Autonomous Question Answering with Mobile Robots in Human-Populated Environments

Michael Jae-Yoon Chung*, Andrzej Pronobis*, Maya Cakmak, Dieter Fox, Rajesh P. N. Rao

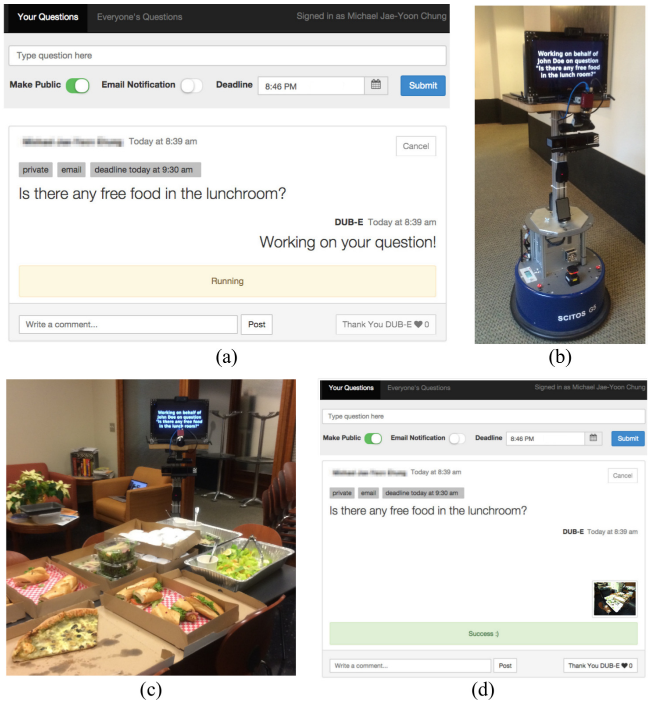
***Abstract*— Autonomous mobile robots will soon become ubiquitous in human-populated environments. Besides their typical applications in fetching, delivery, or escorting, such robots present the opportunity to assist human users in their daily tasks by gathering and reporting up-to-date knowledge about the environment. In this paper, we explore this use case and present an end-to-end framework that enables a mobile robot to answer natural language questions about the state of a large-scale, dynamic environment asked by the inhabitants of that environment. The system parses the question and estimates an initial viewpoint that is likely to contain information for answering the question based on prior environment knowledge. Then, it autonomously navigates towards the viewpoint while dynamically adapting to changes and new information. The output of the system is an image of the most relevant part of the environment that allows the user to obtain an answer to their question. We additionally demonstrate the benefits of a continuously operating information gathering robot by showing how the system can answer retrospective questions about the past state of the world using incidentally recorded sensory data. We evaluate our approach with a custom mobile robot deployed in a university building, with questions collected from occupants of the building. We demonstrate our system's ability to respond to these questions in different environmental conditions.**

I. Introduction

Many day-to-day tasks of occupants of large-scale human environments, such as office buildings, hospitals, or warehouses, require up-to-date knowledge about the state of the environment. However, human environments are also inherently dynamic. As a result, a large component of everyday tasks performed by humans in such environments is simply collecting up-to-date information.

A lot of information about our world can be found online. We can plan our time more efficiently by looking up the operating hours of a store or checking traffic conditions for a daily commute. However, a large portion of our world state is not as easily accessible (e.g. "Is food available in the downstairs kitchen?"). We believe that mobile robots hold the key to broadening the spectrum of dynamic environment knowledge that is readily available to human users.

In our previous work, we demonstrated the potential of mobile information gathering robots in this scenario using user surveys and Wizard-of-Oz deployment [1]. Inspired by the results, in this work we demonstrate an end-to-end framework capable of answering natural language questions about the state of a dynamic environment using an autonomous mobile robot. We use an image captured by the robot as a medium of choice to convey the information answering the question back to the user, since images can be analyzed quickly and accurately [2] while carrying rich context.



Signed in as Michael Jae-Yoon Chung

Type question here

Make Public ● Email Notification ○ Deadline 8:46 PM Submit

[redacted] Today at 8:39 am

private email deadline today at 9:30 am

Is there any free food in the lunchroom?

DUB-E Today at 8:39 am

Working on your question!

Running

(a)

(b)

(c)

Signed in as Michael Jae-Yoon Chung

Type question here

Make Public ● Email Notification ○ Deadline 8:46 PM Submit

[redacted] Today at 8:39 am

private email deadline today at 9:30 am

Is there any free food in the lunchroom?

DUB-E Today at 8:39 am

Success :)

(d)

Fig. 1. The end-to-end system: (a) a question is submitted through the web interface; (b) robot estimates the initial configuration, navigates to the destination while iteratively refining the estimate; (c) it captures the image containing the requested information; (d) the image is delivered to the user through the web interface.

Our framework assumes the availability of prior knowledge about the environment in the form of simple, semantically annotated 2D floor maps. This information is already available in many buildings, often in electronic form, e.g. indoor Google Maps. The system receives a natural language question posted by users through a web interface (see Fig. 1 for an example). Once a question arrives, the framework parses it and computes an initial robot configuration for capturing an image containing the information. To this end, our framework scores candidate robot configurations regarding visibility of requested information according to the semantic annotations and the previous, but possibly outdated knowledge about the state of the world. Then, the robot begins navigating to achieve the desired configuration while refining its estimate as more accurate and up-to-date information becomes available during the execution of the task. This refinement process allows the robot to compute a view maximizing the amount of captured information despite

* M. J. Chung and A. Pronobis contributed equally to this work.

