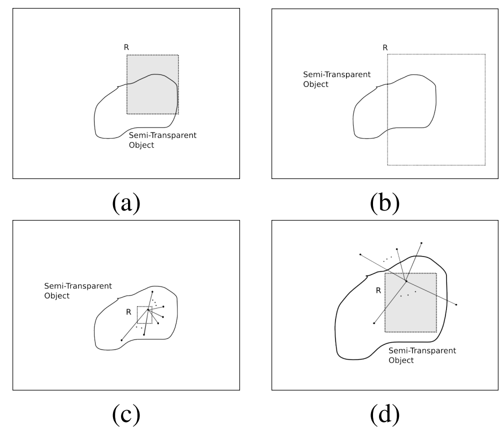


Fig. 4. (a) Figure illustrates an image containing a semi-transparent object with an optimal sized region R. The region is shaded because the detection ratio crossed the threshold. (b) Figure illustrates the case when R is too large with a possible miss-detection and also leads to a large object localization 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.

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 in the image, we note down all those regions where the ratio of the detected transparent points to the total number of points in the region, termed as *Detection-Ratio*, is greater than 0.25. Figure 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 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. Figure 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 4(c).

As the robot moves towards an object, the object's projected image size increases, therefore a region which is not too small or not too large is selected for a successfull 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 for the detection of inner regions of semi-transparent objects using a region as shown in the Figure 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 (See Algorithm 1). We make use of 12 regions spanned accross the image for the obstacle avoidance. The velocity of the robot reduces if the number of regions highlighted in the top row (N_{topR}) is greater than 0, while the bottom row is responsible for the turn maneuver. Based on the number of regions highlighted left ($N^{left}_{bottomR}$) or right ($N^{right}_{bottomR}$), a respective turn maneuver is generated.

Algorithm 1 Semi-transparent obstacle avoidance using collective-reward based approach

for each region R **do**
  for each $p_i \in R$ **do**
    for each feature f **do**
      find the k-point neighborhood via random selection
      compute fitness values $\forall k$ points
      compute the feature-reward values
      compute the collective-reward value
    end for
    classify p_i using ensemble of feature-classifiers
  end for
  Highlight R if Detection-Ratio > 0.25
end for
if $N_{topR} > 0$ **then**
  reduce robot speed
end if
if $N^{left}_{bottomR} \geq N^{right}_{bottomR}$ **then**
  turn right
else
  turn left
end if

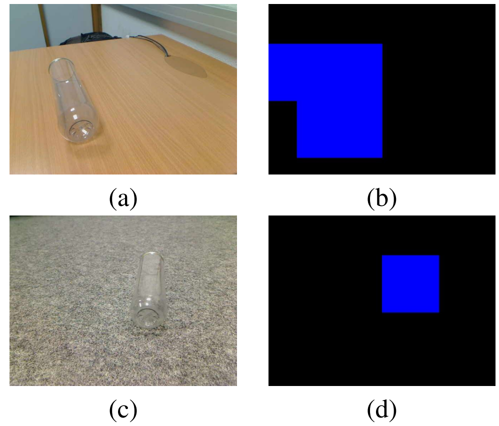


Fig. 5. Figures (a)-(d) show a few semi-transparent objects and the corresponding detection results. The location of the object in the image is found as the highlighted rectangular regions in which the detection ratio has crossed the threshold.

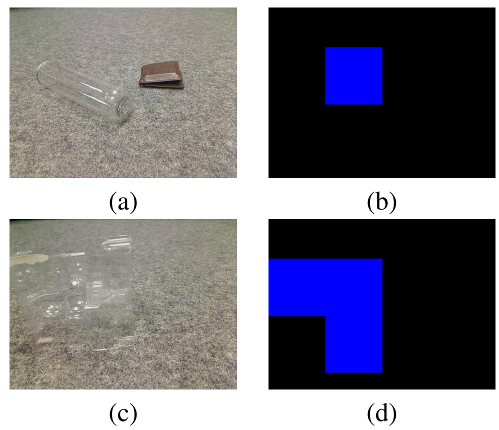


Fig. 6. Figures (a)-(d) show a few semi-transparent objects and the corresponding detection results. Figure (b) shows that the algorithm only detects semi-transparent objects.

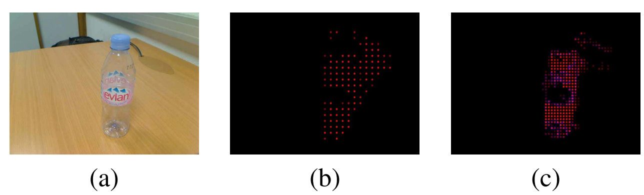
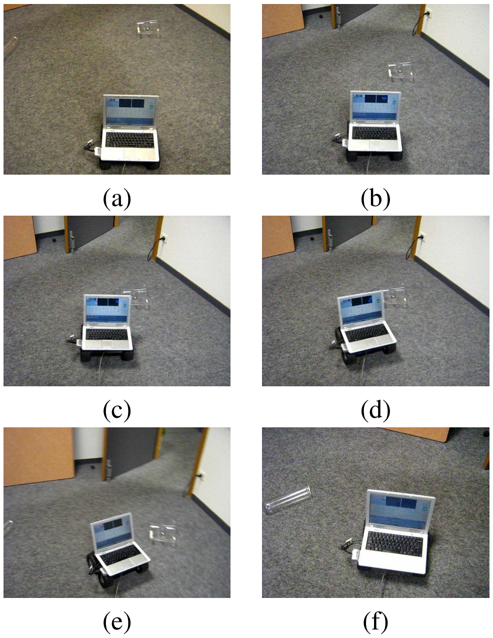


