Guess from Far, Recognize when Near: Searching the Floor for Small Objects

M Siva Karthik[*]
Robotics Research Centre
IIIT-Hyderabad

Sudhanshu Mittal[†]
Robotics Research Centre
IIIT-Hyderabad

K Madhava Krishna[‡]
Robotics Research Centre
IIIT-Hyderabad

ABSTRACT

In indoor environments, there would be several small objects lying around on the floor. In this work, we develop an efficient strategy to search for a set of queried objects amongst a large number of small objects lying around. Small objects of the order of $1cm - 5cm$, appear very small, making it difficult for the present algorithms to recognize them from far away. A human like strategy in such cases is to infer each object's similarity to the queried objects, from far away. Subsequently, the objects of interest are approached and analyzed from a closer proximity through an optimal plan. We develop an optimal plan for the robot, to strategically visit a selected few among all the objects. From far away, we assign Existential Probabilities to the objects, indicating their similarity to queried objects. A Bayes' Net is constructed over the probabilities, to overlay and orient a Viewpoint Object Potential(VOP) map over potential search objects. VOP quantifies the probability of accurately recognizing an object through its RGB-D Point Cloud at various viewpoints. The belief from the Bayes' Net and the discriminative viewpoints from the VOP are utilized to formulate a Decision Tree which helps in building an optimal control plan. Hence, the robot reaches strategic viewpoints around potential objects, to recognize them through their RGB-D point clouds. The framework is experimentally evaluated using Kinect mounted on a Turtlebot using ROS platform.

Keywords

Visual Object Search, Mobile Robotics

1. INTRODUCTION

In an indoor setting, a robot has to search for a set of objects in large unknown environments, where many objects lay scattered on floor. We try to address a similar case, where the environment spans over $10m \times 10m$ and objects as small as 1-5cm lie on the floor. Using early probabilistic inferences based on sparse images and object viewpoint selection for robust object recognition, a set of optimal control actions is planned. In this work, we accomplish fast active visual object search, by maximizing the reward earned by a robot, through a Decision Tree [17] formulation.

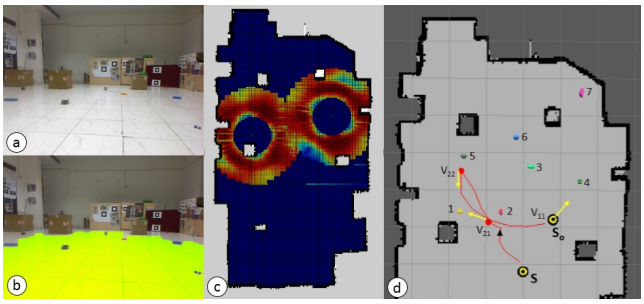

Figure 1: (a)Typical scene. (b)Floor segmentation for object detection. (c)VOP map integration for viewpoint planning. (d)Path traversed by robot during object search.

Present object search paradigms [3] cater to the cases where the objects are close to the camera, large in size and are generally lying on tables in an environment, which is small. Since a dominant part of the image captures the object, both 2D and 3D quality features points can be extracted. Also, when the objects are all close by, the robot need not travel large distances to verify all objects. On the contrary, for a mobile robot working in a moderately large indoor environment, the robot-object distance can vary significantly, which is often too large to infer reliable information using traditional approaches. In this paper, we tackle the problem of searching for small objects (of size 1cm-5cm) in large environments (of size $10m \times 10m$).

Methods applying Spatial topological relations [4] prove their efficacy very well in large spaces with distinct small partitions. While searching in a large room, lack of an accurate prior estimate about the object from far away would lead the robot to approach each object in pursuit of queried object. In a large room, such strategies would prove to be fairly time-consuming.

We try to build a novel framework inspired by human perception in such scenarios. When a particular object is to be searched in a large room, where many small objects are

[*]sivakarthik.m@research.iiit.ac.in, Corresponding author
[†]sudhanshu0301@gmail.com
[‡]mkrishna@iiit.ac.in

ICVGIP '14, December 14-18 2014, Bangalore, India

present, an initial prior estimate of each object's similarity to the object of interest is ascertained from far away. Of the many objects present in the large scene, we move only towards those objects that at most resemble our interest objects while neglecting others. Finally, we do a check from the best viewpoint before any manipulation. We adopt a similar sequence of actions to search for a set of queried objects $\mathcal{O}_f$.

Our system has four modules. Firstly, for a given scene we detect and segment out the objects that are lying on the floor(Fig 1(b)). The small objects are obtained as apart from the ground, as the result of an MRF energy minimization, over the residual of the homography error. Secondly, we find the existential probability of a detected object f being a particular object o_q from far away, from where, images with resolution of around 20×20 pixels are available. This is done by training Gaussian Mixture Models(GMM) for each of the objects(section 3.4). If the existential probability of the object f being the object o_q poses a strong belief, its best viewpoints are estimated accordingly based on its profile which is encoded as a VOP map(Fig 1). This constitutes the third phase(section 3.5). Further, we use a Decision Tree(DT) formulation (section 3.7), to choose the sequence of optimal control actions that ends with object recognition. The utility function being maximized tradeoffs between robot trajectory length and object recognition accuracy for different viewpoints.