The authors are with Computer Science & Engineering, University of Washington, Seattle, Washington 98195-2350, USA
{mjyc,pronobis,mcakmak,fox,rao}@cs.washington.edu

potential dynamic changes such as unexpected occlusions or obstacles that prevent the robot from reaching its initial goal.

We present an implementation of the proposed approach on a custom mobile robot deployed in a university building. We evaluate our system from several perspectives. First, we quantify the performance of the language parser itself with the natural language questions collected from the inhabitants of the building. We then evaluate the complete system on a selected, representative subset of questions for varying environment conditions. Finally, we demonstrate that the same framework can be used to answer questions retrospectively ("Was Mike Chung in the robotics lab?") by choosing the most relevant frame from incidentally recorded past data.

II. Related Work

Previous work has explored many applications of autonomous mobile robots in human environments. These include fetching objects [3]–[6], giving guided tours [7]–[9], escorting people to a target location [3], [10], or acting as a kiosk to provide information [11]–[13]. However, the use of such robots to answer people's questions about a dynamic environment has been largely unexplored.

Most closely related to our work is research focused on *object search* [14]–[19]. These approaches utilize domain knowledge about human environments and reason about potential target object locations. Some of them acquire parts of domain knowledge from the web [15], [19], or gather the knowledge required for each object search from the web on-demand [16]. While we consider search as a type of information gathering, our work focuses on a different type of information gathering task that involves checking the dynamic state of a target specified in terms of natural language. Our goal is to provide a human-centric, end-to-end experience by combining information gathering with a natural interface and handling tasks requested by real-world users. Another line of relevant works focus on understanding human language in the context of tasks in human environments, such as following natural language directions [20], [21] or spatial modeling of linguistic prepositions [22], [23].

Outside the realm of human-populated environments, the use of robots for information gathering is not a new idea. Mobile robots have been used for exploring and gathering information in challenging environments such as space, underwater, or disaster zones [24]–[27]. Some works [28]–[31] develop general algorithms for information gathering that could potentially be used in human-populated environments.

III. User-centered Design

We begun exploring the potential of mobile robots for answering questions about the environment from the end-user perspective [1]. We conducted several initial interviews and administered a survey to occupants of two buildings at the University of Washington. Our survey indicated that robots might provide a useful service and allowed us to formulate a coarse design of a practical framework.

To confirm our findings in practical situations, we deployed our robot in one of the buildings using the Wizard-of-Oz technique, i.e. the questions were interpreted manually and the robot was controlled by a human operator. As a front-end, we created a web interface through which users could post free-form questions and monitor the status of the answer. Users were recruited from graduate and undergraduate students, staff and faculty.

The experiment was conducted for four business days (9am-5pm). When a user asked a question using the web interface, the operator received and validated it. The operator then teleoperated the robot to a location where relevant information could be acquired and positioned the on-board camera to achieve the desired viewpoint. The question was answered by delivering the picture taken from the viewpoint together with a brief textual answer to the web interface.

Over the deployment period, we received 88 valid questions posted by 45 unique users. The majority of questions (71%) were concerned with the presence of things at certain locations in the building. Users were mostly interested in the presence of people (33%). Common examples of this type of question are "Is there anyone at {location}?" and "Is {person} in his/her office?" Among questions concerning objects in the environment, users were most interested in the presence of food and mail; e.g. "Is there anything in my mailbox?" and "Is there any food in the downstairs kitchen?". Another major group of questions was about the state of the environment at target locations. Questions ranged from checks about accessibility of various services (e.g. "Is the door to the conference room open?", "Is the reception still open?") to ambient conditions (e.g. "How noisy is it in the atrium right now?" or "Is it raining outside?").

The results of this formative study gave us insights into the types of questions people might ask if an autonomous framework was to be developed. Moreover, the intuition that images can be a powerful medium when conveying the answers to questions was supported by two users indicating that, while the text answer did not answer their question, they could extract the answer from the associated image.

IV. Framework

Our primary goal in this work was to design and implement an autonomous system realizing the tasks requested by the users during the aforementioned study, using the same end-to-end approach. To this end, we replaced the human operator with an integrated framework realizing all the steps from input question understanding to delivering the answer to the user. We settled on answers in the form of images aiming to achieve a balance between usability and reliability of the system. While extracting answers from images could also be automated, even with the recent image understanding methods, providing very accurate and context-rich image understanding remains a challenge in real-world settings. At the same time, this task can be solved robustly and efficiently by the end users [2] without significant impact on the experience [1].

