An energy saving approach to active object recognition and localization

Andrea Roberti, Riccardo Muradore, Paolo Fiorini, Marco Cristani, Francesco Setti
Department of Computer Science, *University of Verona* – Verona, Italy
{andrea.roberti, riccardo.muradore, paolo.fiorini, marco.cristani, francesco.setti}@univr.it

***Abstract*—We propose an Active Object Recognition (AOR) strategy explicitly suited to work with robotic arms in human-robot cooperation scenarios. So far, AOR policies on robotic arms have focused on heterogeneous constraints, most of them related to classification accuracy, classification confidence, number of moves etc., discarding physical and energetic constraints a real robot has to fulfill. Our strategy overcomes this weakness by exploiting a POMDP-based AOR algorithm that explicitly considers manipulability and energetic terms in the planning optimization. The manipulability term avoids the robotic arm to get close to singularities, which require expensive and straining backtracking steps; the energetic term deals with the arm gravity compensation when in static conditions, which is crucial in AOR policies where time is spent in the classifier belief update, before doing the next movement. Several experiments have been carried out on a redundant, 7-DoF Panda arm manipulator, on a multi-object recognition task. This allows to appreciate the improvement of our solution with respect to other competitors evaluated on simulations only.**

***Index Terms*—Active object recognition, reinforcement learning, POMDP**

I. Introduction

In the robotics context, a correct scene interpretation plays a crucial role for the decision making process that follow. Think for example at collaborative robots that have to perform specific tasks while interacting with humans. Autonomous collaborative robots are usually asked to explore the space they are navigating, reconstruct the 3D structure of the environment, and understand the scene semantics. All these modules are part of a unique higher level task usually called *perception*, and will be used to decide which action to take and to plan the motion according to specific goals.

Active perception is a specific case on perception where the agent, also called *active perceiver*, dynamically determines its behaviour according to the goal of perceive — *i.e.* understand— the environment. The main benefit in using active perception instead of a static approach is that the confidence in recognition can be increased by dynamically modulating the overall agent's behaviour [3]. Among the several sources of information the perceiving agent is in contact with, objects are the main elements to reason about when trying to infer a semantic understanding of a visual scene [26].

Active Object Recognition (AOR) [24] consists in recognizing one or more objects with a moving sensor that can observe the scene under different points of view. Different disciplines have been involved with this topic. Computer vision techniques focus on how to cover the scene in order to capture maximally informative objects' views. Reinforcement learning studies planning strategies to minimize heterogeneous constraint, like classification accuracy, classification confidence, number of moves, etc.. Few of these approaches consider that the sensor has to be moved by a robotic arm [1]: this makes the optimization hard due to the highly nonlinear mathematical model of robotic manipulators.

In this paper, we propose an AOR approach that takes into account the robot's structure and capabilities. We model the sensor planning as a Partially Observable Markov Decision Process (POMDP). POMDP allows to sample the optimization space in a very efficient manner, and the objective function to minimize can be easily enriched with arbitrary terms. Specifically, here the robot should be able to recognize multiple objects by minimizing the estimated energy consumption and, at the same time, by considering manipulability constraints.

Energy constraints are important especially when the objects to recognize are more than one, requiring the robot to get many scene acquisition, in order to deal with occluded objects that are visible only from few particular points of view.

Manipulability constraints amount to let the robotic arm avoid singular configurations, that otherwise would force it to perform expensive and straining backtracking steps.

These two constraints have never been taken into account jointly in an AOR framework; moreover, most of the designed approaches have been tested on simulations. Our focus here is to move on real systems, and specifically on a redundant, 7-DoF Panda arm manipulator[1].

We report experiments with a real tabletop scenario with several objects belonging to four different, but similar, semantic classes. Results show that we outperform the state of the art in AOR both in object classification and in localization precision. Doing this, our method is also able to converge to a reliable prediction earlier than the competitors and with a higher confidence score. Energy spent in the process is also about 25% lower than the best competitor.