The main contribution of this effort is a formulation that captures human intuition while searching for small objects. In other words the ability to make a prior guess about the object from far and then move towards it, make further observations on it and confirm its presence or absence. The ability to guess about an object's existence is achieved through the GMM described above as it provides for the initial guess through the object's existential probability. Secondly, the paper also brings to light that in a robotic setting the object can be seen from various viewpoints and the recognition probability varies with viewpoint and distance. To this effect a new data structure called Viewpoint Object Potentials (VOP) introduced by the authors in the supplementary material attached with this submission [20]. The VOP essentially summarizes the detection probability of an object at various angles and distances around it. For example a shampoo bottle is best recognized from its frontal view than from its side view due to sparsity of features and keypoints. Thirdly, the proposed approach combines early guesses with final best viewpoint recognition locations into an optimal decision making framework through Decision Trees (DT). DT scales to situations where more than one object needs to be searched and the scene is littered with multiple objects. Most crucially and in order to capture the full ramifications of the problem, DT computes the optimal path by also incorporating failure probabilities of its control actions along its edges. Without which the resulting path is likely to be suboptimal.

2. RELATED WORKS

The problem of object search has been studied in the past, in various related contexts like environment summarization, object oriented exploration, spatial semantic modelling, etc. In 1976, Garvey [5] proposed an indirect object search method showcasing the need to limit the search space. Subsequently, Bajcsy [8] introduced the term active perception which encourages sensor and viewpoint planning to achieve higher levels of accuracy in object recognition [18]. In the recent past, works like [2], [7] argue about strong correlation between 3D structure of the surrounding environment and object placement, showing that organization is highly expressible in terms of spatial topological relations. [6] provides a solution for search and localization of objects using a monocular camera with zooming capabilities to overcome the limitations of low resolution images of distant small objects.

[4] gives a strategy based on the probabilistic model, POMDP, making use of uncertain semantics between the object and its location, for prioritizing the search effort to promising locations in a partially known environment. In it, a probabilistic semantic mapping framework is proposed, defining joint distribution between each object category and room category. Hence, at a higher level of abstraction, it would be able to discover a plausible location of the object O_q . In our work, we try to bridge the voids encountered in scenarios where semantic relations start to weaken. For instance, when robot enters a particular room searching for objects like marker pen or water bottle, it may find such small objects anywhere impartial to any location. Since such objects do not possess any semantic relationships with the environment or among themselves, they have to be searched explicitly all over the place.

In line with this paradigm, in our work, we develop a system to counter such unaddressed subjects. We explore how early inferences about small, far away objects can be effectively used for efficient control action planning. We maximize the object recognition accuracy by identifying the discriminative viewpoints. On top of this we develop an overall framework that integrates the above strategies in a robust manner through a DT.

3. PROPOSED APPROACH

3.1 System Overview

The motivation behind this object search system is not only to reduce navigation but also to boost the robustness of object recognition module simultaneously. The flow diagram(Fig 2) presents the main idea of our framework for the object search problem. We now cast our search problem, more formally.

Let $\mathcal{O} = \{o_i\}_1^N$ be the set of N objects that exist in the environment. A robot is given the task of searching for a set of queried objects $O_s \subseteq \mathcal{O}$. We would want to find all the objects in O_s in minimum time with maximum accuracy. This can be achieved with a fast efficient plan, by visiting only those objects in the scene, which we believe are similar to objects in O_s.

There are four modules which contribute to achieve the goal of fast object search. The object detection and localization module(section 3.3) is responsible for detecting and segmenting the objects on the floor. Let the set of these segmented objects be called $\mathcal{F}$. A belief of each object in $\mathcal{F}$ being similar to an object in O_s is assigned a probability. This is done using a set of GMMs $\mathcal{G}$ learned over feature vectors(section 3.4) generated for each of the objects in $\mathcal{O}$. There would be some objects $\mathcal{C}$, which appear to be similar to queried objects O_s. Hence $\mathcal{C} \subseteq \mathcal{F}$. To this end, we guide the robot to strategic viewpoints around $\mathcal{C}$ through a set of planned controls(section 3.7). This is followed by recogniz-

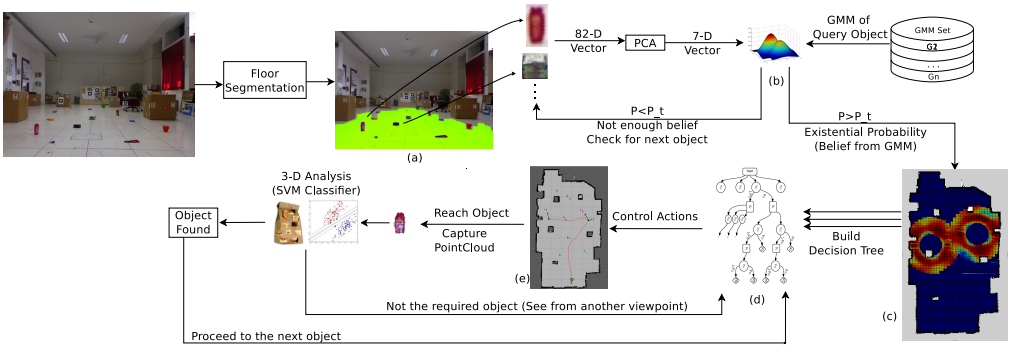

Figure 2: Object search system overview.

ing the objects using a RGB-D data based BOW [12] model from the best possible viewpoint, which is derived from an object recognition profile(section 3.5).