A. Problem Formulation

The goal of the framework is to find the best robot camera configuration (viewpoint) v^*, for which the requested information is present ($\mathcal{I} = 1, \mathcal{I} \in \{0, 1\}$) in the captured image, given a natural language question q. We assume the robot is operating in a dynamic environment described by a body of coarse, domain-specific, static world knowledge W, which is unlikely to be affected by dynamic changes, as well as accurate, but potentially outdated, dynamic 3D map M^{3D}. We formulate our problem as

$$v^* = \underset{v}{\operatorname{argmax}} P(\mathcal{I} = 1 | v, q; W, M^{3D}). \tag{1}$$

We factorize $P(\mathcal{I} = 1 | v, q; W, M^{3D})$ as

$$\sum_z P(\mathcal{I} = 1 | v, z; W, M^{3D}) P(z | q; W) \tag{2}$$

$$\approx \max_z P(\mathcal{I} = 1 | v, z; W, M^{3D}) P(z | q; W) \tag{3}$$

where z is a descriptor of the information requested in the question. Factoring the problem in this way allows us to independently estimate the **viewpoint** given a descriptor with $P(\mathcal{I} = 1 | v, z; W, M^{3D})$ and the **natural language parse** of the question as a descriptor with $P(z | q; W)$.

For a question obtained through the web interface (Fig. 1a), the information descriptor z takes the form of a tuple $z = (l, t)$ where l describes the coarse location indicated in the question (e.g. a room $cse101$), and t is a target specifying the object presence or the state of the environment relevant to the question (e.g. $person$, $stapler$ or $occupied$). We then define $P(z | q; W)$ as a distribution over information descriptors z for a question q. For example, given the question $q =$ "Is Mike Chung in his office?" and the record in W that identifies Mike Chung as $person$ and Mike Chung's office as $cse101$, the desired information descriptor z is a tuple $(cse101, person)$. Since we defer the image interpretation to the end user, we do not need to make a distinction between instances (Mike Chung) and categories (person) of objects in question. Then, the desired output v^* is a robot pose near or in the room CSE101 that provides the best viewpoint for seeing a person, despite occlusions due to moving people, chairs, doors or cubicle partition panels.

While we assume that the static domain knowledge is sufficient for parsing, we cannot make that assumption about navigation and viewpoint estimation in dynamic environments. Therefore, we tightly couple the task execution process with the continuous refinement of the best viewpoint estimates with the help of the constantly updated M^{3D}. We begin with an initial prior on the 3D map estimated purely from the coarse static information in W. That is sufficient to compute an initial v^* to which the robot can start navigating (Fig. 1b). However, M^{3D} is likely to be inaccurate or outdated due to dynamic changes. As a result, v^* may not provide the best viewpoint with respect to the current state of the environment or the viewpoint might be inaccessible. We address this problem by letting the system continuously update M^{3D} during task execution and re-evaluate $P(\mathcal{I} = 1 | v, z; W, M^{3D})$ to update the target navigation goal and camera configuration accordingly. This process continues until there is no change in the estimate v^* and the robot reaches the goal. Note that we do not re-compute $P(z | q; W)$, since language parsing requires only static world knowledge W. Once the robot reaches the final v^*, it captures an image using the on-board camera (Fig. 1c) and returns this image as its response (Fig. 1d).

B. *World Model*

As described in Sec. IV-A, our world model consists of a static component W and a dynamic component M^{3D}. Furthermore, we introduce a topological map M^T in which each topological node serves as a candidate *place* from which a set of discrete candidate *viewpoints* originate. When computing Eq. 1, we search for v^* only among the viewpoints in a M^T which makes our problem tractable and allows for real-time continuous updates to the viewpoint estimates during task execution. Below, we describe each component in detail.

1) Static World Knowledge: The static component W is a tuple (M^{2D}, M^S, D). M^{2D} is a 2D occupancy grid map with a resolution of 0.05m in which each grid cell is is either empty, occupied, or unknown. M^{2D} is acquired by mapping the environment using a method developed by Grisetti et al. [32] and post-processed to only contain static information (e.g. walls and stationary furniture). Obtaining such maps, even for large environments, is easy, yet they provide sufficient amount of information in our framework for global navigation planning and anchoring semantic annotations.

M^S is a representation of static semantic information about the environment anchored in the 2D metric map M^{2D}. It takes the form of groups of the 2D map cells associated with semantic symbols compatible with the descriptor $z = (l, t)$. We use polygon regions on top of M^{2D} associated with symbols in l to encode spatial extent of semantic information about locations, e.g. a region describing a room CSE101. We use distributions over cells in M^{2D} associated with symbols in t to express the likelihood of presence of semantic target at a cell, e.g. a distribution modeling the presence of a person or object over 2D map cells near a desk. A large portion of such static semantic annotations is already available in electronic form in many buildings (e.g. floor plans or indoor Google Maps). Additional annotations can be placed manually or with the help of semantic mapping algorithms (e.g. [33]). Often, the benefit in terms of reliability will outweigh the one-time effort required to provide such annotations.

Finally, D is a relational database of domain specific knowledge about the environment. In our implementation, the database was generated from existing databases of employees of the building and offices they were assigned to. It is used primarily to support language understanding and enrich the semantic information captured in M^S. For instance, in order to map the question "Is Mike Chung is his office?" to a set of viewpoints that could provide an answer, we can combine the annotations in M^S of room CSE101 and locations likely to contain a person, with the knowledge in D that Mike Chung is a person assigned to office CSE101.