This work has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 779813 (SARAS project, www.saras-project.eu) and from the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation programme, grant agreement No. 742671 (ARS project). This work has been partially supported by the project of the Italian Ministry of Education, Universities and Research (MIUR) "Dipartimenti di Eccellenza 2018-2022".

[1]https://www.franka.de/

978-1-5090-6684-1/18/$31.00 ©2018 IEEE

The rest of the paper is organized as follows: in Section II we briefly review the recent literature on AOR; the problem formalization and the presentation of our approach are discussed in Sections III and IV, respectively. Finally, an experimental validation is reported in Section V. Conclusions are drawn in Section VI.

II. Related work

Active perception systems usually drive the robot by computing the best strategy (commonly called *policy*) to pursue a certain goal, *i.e.* the best camera trajectory to optimize a specific cost function. The process of choosing the best sequence of camera configurations can be *myopic*, quantifying only the reward at the next view, or *non-myopic*, maximizing the utility over a sequence of future configurations.

In traditional myopic approaches, actions are chosen by predicting the information gain of observations, commonly measured by the resulting reduction in uncertainty (entropy) and quantified with mutual information. Many approaches try to select the next best viewpoint within a set of candidates lying on a viewing sphere around the object reasoning on the information content. In [2], object appearance is represented by parametric eigenspaces, and probability distributions in the eigenspace are used to greedily select discriminative views. [8] proposes to learn an "attentive" interest map to track objects of interest using peripheral vision, while [18] addresses the next-best-view problem as the one that maximizes information gain increasing spatial resolution by changing the focal length of the camera. In [19], a belief model of the unobserved space is exploited to estimate the expected information gain of each possible viewpoint. Similarly, in [5], the next-best-view prediction is based on Hough Forests running on unsupervised features learned from depth-invariant patches using a sparse autoencoder. Entropy is exploited in several myopic approaches that control the camera motion via a greedy maximization strategy [4], by minimizing the conditional entropy [11], [23], or by reducing the differential entropy in the object pose and class distributions [6]. Despite being computationally less complex than the nonmyopic approach, myopic planning for an adaptively submodular objective function have been proved to be worse than the optimal strategy by a constant factor [7].

Long-term planning is usually modeled with reinforcement learning techniques, in order to reach a good trade-off between a higher classification accuracy and a lower cost for moving the robot planning along an optimal sequence of actions by accounting for a long to infinite horizon. In [16], an approximate policy that maps a sequence of received measurements to a discriminative viewpoint is obtained offline, but the cost of different actions is assumed to be uniform and does not apply well to real robotic scenarios. Here, as in many other works, the problem is formulated as a Markov Decision Process (MDP) where the state is not fully observable —*i.e.* the object class is unknown—, and thus its Partial Observability formulation is used (POMDP). In [9], a probabilistic model is used to encode structural relations among objects and locations. An object search task is then represented by fitting the probabilistic model with the visual appearance of the object of interest, and conditional entropy is used as the reward function. The approach of [1] formulates the problem as an active hypothesis testing problem solved with a point-based approximate POMDP algorithm. A similar approach that also includes a reasoning on the saliency of each viewpoint for the recognition of the object class is presented in [20].

POMDP-based approaches suffer from the main problem of being intractable in the continuous belief search space. Very recent works try to overcome this issue by employing Q-learning [14], Monte Carlo approaches [17] or deep learning [10]. In [17], a particle filter is combined with Gaussian process regression to estimate joint distributions of object class and pose, and predict sensor observations from future viewpoints. In [14] a layer of Dirichlet distribution is embedded into a Convolutional Neural Network (CNN) for modeling the distribution of beliefs for different object-action pairs. CNNs are also used in [10] for entropy regression and action prediction for the set of next viewpoint candidates. The optimal trajectory is then approximated by maximizing the sum of cross entropy over adjacent viewpoint pairs.

On the contrary, in this work we build upon a point-based algorithm to compute approximate solutions of the POMDP problem, exploring only a reachable subspace of the complete belief search space [12].