3.2 System Pipeline

Below, we explain in detail our system pipeline,

1. Initially, objects as far as $5m$, which are found lying on the floor, are segmented out (Fig 2(a)) using the algorithm proposed in [1]. The set of objects found is denoted by $\mathcal{F}$. A detailed description is given in section 3.3.

2. For each object in $\mathcal{F}$, if the pointcloud size $Cs_i \geq Cs_t$(threshold) (Fig 2) recognition is directly performed over the RGB-D point cloud data using an SVM [14] classifier over Visual BOW models.

3. If $Cs_i < Cs_t$, no clear RGB-D information is available. From far away, for each of the objects in $\mathcal{F}$, we assign a probability(P_{fi}), of f_i being similar to an object o_q in O_s using a GMM module detailed in section 3.4(Fig 2(b)). The probabilities help the robot rule out some of the objects in $\mathcal{F}$. $\mathcal{C} \subseteq \mathcal{F}$ is the set of objects that need closer inspection.

4. For an object c_i similar to an object o_q in O_s, we build a Viewpoint Object Potential Map(VOP) of o_q around c_i on the original map(Fig 2(c)). VOP map helps us determine best viewpoints for object recognition and helps formulate the DT. A set $\mathcal{V}$ of such high accuracy viewpoints are determined for all the objects in $\mathcal{C}$. The VOP map and its importance in context with the problem is described in section 3.5.2.

5. The viewpoints ($\mathcal{V}$) from VOP map and the existential probability (P_{fi}) from GMM module are used to build a DT((Fig 2(d)) which would help in guiding the robot to strategic locations while it maximizes the utility and the reward obtained(Fig 2(e)). Hence, the cost incurred in the process of navigating through all objects in $\mathcal{C}$ is minimized.

6. If for all objects in $\mathcal{F}$, $P_i < P_t$, then the algorithm iterates after moving a finite step towards the objects, to gather more information about them.

3.3 Object Detection and Localization

The first function that is performed by the robot is to segment out objects from the floor. Through the approach presented in [1], we differentiate small objects (1-5cm) of interest from the floor, in which a state of the art superpixeling technique is used followed by a Graph Cut over the MRF formulation using the superpixels. This produces promising results in diverse set of environments(Fig 7). Further, the objects can be easily segmented out because the superpixels that are formed, align around the edges of the objects(Fig 7). It is important that the location of the objects is also known by the robot, to reach the objects. Since the camera height is fixed, objects can be fairly localized using the traditional pinhole projection approach [9].

3.4 Guess from far by GMMs

Small objects seen at a distance of few meters hardly take more than 20×20 pixels and hence are difficult to detect with features based object recognition methods. However it is still possible to discern the contour [16] and RGB texture. We build this module upon our previous work [20]. We construct 7-dimensional feature descriptors for each small or far away object images. Based on these descriptors, a GMM model is estimated for each object. In this work, modelling is performed over contours using Hu image Moments [15] and RGB histogram. The complete process of building the model is discussed in [20].

The GMM gives the likelihood P_{fi}, of correspondence between testing feature vector and object O_q. If P_{fi} (existential probability) crosses a threshold, then that object is examined from proximity. In section 3.6, we describe how P_{fi} over several images are used by a Bayesian Network to update the existential probability and further integrated with viewpoint object probability for object recognition decision process from distance.

Then comparison is done between the testing feature vector V_f' with all stored feature vectors V_l^n of the recognized object. By finding the best correspondence match with stored feature vectors, object's orientation in that view is estimated. This helps lay the VOP map for viewpoint planning explained in section 3.5.

3.5 Recognition from Near: Viewpoint Planning based Object Recognition

3.5.1 Object Recognition

As we deal with small objects, there is always a question

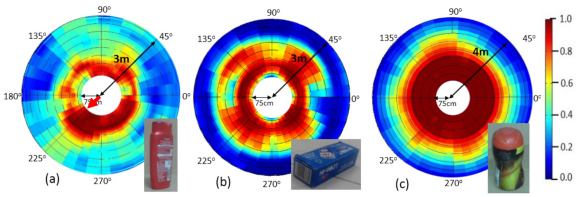

Figure 3: VOP map for (a)an object with a slim sideline and a wide object (b)cuboid shaped object (c)a symmetric object.

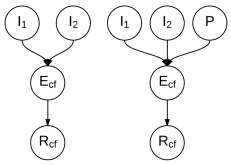

Figure 4: (a)Bayes' net to estimate existential probability of an object through two images. (b)Bayes' net integrates successful object recognition.

as to whether sufficient quality keypoints would be available for high precision object recognition. Thus, the choice of keypoint and descriptor becomes critical. It is shown in [20], that keypoints subsampled from the object point cloud yield higher accuracy, when compared to standard keypoints like SIFT-3D and Harris-3D. Experimentally proven in [20], we use PFHRGB descriptor [10] available in PCL library [11] and use a Visual-Bag-of-Words architecture over this descriptor to train an SVM classifier for accurate object recognition.

3.5.2 *ViewPoint Planning*

Viewing angle of an object plays a vital role in object recognition for human beings. Long and thin objects like doors when viewed from the narrow side, will be harder to recognize, as compared to when they are viewed from the front, where, additional distinguishing features are visible. In our experiments, we found that in the case of certain asymmetric objects, the recognition accuracy could vary drastically with viewing angle, as quantitatively shown in Section 4.3 .