2) Dynamic World Knowledge: The dynamic component M^{3D} of our world model is a 3D occupancy grid map, aligned with M^{2D}, with a resolution of 0.05m in each of the three dimensions. M^{3D} provides a richer representation of the environment; however, in dynamic environments it can

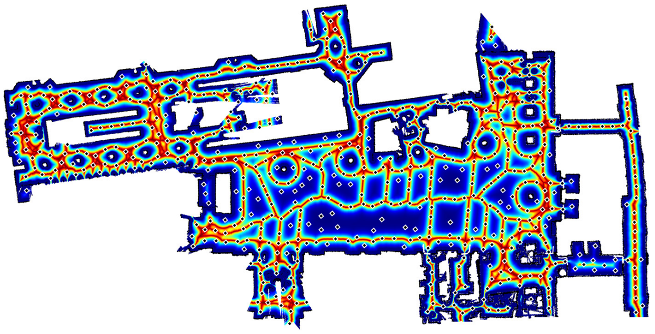


Fig. 2. A typical example of a set of topological nodes on top of the values of $\phi_G(N_i)$ for each pixel of an occupancy grid map.

quickly become outdated. Hence, we continuously update it with incoming depth data from RGB-D sensors using the approach of Hornung et al. [34]. M^{3D} is the representation that stores the most up-to-date, detailed environment knowledge collected over the course of the robot's deployment and is used for reasoning about visibility.

To obtain an initial 3D representation required for viewpoint estimation, before any 3D data is acquired, we rely on the static information stored in W. We transform W into the coordinate system of M^{3D} by extending the 2D information in M^{2D} and M^S into the third dimension. The occupancy information in the new dimension is generated based on the semantics of the objects. Walls and windows separating rooms in M^{2D} are assumed to have a fixed height (2m and 0.85m in our environment), while other objects annotated in M^S are assigned with a Gaussian distribution modeling their approximate size and distance from the ground with μ and σ depending on the semantic class. These distributions are then used to sample initial occupancy information.

3) Topological Map: The topological map allows the framework to constrain the problem of viewpoint estimation to viewpoints originating from a discrete subset of all possible locations in the 2D map. This makes the problem tractable, but also results in a commitment that could harm performance. Therefore, it is important to select a discretization that properly supports the problem at hand.

We generate topological maps from a probability distribution that models the relevance of 2D grid map locations to the task and distributes topological *places* accordingly. The distribution is defined as:

$$p(N|M^{2D}) = \frac{1}{Z}\prod_i \phi_R(\mathscr{N}_i)\phi_G(N_i),$$

where $N_i \in \{0,1\}$ determines whether a place exists at location i in the 2D map and $\mathscr{N}_i = \{N_j : j \in \text{neighborhood}(i)\}$ is a set of N_j for a local spatial neighborhood of 1m radius.

The potential function $\phi_G(N_i)$ models the relevance of a location for the task and is defined in terms of three potentials calculated from the 2D occupancy map M^{2D}:

$$\phi_G(N_i) = \phi_O(N_i)\left(\phi_C(N_i) + \phi_V(N_i) - \phi_C(N_i)\phi_V(N_i)\right),$$

where:

- ϕ_O depends on the distance d_i^O between location i and the nearest obstacle and equals 0 for distance smaller than the radius r of the robot and $\exp(-\alpha(d_i^O - r))$ otherwise. The cost function and α are the same as in the local planner of the robot's navigation algorithm [35].
- $\phi_V = \exp(-\beta|d_i^O - d^V|)$ depends on the relation between the distance d_i^O and a fixed distance d^V that provides good visibility of obstacles and is determined by the camera parameters.
- $\phi_C = \exp(-\gamma d_i^C)$ depends on the distance d_i^C of location i to the nearest node of a Voronoi graph of the 2D map. This promotes centrally located places since central locations are often safe for navigation.

Overall, the definition of $\phi_G(N_i)$ ensures that candidate *places* are located only in areas that will not lead to a collision with permanent obstacles and are either preferred due to their central location or good visibility of objects of interest (usually located near occupied cells on the 2D map). The potential $\phi_R(N_i)$ ensures that places are spread with a certain distance to one another, by enforcing low probability for locations that are close to other existing places. The resulting places, despite being generated based on the static elements of the environment (e.g. walls, permanent furniture), provide sufficient coverage, even when considering the presence of dynamic objects. At the same time, the topological map eliminates a large portion of irrelevant locations.

We employ Gibbs sampling to perform the maximum a posteriori inference and choose samples corresponding to topological maps with highest posterior probability. A typical example of a generated set of topological nodes for a single floor of a building is shown in Fig. 2. For each place, we assume a discrete set of orientations evenly spread across the full circle (in our implementation, every 30°), which together with the metric position of a place fully specify a *viewpoint*.

C. Parsing Natural Language Questions

Parsing a language input question q to an information descriptor $z = (l, t)$ is equivalent to evaluating $P(z|q;W)$. We first process q by using Stanford CoreNLP Natural Language Parsing Toolkit [36] to extract part-of-speech (POS) tags and a context-free phrase structure tree, and apply co-reference resolution. We merge all outputs from the CoreNLP to a parse tree q' by copying the output parse tree and replacing its leaf nodes with the input words and the POS tag pairs. We then use the results from the co-reference resolution to replace the subtrees corresponding to the referring words with the subtree corresponding to the referred words. For example, given q = "Is Mike Chung in his office?", the sentence extracted from q' is "Is Mike Chung in Mike Chung's office?".