III. Problem formulation

Given a robotic arm, the objective is to actively classify unidentified objects in a tabletop scenario (Figure 1) with a good balance between the highest possible classification confidence and the lowest possible energy consumption. For this problem, the robot is equipped with a range sensor in an eye-in-hand configuration, and it is deployed in an environment with an *unknown number* of static objects with *unknown identities*.

The environment is composed of a finite number of objects, where each object is assumed to belong to a class c whithin a set of classes C of cardinality N_C. Each class is represented by a class prototype 3D model $\mathcal{M}_c$.

We formulate hypotheses $H(c, r)$ about the object class and orientation, which means we hypothesize that the object belongs to class $c \in C$ and its orientation is described by the rotation[2] $r \in SO(3)$ with respect to a canonical view.

At time t, the robot's end-effector, where the depth sensor is mounted, is in the location $\mathbf{x}_t = [x_t, y_t, z_t]$ with an orientation r_t in the 3D space. We refer to the 6DoF sensor configuration as *viewpoint*, $v_t = \{\mathbf{x}_t, r_t\}$. At each time step, the robot acquires a 3D point cloud $\mathbf{Z}_t$ of the entire scene that is partitioned into subsets $\mathbf{z}_t^i$ (using 3D segmentation) such that each subset corresponds to an object. Each point cloud subset is then processed by a static classifier that returns the hypothesis that best fits the input data $\mathbf{z}_t^i$ at time t.

The goal is to choose a sequence of viewpoints $\{v_0, \ldots, v_n\}$ with an optimal trade-off between the energy used to move

[2] $SO(3)$ is the Lie group of rotation matrices.

Fig. 1. The operative scenario. A robot is asked to actively recognize and localize all the objects on a table top.

the robot and the expected costs of incorrect decisions, as formulated in (1). The process ends when the cost for moving the robot to a new viewpoint is higher than the one for making an error in the hypothesis selection.

A. *Static hypothesis testing*

Given a segmented point cloud associated to an object in the scene, we need to estimate the class c it belongs to, and its orientation r on the table plane. We assume here, and in our experiments too, that all the objects are standing on the plane, so that orientation is represented by a single angle, θ, *i.e.* the rotation around the normal axis of the table plane.

For this task we employ the Viewpoint Pose Tree (VP-tree) algorithm proposed in [1]. This algorithm is an extension of the vocabulary tree, first introduced in [15], that extends the utility to joint recognition and pose estimation in 3D by using point cloud templates extracted from different viewpoints.

Training of the VP-tree is performed with simulated data. For each hypothesis, a set of point clouds have been generated with a simulated sensor. For each point cloud, Fast Point Feature Histograms [21] are computed on a set of uniformly distributed keypoints. Hierarchical clustering is then applied to subdivide features into groups of visual words.

At testing time, the training models are ranked according to the similarity of the query feature histogram and the training histograms. The top ranked hypothesis is assigned as the correct one.

IV. Energy Saving Planner

Our goal is to choose a sequence of viewpoints for the sensor $\{v_0, \dots, v_n\}$ optimizing the trade-off between the energy required to the robot to acquire new data and the expected costs of incorrect decisions. In this work we propose to model the problem as a Partially Observable Markov Decision Process (POMDP), that is an extension of a MDP to systems where the measurements do not allow a full observation of the system's state, or for systems with measurement noise. In this section we first define the cost, *i.e.* reward function, and then we present the POMDP formulation and an approximated solution strategy.

A. *Cost function*

We define the best strategy for the robot, also called *optimal policy*, as the one that minimizes the cost function:

$$\mathbb{E}\{J_H + \lambda_1 J_M + \lambda_2 J_E\} \quad \lambda_1, \lambda_2 \geq 0 \, , \tag{1}$$

where J_H is a term that accounts for correct hypothesis selection, J_M prevents from approaching singular configurations, and J_E takes into account the energy spent to acquire a new point-cloud. The expectional $\mathbb{E}\{\cdot\}$ is needed since all the indices are stochastic variables.