Keeping this variance in mind, we use the data-structure proposed in [20] that indicates optimal viewpoints for high accuracy object recognition. The data structure, called the Viewpoint Object Potential (**VOP**), is a polar histogram that gives belief values for correct object recognition as a function of viewing distance and angle. The radius of the polar histogram varies from 75cm to an object dependent radius beyond which recognition probabilities fall to below 25%. The angle of the polar histogram is the viewing angle of the robot with respect to a 0° configuration, we decide on beforehand.

Fig 3 shows the VOPs of various objects. In the histogram, the colors indicate belief values as given by the scale on the extreme right of the figure. Red represents high belief values, and blue represents low. For example in Fig 3(a) the red arrow depicted indicates that if the object is viewed at an angle of 225° degrees and distance of 75-100cm has a very high probability of being recognized, as the arrowhead falls on a red bin. In 3(b), when the object is viewed at angles 45°, 225°, 315° at distances 75-100cm, the recognition accuracy is quite high. Similarly, Fig 3(c) shows that the recognition accuracy is uniform for all viewing angles as the object is symmetric.

3.6 Integrating GMM and VOP map

We now present our formulation for updating the VOP for an object of interest by integrating existential probabilities over multiple images. Consider the Bayes' network shown in Fig. 4. Here, I_1 and I_2 denote the images, which contain segmented object, O_f in $\mathcal{F}$. E_{cf} is a binary random variable which takes 1 when the GMM belief on the object is greater than a threshold(P_t). $P(E_{cf})$(referred as P_{fi} in Section 3.4) is the probability that the object O_f is similar to O_c in set $\mathcal{O}_s$. $P(E_{cf}|I_1)$ is typically the GMM score obtained over the segmented object in the image. The score computes the probability that the object O_c exists in the given image I_1. $R_{cf}(x, y, \theta))$ is a random variable that denotes if the object O_f can be recognized as O_s at a viewpoint (x, y, θ) of the robot surrounding O_f. This viewpoint is obtained from the VOP map of the object O_c. Then $P(R_{cf})$ is given by the VOP of the O_c for a pose (x, y, θ). The VOP around O_f is built using that of O_c as explained in section(3.5). Let Φ be the set of all poses around the object O_f where the VOP has non-zero value. The objective is to compute a pose $\phi^* \in \Phi$, wherein

$$\phi^* = argmax_{(\Phi)} P(R_{cf}(\Phi), E_{cf}|I_1, I_2) \quad (1)$$

In other words, we want to find the viewpoint within the VOP map area that has the maximum joint probability of detection and existence given two subsequent views I_1 and I_2 of the same segmented object. The above expression helps solve this decision problem under uncertainty, by propagating the belief about the presence of object over series of images I_1, I_2, I_3,...,I_n. All expressions are shown for only two image frames, though they are scalable over n frames.

We now derive the expression for the conditional distribution $P(R_{cf}, E_{cf}|I_1, I_2)$.

$$P(R_{cf}, E_{cf}|I_1, I_2) = \frac{P(R_{cf}, E_{cf}, I_1, I_2)}{p(I_1, I_2)} \quad (2)$$

$$P(R_{cf}, E_{cf}|I_1, I_2) = \frac{P(R_{cf}|E_{cf})P(E_{cf}|I_1, I_2)P(I_1)p(I_2)}{P(I_1)p(I_2)} \quad (3)$$

In the above equation, the numerator is derived from the Bayes' network in Fig 4(a). Hence,

$$P(R_{cf}, E_{cf}|I_1, I_2) = P(R_{cf}|E_{cf})P(E_{cf}|I_1, I_2) \quad (4)$$

The above expression upon application of Bayes' rule can be shown to reduce to

$$P(R_{cf}, E_{cf}|I_1, I_2) = \eta P(R_{cf}|E_{cf})P(E_{cf}|I_1)P(E_{cf}|I_2) \quad (5)$$

where η is the normalization constant. Hence, in general when there are n images,

$$\phi^* = argmax(\Phi)P(R_{cf}|E_{cf})P(E_{cf}|I_1)\dots P(E_{cf}|I_n) \quad (6)$$

The first term on the right(Eqn 6) is nothing but the VOP of the object O_c. The subsequent terms compute the existential probability of the object O_f as O_c over multiple views, where each such $P(E_{cf}|I_k)$ is computed from the GMM score.

For conciseness, we denote both the conditional and joint distributions as recognition probability distribution for the

rest of the paper. It is the probability of recognizing the object by 3-D point cloud analysis at a pose $\phi \in \Phi$.

At times, the object search algorithm is entailed to integrate the 3-D recognition probabilities apart from the GMM scores to compute the existential probability of the object. This is shown in the network(Fig 4(b)) where the classifier probability is shown as P_C. This is further elaborated in section (3.7) where its utility comes into play.

3.7 DT based object exploration

A Decision Tree(DT) is a Directed Acyclic Graph that computes a sequence of control actions which maximizes the expected utility over the graph. DT is typically represented by $DT = (V, E)$, a set of nodes V and edges E, wherein $V = V_c \bigcup V_d$ is the union of two disjoint sets of Chance nodes(V_c) and Decision nodes(V_d). $E = E_p \bigcup E_u$ is the union of Probability edges(E_p) whose weights are probabilities and Control edges(E_u). The leaf nodes of the DT contain the reward for choosing the path from the root node to the respective leaf nodes. Starting from the leaf node, the expected utility is calculated at every Chance node V_c bottom up [17]. Thus at every decision node V_d, the control sequence that maximizes the expected utility amongst all paths emanating from V_d is calculated.