Given q' we evaluate:

$$P((l,t)|q';W) \propto \max_i \mathbf{1}(T_i(q'))$$
$$\times \left\{ \max_j d(L(T_i(q')), A_j(l)) + \max_k d(G(T_i(q')), B_k(t)) \right\},$$

where:

- $T_i(q')$ is an ith relation template that can detect words describing a location and a target type in q'. Templates use relationships between tags (e.g. check if a node has children with tags PP and NP) and predefined keywords (e.g. check if a word paired with a IN POS tag equals to the locational preposition such as "in", "at", etc.) to detect the words. $\mathbf{1}(T_i(q'))$ returns a boolean variable indicating whether $T_i(q')$ fits on q' or not.

- $L(\cdot)$ and $G(\cdot)$ operators return detected words describing a location and a target type, respectively, from applying a template $T_i(q')$. Using the q' mentioned earlier, $L(T_i(q')) =$ "Mike Chung's office" and $G(T_i(q')) =$ "Mike Chung" for some i.
- $A_i(l)$ returns ith name describing l and $B_j(t)$ returns jth name describing t by looking up the data stored in the domain knowledge database D. For example, $A_i(cse102) =$ "Mike's Office" and $B_j(person) =$ "Mike Chung" for some i, j.
- $d(\cdot,\cdot)$ function measures the similarity between two text inputs (e.g. Levenshtein distance).

In Sec. IV-A, we approximate the summation in Eq. 2 with the max (Eq. 3). In other words, we are only considering the most likely information descriptor instead of all possible information descriptors in order to ensure real-time performance when scoring viewpoints during 3D map changes.

If the input sentence is not an information checking question, then the distribution $P((l,t)|q';W)$ will not be proper; no relation templates $T_i(q')$ can cover the input q'. We can use this property to detect valid questions.

We evaluated the ability of the parser to (i) detect valid questions and (ii) predict an information descriptor given sentence on the real user questions collected during the deployment experiment described in Sec. III. The labels for the questions were acquired by a coding process performed by two of the authors. Labeling involved adding an information descriptor $z^* = \operatorname{argmax}_z P(z|q;W)$ for each question q. For the valid question detection task, we attained an accuracy of 74%, a precision of 94% and a recall of 71% (# of true positives: 48, true negatives: 17, false positive: 3, false negatives: 20). For the 65 questions that were correctly identified, we evaluated our system's ability to extract the corresponding information descriptor. Our parser achieved an accuracy of 72% in correctly classifying the full information descriptor $z = (l,t)$.

D. Viewpoint Estimation

As mentioned in Sec. IV-A, estimating the best viewpoint providing answer to the question asked by the user is equivalent to evaluating $P(\mathcal{I} = 1|v,z;W,M^{3D})$ for the given information descriptor z. Importantly, in our system, this evaluation is performed continuously and in parallel to task execution as M^{3D} is updated, resulting in updates to the target navigation goal and camera configuration. The distribution $P(\mathcal{I} = 1|v,z;W,M^{3D})$ can be decomposed as follows (W, M^{3D} omitted to keep notation uncluttered):

$$P(\mathcal{I} = 1|v,z) = \sum_x P(\mathcal{I} = 1|x,z)P(x|v) \qquad (4)$$

where x are the cells in the coordinate system of the 3D occupancy map M^{3D}.

The first term $P(\mathcal{I} = 1|x,z;W,M^{3D})$ models the presence of the information specified by descriptor z at cell x by combining the semantic information in M^S with the current state of the 3D occupancy map and the information in M^{2D} projected into three dimensions for the parts of the environment for which a detailed 3D map has not been acquired yet (see Sec. IV-B.2 for details). The annotations in M^S are compatible with symbols in $z = (l,t)$ and can be expressed directly in terms of probabilities assigned to the occupied cells of the dynamic 3D representation. We use $P(\mathcal{I} = 1|x,l;W,M^{3D}) = 1$ for every occupied cell of every vertical column within the polygon region describing location annotation l. For target annotations t, we assign $P(\mathcal{I} = 1|x,t;W,M^{3D})$ to the value of the distribution over 2D map cells associated with the annotation, uniformly distributed across all the occupied cells of the vertical column. We assume independence between the two types of annotation (W, M^{3D} omitted): $P(\mathcal{I} = 1|x,z) = P(\mathcal{I} = 1|x,l)P(\mathcal{I} = 1|x,t)$.

The second term in Eq. 4, $P(x|v;W,M^{3D})$, models the visibility of cell x from viewpoint v. We compute it using raytracing in the most up-to-date 3D representation. We assume a fixed angular resolution when projecting rays from the origin of the viewpoint (100×100 rays within the field of view of the robot's camera). This approach provides a heuristic behavior promoting viewpoints that are neither too close nor too far from the target. Viewpoints that are too close miss information outside the field of view, while viewpoints that are too far observe the target with low resolution. We place a threshold on the maximum length of a ray to be 15m and assume that cells x that are hit by one of the rays without occlusions are visible from the viewpoint v. Finally, while the robot is executing the task, it might discover that certain v is not reachable (cannot be navigated to). For such v, we set $P(x|v;W,M^{3D}) = 0$ for all x.

V. Experiments and Results

We evaluated our end-to-end framework in two question answering scenarios with a mobile robot deployed in an office building: autonomous information acquisition and retrieval of answers from previously collected data.