Specifically, the cost for an incorrect hypothesis selection J_H is:

$$J_H(\hat{c}, \hat{r}, c, r) = \begin{cases} K(\hat{r}, r), & \hat{c} = c \\ K^*, & \hat{c} \neq c \end{cases} \tag{2}$$

where K^* is an arbitrary cost associated to an incorrect categorization of the object class, while $K(\hat{r}, r) < K^*$ is the cost for an incorrect orientation estimate, when the class is correctly estimated. In our formulation $K(\hat{r}, r)$ is defined as the distance between rotations represented by unit quaternions. Let p and $\hat{p}$ be unit quaternions representing two rotations r and $\hat{r}$ in the same basis, the distance between rotations is the angle of the difference rotation represented by the unit quaternion $s = p\hat{p}^*$, where * denotes quaternion conjugation. Thus, the

$$K(\hat{r}, r) = 2 \, \arccos |\langle p, \hat{p} \rangle| \, , \tag{3}$$

with $\langle p, \hat{p} \rangle = p_1\hat{p}_1 + p_2\hat{p}_2 + p_3\hat{p}_3 + p_4\hat{p}_4$ and $|\cdot|$ the modulus function.

For defining J_M and J_E we need to introduce the mathematical model of a robotic arm. A n-degrees of freedom robotic manipulator can be described by a set of nonlinear differential equations

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau \, , \tag{4}$$

where $\tau = \begin{bmatrix} \tau_1 & \cdots & \tau_n \end{bmatrix}^T$ is the command torque vector, $q = \begin{bmatrix} q_1 & \cdots & q_n \end{bmatrix}^T$ is the vector of generalized coordinates (joint angles in the present case) with corresponding angular velocity $\dot{q}$ and acceleration $\ddot{q}$. In this standard Lagrangian representation, M is the symmetric non singular moment of inertia matrix, C is the Coriolis and centrifugal force matrix, and g is the gravity torque. Cartesian coordinates x and joint variables q are related by the forward kinematic function κ, i.e. $x = \kappa(q)$.

The term J_M accounts for robot's manipulability and prevents from reaching singular joints configurations by exploiting the redundant degrees of freedom. Let $man(q)$ be the manipulability ellipsoid defined as

$$man(q) = \sqrt{\det\left(J(q)\; J^T(q)\right)} \geq 0, \tag{5}$$

where $J(\cdot)$ is the Jacobian matrix [22]. Since $man(q) = 0$ corresponds to a singular configuration, J_M is defined as

$$J_M(q) = \frac{1}{man(q)}. \tag{6}$$

Lastly, the energetic cost J_E is related to the energy consumption. We define the energetic cost as the squared norm of the torque needed for the gravity compensation in static condition ($\dot{q} = \ddot{q} = 0$):

$$J_E(x) = \|g(q)\|^2 . \tag{7}$$

Remark 1. Formally, $\dot{q} = 0$ implies that the mechanical energy dissipated by the robot is equal to zero

$$\int \dot{q}^T(t)\tau(t)dt = 0. \tag{8}$$

However the i-th component τ_i of the command torque is generated by the i-th motor as $\tau_i = k_m I_i$, where I_i is the current and k_m is a characteristic constant in any DC motor. The electric energy is then proportional to the square of the current, *i.e.* $\propto (I_i)^2$, and so the energy cost is propostional to the square of the torque, *i.e.* $\propto (\tau_i)^2$. Taking the norm of the vector $\tau = \begin{bmatrix}\tau_1 & \cdots & \tau_n\end{bmatrix}^T$ related to the gravity compensation we end up with (7)

Remark 2. In this work we consider within the energetic contribution in the reward function (1) of the POMDP model only the stationary cost $\|g(q')\|^2$ related to the final position q'. In applications where it is important to weight also the specific trajectory to reach the target point, it would be possible to add to (7) a term related to the energy needed to move the robot from the starting position $q = q(t_i) = \kappa^{-1}(x(t_i))$ at time t_i to the final position $q' = q(t_f) = \kappa^{-1}(x(t_f))$ at time t_f

$$J_E(x) = \|g(q)\|^2 + \int_{t_i}^{t_f} \dot{q}^T(t)\tau(t)dt. \tag{9}$$

B. Partially Observable Markov Decision Process (POMDP)

Formally, a POMDP is a 6-tuple (S, A, T, R, Ω, O), where S is a finite set of states, A is a finite set of actions, $T : S \times A \to S$ is the transition function defining the probability of state change upon application of a given action. The reward function $R : S \times A \to \mathbb{R}$ represents the reward granted to the system after having reached the new state with the given action, Ω is a finite set of observations, and O is the probability distribution of the observations according to states and actions.