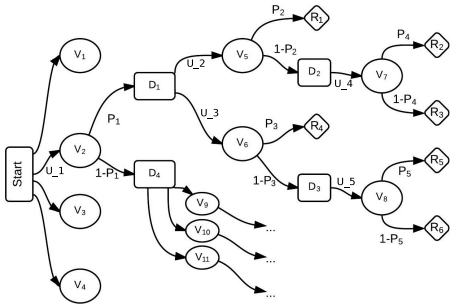

Figure 5: Decision tree depicting the various nodes and controls.

The problem of object search in our scenario is modeled as a Decision Tree(DT)(Fig 5) [17]. The root node(Start) is a Decision node which propagates a set of Control edges to Chance/Probability nodes. The Chance nodes(V_1, V_2, V_3, ...) are the locations on the VOP map of an object of interest from where the accuracy of recognizing that object is high. Each Chance node propagates to two Decision nodes through Probability edges p(Success) and $\bar{p}$(Failure). A Success edge(P_1, P_2, P_3, ...) denotes a successful recognition of the object through its 3-D point cloud in accordance with what was initially guessed about that object using the GMM. Similarly, a failure node(1-P_1,1-P_2,1-P_3, ...) denotes a failure to recognize the object in discordance with the initial guess of the GMM. The Decision node again propagates through a control edge(U_1, U_2, U_3,...) to a new location(Chance node) for further exploration and recognition of objects of interest. A Decision node occurring from a Success edge typically samples high accuracy locations from VOP maps of other objects of interest, whereas the Decision nodes arising from a Failure edge samples at least one high accuracy point from the VOP map of the same previous object along with the points sampled for other objects of interest. The rewards, form the leaf nodes of the graph and are inversely proportional to the distance traveled to reach the leaf node.

Here, we give an example of how the reward is calculated. The reward at the node R_1 for the path via $Start \rightarrow U_1 \rightarrow P_1 \rightarrow U_2 \rightarrow P_2 \rightarrow R_1$ (Fig.5), where it passes through two success edges(P_1, P_2) consecutively is $1/(d(Start, V_2) + d(V_2, V_5))$, where $d(Start, V_2)$ is the distance that needs to be traveled from the location of Start node to viewpoint V_2 when it executed the control action U_1. In other terms, the reward for R_1 can be put as $1/(d(U_1) + d(U_2))$. Whereas the reward through a path that failed to detect one or more objects progressively reduces with increasing number of non-detected objects along the path. For example, the reward along the path $start \rightarrow U_1 \rightarrow P_1 \rightarrow U_2 \rightarrow$ 1-$P_2 \rightarrow U_4 \rightarrow P_4 \rightarrow R_3$ is computed to be $1/d(U_1) + k * (d(U_2) + d(U_4))$ wherein k is a high gain used to reduce the reward due to non detection of the object V_5. However, the object viewed from V_2 was successfully detected with a probability P_1 and hence there is no high gain associated with the distance $d(U_1)$.

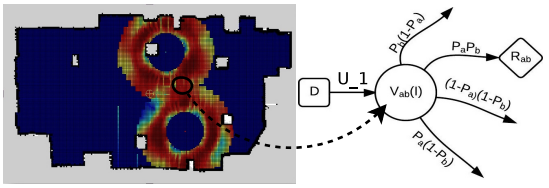

Figure 6: Part of Decision tree when VOP maps intersect

There could be cases where the VOP maps of two objects are overlapping. Then, a viewpoint in the intersection of their VOPs from where both objects have a considerable probability of recognition, could prove to be helpful(Fig 6). Here $V_{ab}I$ is a viewpoint in the intersection of VOPs of objects O_a, O_b. The reward for a path $D \rightarrow U_1 \rightarrow R_{ab}$ would be $1/d(U_1)$ as both the objects are recognized with a probability P_aP_b. This proves to be useful in the sense that both the objects are recognized in one go in a single control action. Note that the probability node $V_{ab}I$ has four probability edges emanating from it, corresponding to successful recognition of both objects(P_aP_b), recognizing one of them ($P_a\bar{P}_b$ or $\bar{P}_aP_b$) or failure to recognize both ($\bar{P}_a\bar{P}_b$).

The expected reward at each Chance node(from where the object needs to be viewed) is computed bottom up from each leaf node. At every Decision node, the control that maximizes the expected reward is chosen. The advantage of using a DT in the current formulation is twofold: Firstly, it provides a mechanism for integrating failure probabilities into the expected reward and hence eventual decision making. Secondly, it provides for alternative best paths, which can be computed a-priori. While the best paths are always along the success nodes, during the execution of such a best path, the robot can fail to recognize the object initially guessed by GMMs. Anticipating such situations, alternate paths are precomputed and stored along failure edges that the robot can anytime execute if it encounters a failure. During a failure event, the VOP of the object under consideration is updated by integrating the 3-D recognition failure probability through the Bayes-Net in (Fig 4), described in section 3.6. While this is done to maintain consistency of the joint probability distribution, in practice it does not change the alternate path precomputed by anticipating the failures and hence the DT is not updated at a failure event. Thus, the DT is updated only when a new object comes on the horizon and if GMM guesses it to be an object of interest. A new DT

is initiated once the queried object is found through the path computed over the previous DT was completely executed.