A. Experimental Setup

The framework was deployed on a custom-built mobile robot based on the MetraLabs Scitos G5 mobile base expanded with a structure providing support for sensors and user interfaces (Fig. 1b,c). A high-resolution camera with 97° horizontal and 79° vertical field of view at the height of 1.31m was used for providing images to the users. An Asus Xtion Pro depth camera placed at 1.25m above the ground was used to collect depth images for the purpose of building the 3D map and navigation. Hokuyo UTM-30LX laser range finder was used for navigation and 2D map building. Another backward facing Xtion depth camera was also placed onboard to assist navigation in tight spaces.

Our experiments were conducted in the building of the Computer Science & Engineering department of the Uni-

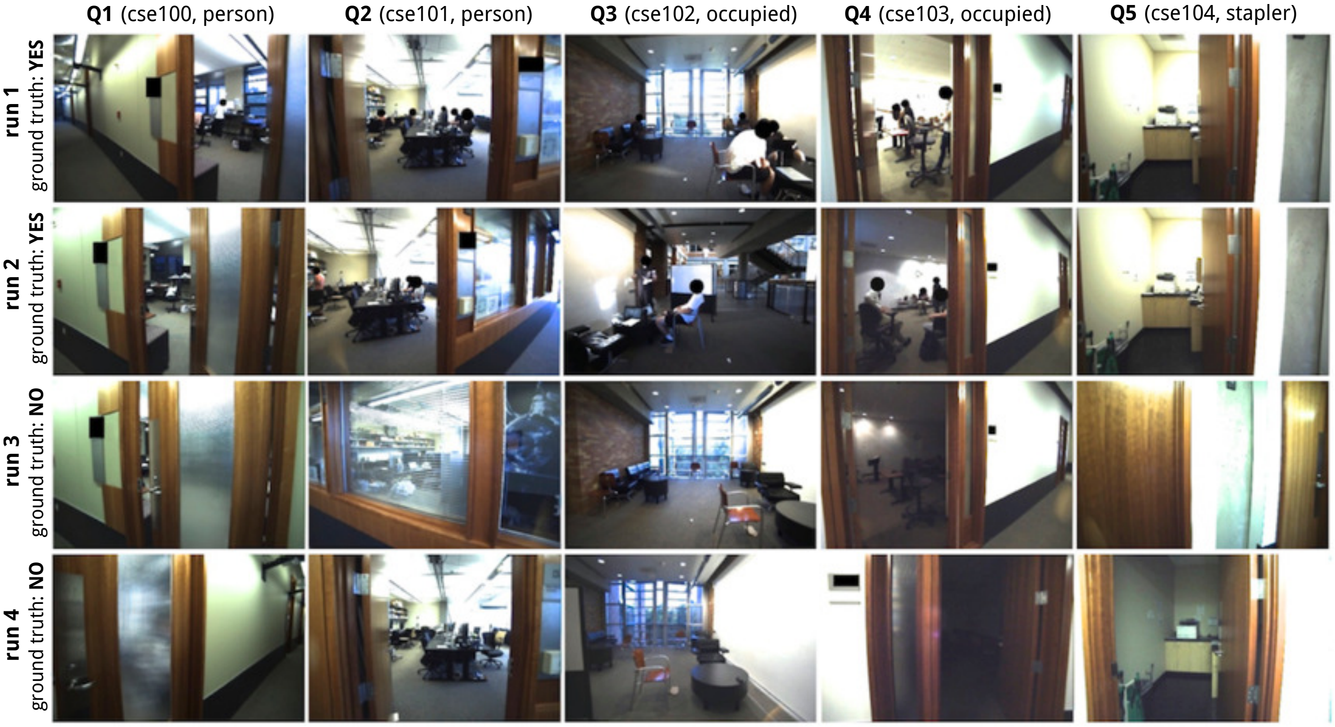


Fig. 3. Images returned as answers from the four runs of Experiment I. The first two rows show images from the runs with the ground truth answer "yes" and the next two rows show images from the runs with the ground truth answer "no" for the corresponding checking questions (columns). The column headers display the inferred information descriptors for the corresponding questions (Q1–Q5).

versity of Washington in Seattle. The building provided an interesting experimental environment with open spaces containing movable chairs, tables and whiteboards for students and visitors (breakout areas) as well as labs and offices separated by walls with large windows. The 2D occupancy maps of the building (M^{2D}) were collected prior to the experiments. In the maps, we limited access of the robot to the the publicly accessible spaces such as corridors and breakout areas to avoid interrupting office occupants during working hours. However, we still reasoned about visibility of the inaccessible space, making it possible for the robot to look inside through open doors or windows. 295 location annotations in M^S were imported from the building floor plan and the target type annotations were placed manually.

B. Experiment I: Information Acquisition

First, we evaluated the ability of our framework to acquire and deliver relevant images as answers. We used questions frequently asked by real users during the Wizard-of-Oz deployment (Sec. III):

Q1. *Is {person} in his/her office?*
Q2. *Is there anyone in the mobile robotics lab?*
Q3. *Is the breakout area occupied?*
Q4. *Is the conference room occupied?*
Q5. *Is there a stapler in the printer room?*

The corresponding information descriptors from the language parser are shown as column headers in Fig. 3. We ran the system four times throughout the day for each question. We chose the timing of the runs so that the ground truth answer was twice "yes" and twice "no". However, we did not control the visibility and reachability conditions of the target locations to capture natural variations in the environment. Fig. 3 shows the images returned as answers for each run.