At each time step, given a current state $s \in S$, the agent receives an observation $o \in \Omega$ with probability $O(s, o) = Pr(o \mid s)$. Depending on this observation and the current state, the agent takes an action $a \in A$, which causes a transition to state s' with probability $T(s, a, s') = Pr(s' \mid s, a)$. Finally, the agent receives a reward r equal to $R(s, a)$. Then the process repeats.

Solving a MDP means to find an optimal policy mapping a state into an action that maximizes the expected total reward. However, since in a POMDP the state is partially observable, the concept of *belief* has to be taken into account. The *belief state* b is the probability distribution over all states; in our formulation, the belief state corresponds to the likelihood the robot assigns to an object to belong to all the classes in C, by taking into account not only the current observation, but also the whole history of observations. A POMDP policy π maps a belief b to a prescribed action a. A policy π induces a value function $V_\pi(b)$ that specifies the expected total reward of executing that policy π starting from b.

Starting from an initial distribution b_0, at every iteration the belief is updated using the formula:

$$b'(s') = \alpha\, O(s', o) \sum_{s \in \mathcal{S}} T(s, a, s')\, b(s) \tag{10}$$

where α is a normalization constant and all the new beliefs are guaranteed to be reachable from b_0.

The goal for the robot is to choose the optimal policy π^*, *i.e.* the policy that maximizes the associated value function $V^* = \mathbb{E}\left[\sum_t R(s_t, a_t)\right]$. In our formulation the state space of the POMDP is the discrete space of the sampled viewpoints ($\mathcal{V}$) and the continuous space $\mathcal{B} := [0, 1]^{N_h}$ of distributions over the N_h hypotheses. The main computational challenge comes from the exponential growth of the size of $\mathcal{B}$ with the number of hypotheses N_h. This problem can easily become computationally intractable, but approximate solutions have been proposed in the literature. In this work we resort to the SARSOP approach [12], which uses samples to compute successive approximations to the optimally reachable part of $\mathcal{B}$. The idea behind this algorithm is to sample a set of points from the belief space $\mathcal{B}$ and use it as an approximate representation of $\mathcal{B}$, instead of representing $\mathcal{B}$ exactly. For efficiency, most recent algorithms sample from $\mathcal{R}(b_0)$, the set of points reachable from a given point $b_0 \in \mathcal{B}$, under arbitrary sequences of actions.

Theoretical analysis shows that approximate POMDPs solutions can be computed efficiently when $R(b_0)$ has a small *covering number* [13].

V. Experiments

We evaluate the effectiveness of the proposed approach with four different objects in a real scenario. We implemented the proposed AOR framework using the Robot Operating System (ROS) [25] and a Panda arm manipulator from Franka Emika GmbH. The Panda arm is a redundant robot with 7 DoF, this allows us to have high dexterity and so to reach a larger number of poses in Cartesian space where we can examine the scene from different views. The robot is equipped with a Real Sense D415 mobile depth sensor mounted on the end effector with the optical axis approximately aligned with the last link of the arm.

We formulate hypotheses about 23 classes, including objects like 'pot', 'brush' and 'glass', and yaw angles from 0° to 330° with a step of 30°.

We restrict the motion of the sensor to a set of viewpoints $V(\rho)$ uniformly sampled on the surface of a hemisphere of

TABLE I

ACTIVE RECOGNITION RESULTS ON FOUR OBJECT CLASSES. THE BEST OPTION IS TO ACHIEVE THE HIGHEST BELIEF IN THE MINIMUM AMOUNT OF TIME (# STEPS) WITH THE MINIMUM ENERGY.