4. RESULTS

We show the performance of each of the modules in the system pipeline. Further, we validate the performance of the system through various experiments.

4.1 Object detection and segmentation

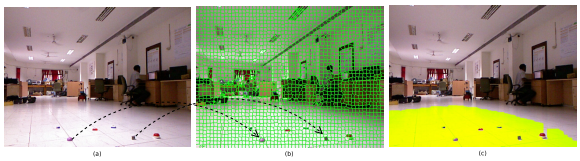

Figure 7: (a)A sample scenario. (b)Superpixelled scene. (c)Segmented Image.

In Fig 7, by using the approach presented in [1], objects of height 1-cm are classified as non floor through monocular images. The objects from these images are further segmented for further processing. [1] shows several scenarios where objects on the floor can be reliably extracted out of floor.

4.2 Probabilistic recognition using GMMs

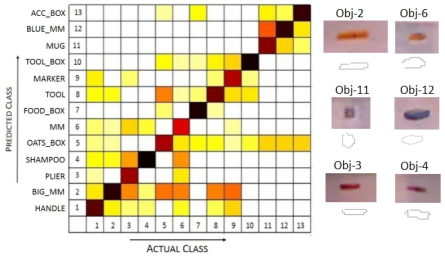

Figure 8: (Left)Confusion matrix for object recognition using GMM module. (Right) Sample contours for various objects.

On the whole, the GMM module is able to recognize small objects, with high accuracy as shown in the confusion matrix(Fig 8). Inaccuracies arise due to objects of similar texture and nearly identical shape contours at a distance. A confusion can be seen where the actual object class is 6 and predicted class is 2(Fig 8). If 2 is bigger in size than 6, they would appear similar when 2 is farther than 6 from the camera. Object 12 shows minimal confusion with 2, 3, 4, 6. Object 12 may be similar in shape compared to other objects, but its texture is clearly a differentiating feature. Also, a confusion exists between 11 and 12 due to the texture they share. The significance of GMM-Module in the pipeline is that, even when the objects are far and small, an early weak decision about a certain object's existence can be made.

4.3 Analysis of viewpoint based recognition

The PFHRGB based VBoW shows an improved performance over RGB-D Kinect Washington dataset [19] as well small object dataset [20]. Further, the VOP analysis shown in Fig 9 , illustrates the fact that even after successful detection and localization, the robot may end up not recognizing the object even at closer proximity. This happens because

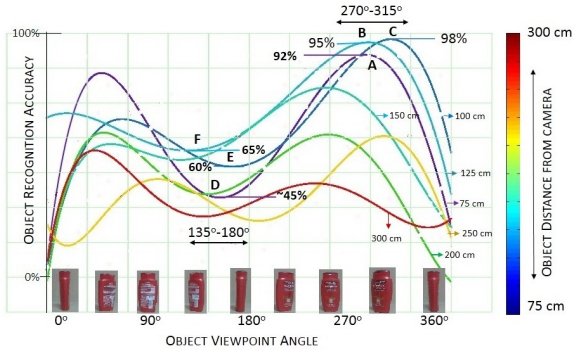

Figure 9: Recognition accuracy analysis of an object.

of weak viewpoint selection. Fig 9 shows one of the objects, for which object recognition accuracy is found to be vary significantly from different angles for the same camera-object distance range(75-125 cm). The variation captured shows accuracy change from $45-60\%$ for worst viewpoint as compared to $92-98\%$ for viewpoints chosen using proposed viewpoint planning method [20] for the same distance range. Therefore, if VOP maps (Fig 3) are known in prior, the performance of the object categorization algorithms can be enhanced for such mobile robotics applications. Section 4.5 illustrates the applicability of this method through numerous object search runs.

4.4 Analysis of the pipeline

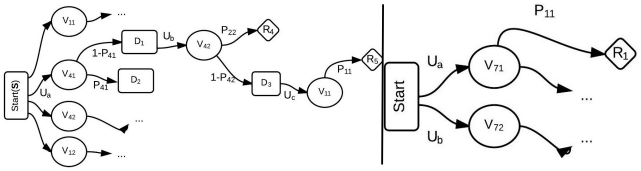

Figure 10: (Left)Control actions followed while recognizing objects O_1 and O_4. (Right) Control actions followed while recognizing O_7

This section demonstrates the overall pipeline, over a typical scene, through the performance of various components associated. In the scenario presented in (Fig 11.1), the robot is required to search two objects O_a and O_b. It starts from the position(S) as specified Fig 11.(d) and proceeds forward. Of the many objects visible, it estimates if any of them is similar to the queried object. At a certain iteration, it finds that objects O_1 and O_4 are similar to O_a. By overlaying the VOP map of object O_a around objects O_1 and O_4, it determines two best viewpoints for $O_1(V_{11}, V_{12})$ and for $O_4(V_{41}, V_{42})$(Fig 10). Then, it would try to choose the best path from the various paths that are possible to traverse the viewpoints(Fig 11.(d)). For this, a DT is built using the viewpoints as Chance nodes (Fig 10). The rewards and utilities are assigned to the nodes accordingly as specified in section 3.7. From the start(Decision node) it has to opt to go to any of the four viewpoints. It selects the control action U_a in Fig 10 to the viewpoint with highest utility. Once it reaches there, it recognizes the object O_4 using 3-D analysis and finds out that it is not object O_a. From the remaining three control actions, it further selects