Handling dynamic changes. Fig. 4 illustrates how the viewpoint estimation algorithm adapts to dynamic changes in the environment by selecting alternative viewpoints with similar information content. For Q2, the robot was able to capture the interior of a lab through its door when it was open, but also through its windows when the door was closed and the blinds on the window were open (run 3). Similarly for Q3, the robot navigated to the other end of the breakout area and turned around to capture the area, when it encountered a whiteboard blocking the view from the initial viewpoint estimated based on prior, outdated information.

Viewpoint quality. To better assess the quality of the viewpoint estimates from the point of view of the end user relying on the captured images as question answers, we conducted a user study among 10 building occupants. For each run, we asked the participants to respond to the corresponding questions (Q1-Q5) using only the returned image as a cue. Response options were "definitely yes", "probably yes", "I don't know", "probably no", and "definitely no". Fig. 5 shows the results from the user study. We consider a response to be correct if a user responds with "definitely yes" or "probably yes" when the ground truth answer is "yes" (similarly for "no"). We consider a response wrong if the user's answer contradicts the ground truth and undecided if the user responds with "I don't know."

Overall participants achieved a high classification accuracy, particularly for questions Q2, Q3, and Q4. As can be seen in Fig. 3, high undecided rates and non-zero incorrect

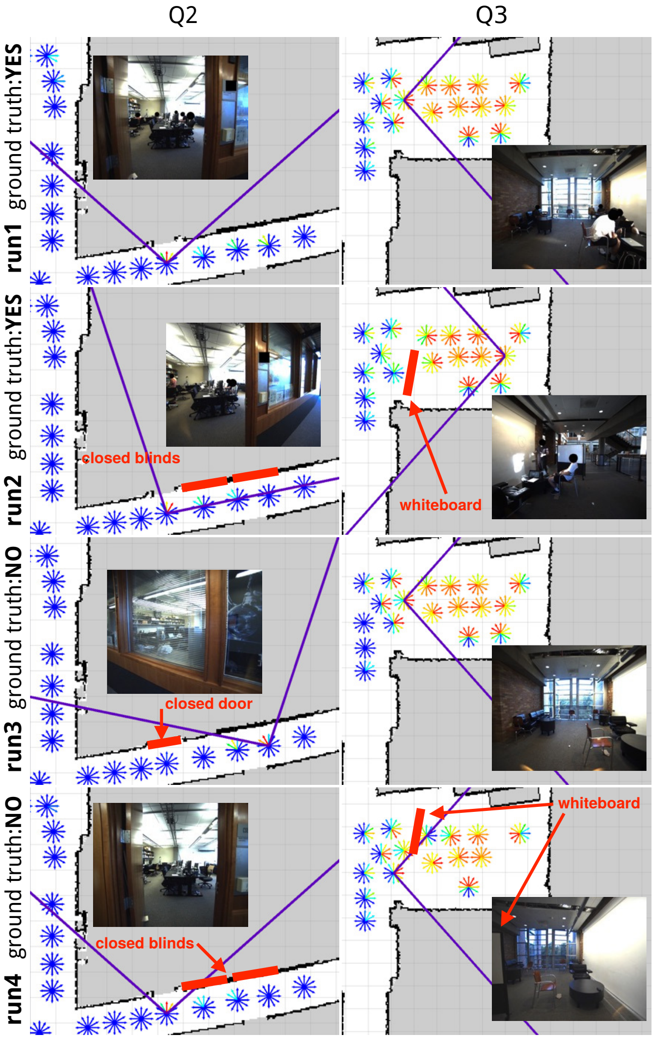


Fig. 4. Viewpoint estimation details for Q2 and Q3 in Experiment I. The evaluated $P(\mathcal{I} = 1|v, q; W, M^{3D})$ for a certain state of M^{3D} and each viewpoint is displayed using colored arrows on top of the 2D map. Warm colors (red) indicate higher probability. The camera field of view of the viewpoint currently considered best is indicated with purple lines and the image captured from this viewpoint is shown. The dynamic changes to the environment that influenced viewpoint estimation are highlighted in red. Gray areas were marked as inaccessible, but could be observed.

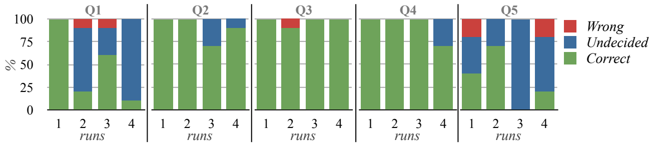


Fig. 5. Statistics of answers generated by the user study participants based on images captured by the robot.

response rates are due to the limitations of the sensors or the encountered situation rather than a limitation of the algorithm (e.g. target locations blocked by closed doors or difficult illumination conditions). For example, in the 4th run for Q1 and 3rd run for Q5, participants said they were undecided if the person or object is present because the door were closed, not because the robot provided insufficient information. In other words, if the users were to answer Q1 or Q5 in this situation by going to the target location, they would reach the same conclusion. In the remaining runs of Q5, the incorrect and undecided answers were due to insufficient illumination