Object	Approach	Belief	# steps	Torque
Mug	OUR	0.92	3.33	30.76
	NVP [1]	0.75	6.67	40.83
Bottle	OUR	0.81	2.50	35.53
	NVP [1]	0.45	4.50	43.82
Handlebottle	OUR	0.95	4	45.10
	NVP [1]	0.77	8.66	62.38
Glasses	OUR	0.91	4.4	35.95
	NVP [1]	0.78	4.4	36.19

radius ρ centered at the location of the object. Each viewpoint assumes the sensor is oriented with the optical axis pointing to the centroid of the object. In our experiments ρ has been set to 60cm to have at the same time a good image occupancy, and as a consequence a good resolution of the point cloud, and some sort of safety to see the whole object[3].

We selected four different classes for our tests, with the aim to perform analysis on a challenging detection task. We decided to use three classes that can be easily confused between each other (*mug*, *bottle*, and *handlebottle*), while the fourth class (*glasses*) was proven to be the most challenging for the static detector VP-tree during the classifier training phase. For each class we fix the object location and orientation in a fixed configuration at the center of the robot's workspace, and we perform 5 separate runs by randomly changing the initial viewpoint.

In Table I we report active recognition results (averaged over five runs) on four classes in terms of

- prediction confidence (cumulative belief over all the hypothesis related to the correct class at the end of the process),
- number of viewpoints visited before providing a prediction (# steps), and
- energy consumption estimated during the process in terms of the squared torque needed for the gravity compensation in static conditions in all the viewpoints (Cartesian poses $\mathbf{x}_i$ corresponding to joint variables q_i) visited by the robot.

We provide comparison with a state-of-the-art method called Nonmyopic View Planning (NVP) [1]. Our proposed method outperforms the alternatives on all the object classes in all the aspects analyzed. In particular we want to point out that the results in Table I should be effectively interpreted in conjunction with Figure 2. This figure shows the process of exploration of the robot in terms of belief evolution and torque provided to the robot for each step. As an example of a common behaviour, we report here the averaged values on five runs with the class *handlebottle*. In the upper plot, we can

[3]We decided to set a constant ρ for all the classes in our experiments. Without loss of generality, one could decide to define a set of class specific hemisphere radius (ρ_c) to take into account different scales in typical object dimensions.

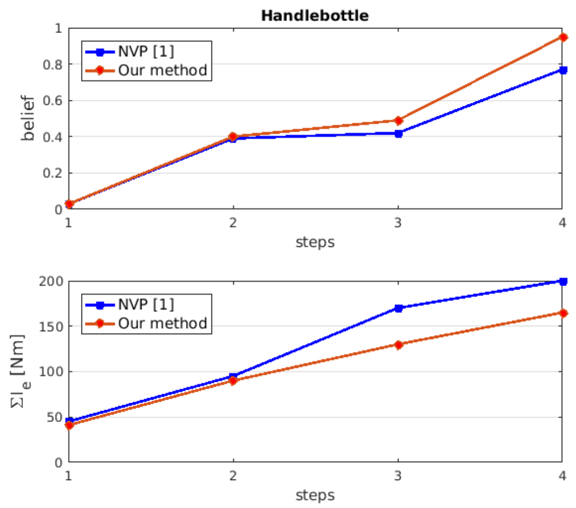

Fig. 2. Robot's navigation process in terms of hypothesis belief evolution (upper) and cumulative torque provided to the robot's joints to compensate gravity in each viewpoint.

clearly appreciate how the belief is consistently higher with our approach than with NVP, witnessing that the convergence of the process is quicker in our case. Moreover, the lower chart highlights that our method saves up to 25% of the energy.