Fig. 7. Figures (a)-(c) show a comparison between the outputs of both the native approach and the modified approach of the algorithm. (a) Sample input image (b) Result of the modified approach (c) Result of the native approach

V. Experimental Results

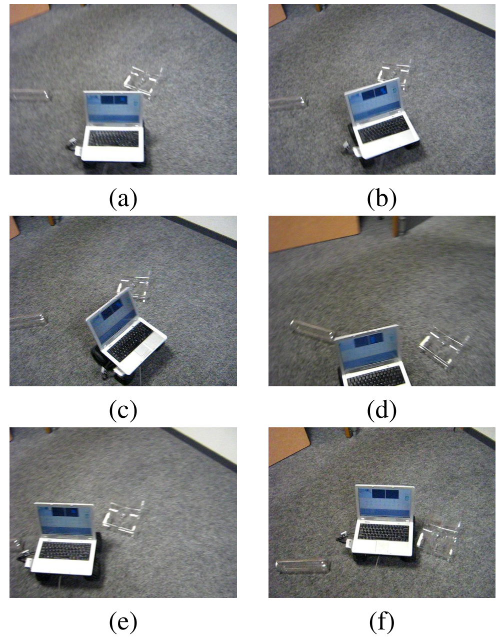
In this section, we will present the experimental results conducted using images captured from a web camera on a mobile robotic platform. Figures 5 show the algorithmic response for the corresponding sample image. We can see in the Figure 5(b) that the semi-transparent object is detected and localized 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(b) and (d). Figure 6(a) shows another input image with a semi-transparent object along with an opaque object. The result (Figure 6(b)) shows that only the semi-transparent object is detected. Figures 6(c)-(d) show another example-pair with a different object in the scene.

Figure 7 shows the outputs of both the native approach presented in [7] and the one presented in this paper compared with each other. It has been reported in [7] that the native approach performed pretty good (given the difficulty of the task) with a precision rate of 77.19% and a recall rate of



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

Fig. 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)

Fig. 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.

65.84% over a dataset of 50 images. The native approach is more accurate, as expected, than the modified approach. On the contrary the modified approach is much faster. The execution time on a Pentium IV 1GHz (single core) machine 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. The reward-generation block takes about 1 sec. The region used for this comparison was one that encompasses most part of the image. Although for the obstacle avoidance we use smaller regions to make the avoidance decision faster relatively. We also conducted experiments on a real robot. 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 6(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 8 and 9 show snapshots taken from a video of the robot while performing the obstacle avoidance with semi-transparent obstacles on its path.

VI. Conclusions and Future Work

We proposed an approach to detect the presence of transparent obstacles and perform obstacle avoidance using the collective-reward based approach. This approach makes use of the dependency between the points belonging to the transparent object and the points that are situated around. Using random selection techniques the algorithm works in quasi real-time on a pentium IV machine. We look forward to carry out several other important applications such as localization and mapping of the robot looking through a glass door. 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] M. Singh and B. L. Anderson, "Toward a perceptual theory of transparency", *Psychological Review*, vol. 109, 2002, pp. 492-519.

[2] E. H. Adelson and P. Anandan, "Ordinal characteristics of transparency", *in Proc. AAAI workshop on Qualitative Vision*, 1990, pp. 77-81.

[3] 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*, Los Alamitos, CA, USA, 2003, pp. 11.

[4] Y. Y. Schechner and N. Kiryati and R. Basri, "Separation of transparent layers using focus", *Proc. ICCV*, 1998, pp. 1061-1066.

[5] K. McHenry and J. Ponce and D. Forsyth, "Finding Glass", *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, Los Alamitos, CA, USA, 2005, pp. 973-979.

[6] K. McHenry and J. Ponce, "A Geodesic Active Contour Framework for Finding Glass", *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, Los Alamitos, CA, USA, 2006, pp. 1038-1044.

[7] V.R. Kompella, "Image-based Detection of Semi-Transparent Objects", *Master Thesis*, INRIA-Rhône-Alpes, 2009.

[8] H. Murase, "Surface shape reconstruction of an undulating transparent object", *Radiometry*, Jones and Bartlett Publishers, Inc., 1992, pp. 213-217.

[9] D. Androutsos and 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, 1999, pp. 46-58.

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

[11] R. Liu and Z. Li and J. Jia, "Image partial blur detection and classification", *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 2008, pp. 1-8.