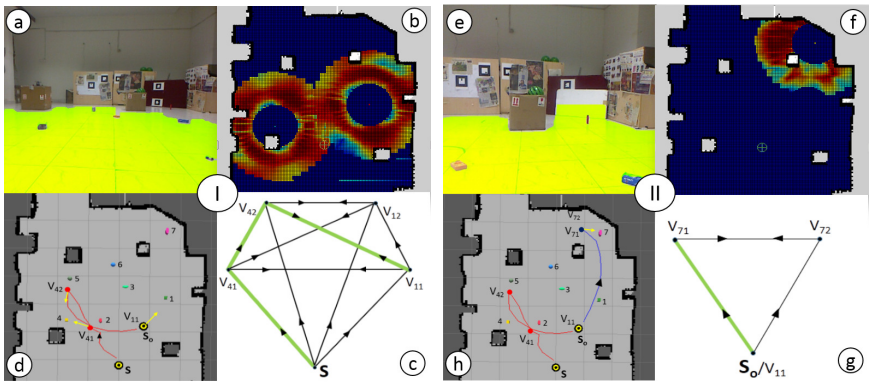

Figure 11: Figure depicts two iterations in a run where the robot searches for objects in the environment. (a,e) Object detection and localization. (b,f)VOP map based viewpoint planning. (c,g)Decision graphs for optimal control. (d,h)Path traced by the robot during object search.

the next best control U_b to reach V_{42}. In this trial as well, it fails to recognize O_4 as O_a, thus it concludes that O_4 is not O_a. Further, in the DT, it has two control options to choose, to reach either of V_{11} or V_{12}. It chooses the control action U_c to reach V_{11}. Here it successfully recognizes object O_1 as O_a and terminates the DT. After recognizing O_1, it explores the environment through the closest frontier [13] based exploration strategy to search the other object O_b. Incidentally, it finds an object O_7 which looks similar to O_b through GMM (Fig11.(h)). Hence the VOP map of O_b is laid around O_7 to determine the high accuracy viewpoints. Accordingly a new DT is constructed and the optimal path is picked. Here it recognizes O_7 as O_b as anticipated by the high GMM probability. And hence, the robot successfully finds both O_a and O_b in its search mission.

4.5 Comparative analysis signifying utility of each module

In this section, we analyze the performance gain accrued as each module is added to the pipeline through an experiment. In the experiment, the robot searches for three objects namely-a tool box, a multimeter and a shampoo in different settings. A comparative evaluation is performed for three cases: Initially, the robot seeks objects of interest without any prior knowledge of distant objects. Then, the robot is equipped with GMM module but lacks any kind of viewpoint planning. Finally, the robot is loaded with both the GMM module and the VOP based viewpoint planning, integrated into our DT formulation. After the 3 cases are explained, we give a quantitative analysis in Fig 13.

4.5.1 Case 1

The robot starts unequipped, without any early inference module like GMM. It therefore lacks any information about farther objects and can only plan up to a very short distance. It explores and examines each object present in range through a greedy approach by visiting the next closest object to recognize them. Each object is searched until the queried object is found. The robot ends up covering a very long distance due to this inefficient tour as shown in Fig 12(a) for all three objects. This inefficiency is especially pronounced for the shampoo search (Fig 12(a)). After taking several runs with various object orientations, we found that the object recognition accuracy was generally low due to bad viewpoint

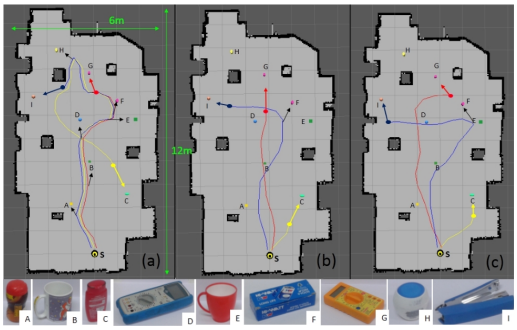

Figure 12: (a)Path traced during greedy object search. (b) Path traced using GMM module without viewpoint planning. (c)Path traced using the complete pipeline.

selection. In addition, we found the average distance covered by the robot was very high compared to the other cases. Over 10 separate runs on shampoo, when the robot landed on the worst viewpoint, the accuracy dropped as low as 30%.

4.5.2 Case 2

Equipped with the GMM module,the robot is able to rule out dissimilar objects from a distance. This reduces the search space to those objects that have a high belief of being the queried one, as given by the GMM. In this case, we found that, while the average trajectory length(Fig 13)was drastically reduced, but the recognition accuracy remained low. This again is due to poor viewpoint selection. As, can be seen in Fig 12(b) the path taken to find the shampoo is drastically reduced.

4.5.3 Case 3

Equipped with both the GMM and VOP module, the robot's trajectory length increased by a moderate amount. However, there was a consistent improvement in accuracy across the objects. In this case the robot trades off trajectory length for a more optimal viewpoint, explaining our observations. Fig 12(c) depicts one such run of this case.