As for the evaluation of classification accuracy and localization precision, we report in Table II the object classification accuracy, *i.e.* the percentage of correctly predicted object classes over the total amount of predictions generated, and the root means squared error in object localization and orientation estimation. Our method outperforms the state of the art both classification and localization accuracy, with a notable improvement in localization of about 50% on average. For orientation estimation the precision is mostly extremely good for our method as well as for NVP. The poor performance with handlebottle is due to the fact that one single run predicts (for both methods) an orientation that is 90° apart from the correct one. A deeper analysis of the video shows how, starting from that particular viewpoint, both methods are able to predict the correct class without looking at the handle, and thus having no information about the orientation of the target both methods output a 0° prediction for the yaw angle (while correct answer was 90°). Note that the classes *bottle* and *glass* have no associated orientation error since they are both symmetric objects.

VI. CONCLUSIONS

In this paper, we proposed an Active Object Recognition approach that takes into account the robot's structure and capabilities, modeling the sensor planning as a Partially Observable Markov Decision Process (POMDP) that allows to sample the optimization space in a very efficient manner. The proposed objective function is able to recognize multiple objects by minimizing the estimated energy consumption and, at the same time, by considering manipulability constraints.

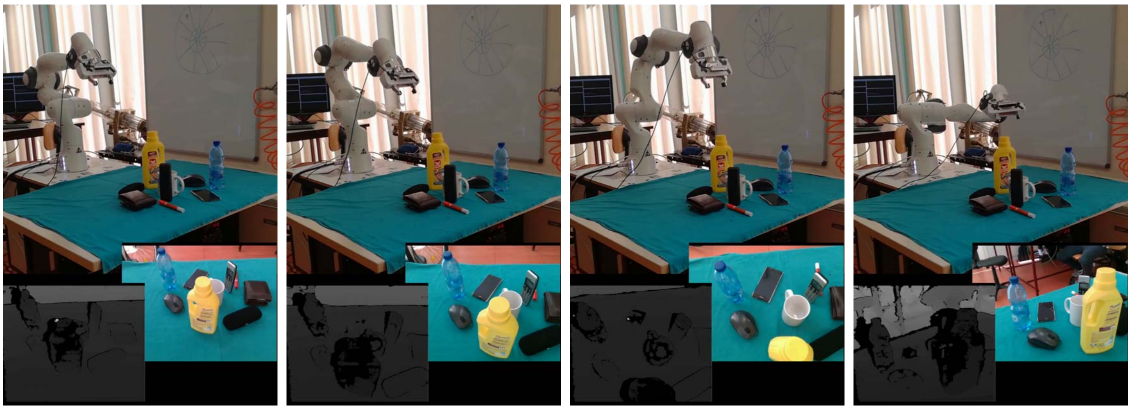

Fig. 3. Example of robot configurations and acquired images (both RGB and depth maps) trying to recognize a 'mug'.

TABLE II
OBJECT CLASSIFICATION ACCURACY (ACC), AND ROOT MEAN SQUARED ERROR IN OBJECT LOCALIZATION (e_{loc}, IN CM) AND IN ORIENTATION ESTIMATION (e_{ori}, IN DEGREES).

Object	Approach	Acc	e_{loc}	e_{ori}
Mug	OUR	1	0.6	0°
	NVP [1]	1	1.0	0°
Bottle	OUR	1	0.5	–
	NVP [1]	0.8	1.3	–
Handlebottle	OUR	1	1.2	16.6°
	NVP [1]	0.6	2.8	16.6°
Glasses	OUR	0.6	0.5	–
	NVP [1]	0.4	0.8	–

We provide experiments with a real arm manipulator, showing that we outperform the state of the art in AOR both in object classification and in localization precision. Doing this, our method is able to save about 25% of the energy with respect to the best competitor, while ensuring to converge to a prediction earlier and with a higher confidence score.

In this work we considered only the energetic cost to compensate the gravity at the visited viewpoints. This is reasonable, since most of the time is spent by the robot in these configurations while acquiring an observation and selecting the next action. In a future work we plan to include also a term that accounts for the specific trajectory to reach the target point as suggested in Eq. 9.

REFERENCES

[1] N. Atanasov, B. Sankaran, J. Le Ny, G. J. Pappas, and K. Daniilidis. Nonmyopic view planning for active object classification and pose estimation. *IEEE Transactions on Robotics*, 30(5):1078–1090, 2014.