The findings from the above cases are summarized in Fig 13. As can be seen in the trajectory length versus object plot, in case 2 and 3 there is a significant decrease in trajectory length, the largest being for object C(shampoo). Fig

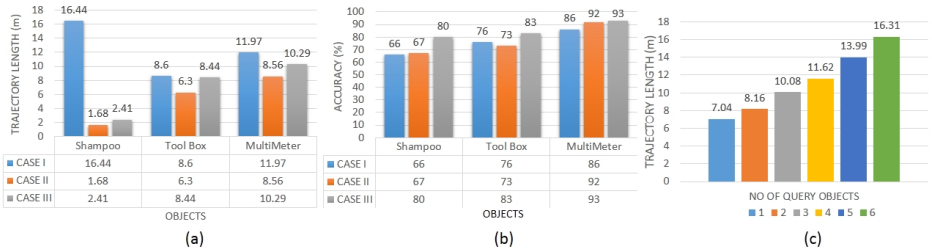

Figure 13: (a) Length traveled by the robot in various experiments in Case I, II, III. (b) Number of times objects have been found in various trials. (c) Distance traveled by robot in searching various number of objects.

13(b), the effect of VOP map shows an overall improvement in the recognition accuracy by 9% as compared to the case where the robot recognizes objects from random viewpoints. Analysis in Fig.13(c) show that as the number of queried objects are increased, the total distance traversed to find objects increases in nearly linear manner.

5. CONCLUSIONS

In this effort, we presented a complete pipeline, for a robot to efficiently search for small objects lying on the floor in large environments with clutter. To start off with, an initial search space reduction is done on the basis of texture and contour cues. Additionally, we improved object recognition by viewing them from their most discriminative viewpoints. For this purpose, we used a data-structure, the Viewpoint object potential(VOP), that led to a higher recognition accuracy. We combine VOP and initial search space reduction method into a DT that jointly optimizes the trajectory length and recognition accuracy. Through extensive experimentation, we showed that the robot strongly benefits from prior knowledge of the objects appearance at a distance, as well as viewpoint information, while still maintaining low trajectory lengths.

6. ACKNOWLEDGMENTS

We would like to thank the Department of Electronics and Information Technology(DeitY), India, for funding this work as a part of National Programme on Perception Engineering(NPPE, PhaseII).

7. REFERENCES

[1] S.Kumar, M.Siva Karthik and K.Madhava Krishna, *Markov Random Field based Small Obstacle Discovery over Images*, in IEEE ICRA, 2014.

[2] A. Aydemir, and P. Jensfelt, *Exploiting and modeling local 3D structure for predicting object locations*, in IEEE/RSJ IROS, 2012.

[3] A. Pronobis and P. Jensfelt, *Large-scale Semantic Mapping and Reasoning with Heterogeneous Modalities*, in IEEE ICRA, 2012.

[4] A. Aydemir, A. Pronobis, M. Göbelbecker, and P. Jensfelt, *Active Visual Object Search in Unknown Environments Using Uncertain Semantics*, in IEEE Transactions on Robotics, 2013.

[5] T. Garvey, *Perceptual strategies for purposive vision*, AI Center, SRI International, Menlo Park, CA, USA, Tech. Rep. 117, Sep 1976.

[6] K Sjöö, G L Dorian, P Chandana and P Jensfelt and D Kragic *Object Search and Localization for an Indoor Mobile Robot*, Journal of Computing and Information Technology, 2009.

[7] K. Sjöö, A. Aydemir, and P. Jensfelt, , *Topological spatial relations for active visual search*, Robotics and Autonomous Systems, 2012.

[8] R. Bajcsy, *Active perception*, Proc. IEEE, 1988.

[9] G. Stein, O. Mano, and A. Shashua, *Vision-based ACC with a Single Camera: Bounds on Range and Range Rate Accuracy.* IEEE Intelligent Vehicles Symposium, 2003.

[10] R. Rusu, N. Blodow, Z. Marton, and M. Beetz, *Aligning point cloud views using persistent feature histograms*, in IEEE/RSJ International Conference on Intelligent Robots and Systems, 2008.

[11] R. Rusu, and S. Cousins, *3D is here: Point Cloud Library (PCL)*, in IEEE ICRA, 2011.

[12] G. Csurka, C. Bray, C. Dance, and L. Fan, *Visual categorization with bags of keypoints*, Workshop on Statistical Learning in Computer Vision, ECCV, 2004.

[13] B. Yamauchi, *A Frontier Based Approach for Autonomous Exploration*, in Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation(CIRA), 1997.

[14] Chang, Chih-Chung and Lin, Chih-Jen, *LIBSVM: A library for support vector machines*, in ACM Transactions on Intelligent Systems for Technology, 2011.

[15] M.-K. Hu, *Visual pattern recognition by moment invariants*, IRE Transactions on Information Theory, 1962.

[16] J. Shotton, A. Blake, and R. Cipolla, *Multiscale Categorical Object Recognition Using Contour Fragments.*, IEEE PAMI, 2008.

[17] Finn V. Jensen and Thomas Nielsen, *Bayesian Networks and Decision Graphs (Information Science and Statistics)*, July, 2001.

[18] Bjorn Browatzki, Vadim Tikhanoff, Giorgio Metta, Heinrich H. Bülthoff and Christian Wallraven, *Active Object Recognition on a Humanoid Robot*, in IEEE ICRA, 2012.

[19] K.Lai, L.Bo, X.Ren, and D.Fox, *A Large-Scale Hierarchical Multi-View RGB-D Object Dataset*, IEEE ICRA, May 2011.

[20] S.Mittal, M.Siva Karthik, S.Kumar and K.Madhava Krishna, *Small Object Discovery and Recognition using Actively Guided Robot*, in ICPR, 2014.