[2] H. Borotschnig, L. Paletta, M. Prantl, and A. Pinz. Appearance-based active object recognition. *Image and Vision Computing*, 18(9):715–727, 2000.

[3] A. Coates, P. Abbeel, and A. Y. Ng. Learning for control from multiple demonstrations. In *ICML*, 2008.

[4] J. Denzler and C. M. Brown. Information theoretic sensor data selection for active object recognition and state estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2):145–157, 2002.

[5] A. Doumanoglou, R. Kouskouridas, S. Malassiotis, and T.-K. Kim. Recovering 6D object pose and predicting next-best-view in the crowd. In *CVPR*, 2016.

[6] R. Eidenberger and J. Scharinger. Active perception and scene modeling by planning with probabilistic 6d object poses. In *IROS*, 2010.

[7] D. Golovin and A. Krause. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *Journal of Artificial Intelligence Research*, 42:427–486, 2011.

[8] S. Gould, J. Arfvidsson, A. Kaehler, B. Sapp, M. Messner, G. Bradski, P. Baumstarck, S. Chung, and A. Y. Ng. Peripheral-foveal vision for real-time object recognition and tracking in video. In *IJCAI*, 2007.

[9] M. Hanheide, C. Gretton, R. Dearden, N. Hawes, J. Wyatt, A. Pronobis, A. Aydemir, M. Göbelbecker, and H. Zender. Exploiting probabilistic knowledge under uncertain sensing for efficient robot behaviour. In *IJCAI*, 2011.

[10] E. Johns, S. Leutenegger, and A. J. Davison. Pairwise decomposition of image sequences for active multi-view recognition. In *CVPR*. IEEE, 2016.

[11] V. Karasev, A. Chiuso, and S. Soatto. Controlled recognition bounds for visual learning and exploration. In *NIPS*, 2012.

[12] H. Kurniawati, D. Hsu, and W. S. Lee. SARSOP: Efficient point-based POMDP planning by approximating optimally reachable belief spaces. In *Robotics: Science and systems*, 2008.

[13] W. S. Lee, N. Rong, and D. Hsu. What makes some pomdp problems easy to approximate? In *Advances in neural information processing systems*, pages 689–696, 2008.

[14] M. Malmir and G. W. Cottrell. Belief tree search for active object recognition. In *IROS*, 2017.

[15] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *CVPR*, 2006.

[16] L. Paletta and A. Pinz. Active object recognition by view integration and reinforcement learning. *Robotics and Autonomous Systems*, 31(1):71–86, 2000.

[17] T. Patten, W. Martens, and R. Fitch. Monte Carlo planning for active object classification. *Autonomous Robots*, 42(2):391–421, 2018.

[18] R. Pito. A solution to the next best view problem for automated surface acquisition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(10):1016–1030, 1999.

[19] C. Potthast and G. S. Sukhatme. A probabilistic framework for next best view estimation in a cluttered environment. *Journal of Visual Communication and Image Representation*, 25(1):148–164, 2014.

[20] A. Roberti, M. Carletti, F. Setti, U. Castellani, P. Fiorini, and M. Cristani. Recognition self-awareness for active object recognition on depth images. In *BMVC*, 2018.

[21] R. B. Rusu. Semantic 3d object maps for everyday manipulation in human living environments. *KI-Künstliche Intelligenz*, 24(4):345–348, 2010.

[22] B. Siciliano and O. Khatib. *Springer handbook of robotics*. Springer Science & Business Media, 2008.

[23] E. Sommerlade and I. Reid. Information-theoretic active scene exploration. In *CVPR*, 2008.

[24] D. Wilkes and J. K. Tsotsos. Active object recognition. In *CVPR*. IEEE, 1992.

[25] Willow Garage. R.O.S., Robot Operating System. http://www.ros.org, 2018.

[26] M. Z. Zia, M. Stark, and K. Schindler. Towards scene understanding with detailed 3d object representations. *International Journal of Computer Vision*, 112(2):188–203, 2015.