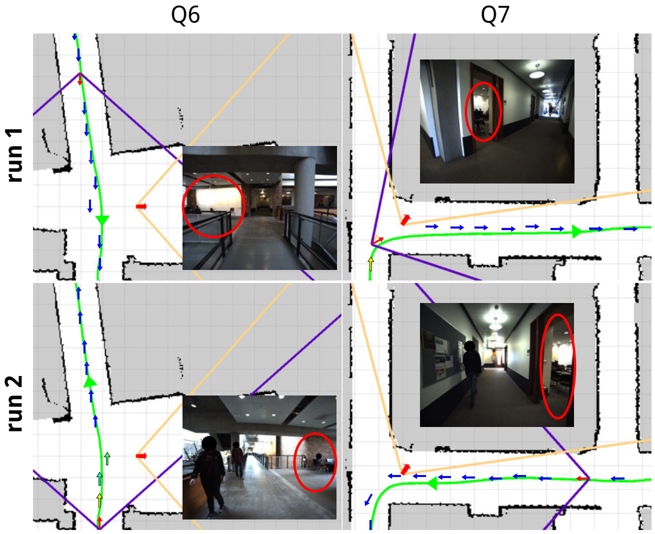


Fig. 6. Results of Experiment II for a sparse set of previously visited viewpoints. The path taken by the robot is shown with a green line (with direction) and visited viewpoints are marked as small arrows along the path. The color of the arrows has the same semantics as in Fig. 4. The camera field of view for the selected previously visited viewpoint is shown with purple lines and the image captured from this viewpoint is provided. The target area is marked with a red ellipse in each image. For reference, an estimate of a viewpoint that would have been selected if the robot were to navigate back to the scene to capture the requested information is shown with the orange lines.

and small, reduced-quality image. This problem could be mitigated by allowing the robot to navigate into the room to obtain a better viewpoint, post-processing images to enhance color contrast, or allowing participants to zoom in on parts of the image to obtain the answer.

C. Experiment II: Information Retrieval

Next, we considered the retrospective information retrieval scenario in which the viewpoint estimation is performed on previously recorded sensory data. This captures scenarios that involve questions concerning the past ("Was Mike Chung in the robotics lab?"), where the system attempts to provide a response based on incidental visits to a place while performing other tasks involving navigation (e.g. patrolling or delivery). Storing and searching all images captured by the robot could be problematic. However, since we considered only a discrete set of possibly relevant viewpoints originating from places in our topological map, we could easily maintain the most recent image for each previously *visited* viewpoint. Then, the viewpoint estimation algorithm was applied with the latest available 3D occupancy map, within the constrained search space of the *visited* viewpoints.

We considered four such cases. In the first two, the robot was navigating near the breakout area in opposite directions as shown in Fig. 6 (left). Later, the system was asked the question Q6: "Was the breakout area occupied?". In the latter two cases, the robot was navigating near the conference room as shown in Fig. 6 (right) and the system was asked the question Q7: "Was the conference room occupied?".

The retrieved images are shown in Fig. 6. We observe that the viewpoint estimation algorithm produces appropriate responses given the rather sparse set of *visited* viewpoints.

In response to Q6, the robot needs to capture the breakout area from a set of candidate viewpoints that are tangential to the area (i.e. the robot passed by the breakout area without turning towards it). We see that the algorithm selects viewpoints that are further away from the target, yet not too far to be useful. This way, despite the viewpoints being tangential to the target, the target is captured within the camera's field of view. In first run for Q6, the robot exploits the fact that the hand rail only partially obscures the view, making a further viewpoint that captures a larger part of the target area more optimal. In response to Q7, the robot is able to capture a part of the target conference room by choosing viewpoints near two different doors to the room in the two different runs in opposite directions.

VI. Discussion and Conclusion

We presented a unique framework for answering natural language questions about the state of a dynamic environment co-inhabited by humans and mobile robots. Overall, our findings indicate that the framework and its implementation address the problem well. Our previous work motivated the usefulness of this capability from the end-user perspective [1], while this paper demonstrates the feasibility of end-to-end implementation on an autonomous mobile robot. There are nonetheless several assumptions made in scoping our problem and some limitations to the proposed approach.

First, we assume that the user's question mentions a single target location that can be feasibly captured from a single viewpoint. Such question as "Is Mike Chung in this building?" requires object search and therefore is out of scope for our framework. However, one can imagine a search method that embeds our approach for checking information at multiple target locations, within a larger planning framework. Similarly, questions that mention multiple target locations, such as "Is Mike Chung in the robotics lab or in his office?" are not handled; however, this task could simply be considered as two separate requests.

Second, in our current framework, we do not utilize the visual information for the purpose of viewpoint selection. Although, we choose to leave the extraction of answers from images to the end user, the system could still rely on visual information to assist the semantic annotations in viewpoint selection (e.g. prefer viewpoints containing the a chair in the area where a person might be present). This is an interesting area of investigation for our future work.

References

[1] M. Chung, A. Pronobis, M. Cakmak, D. Fox, and R. P. Rao, "Designing information gathering robots for human-populated environments," in *IROS*, 2015.

[2] M. C. Potter, B. Wyble, C. E. Hagmann, and E. S. McCourt, "Detecting meaning in rsvp at 13 ms per picture," *Attention, Perception, & Psychophysics*, 2014.

[3] M. Veloso, J. Biswas, B. Coltin, S. Rosenthal, T. Kollar, C. Mericli, M. Samadi, S. Brandao, and R. Ventura, "Cobots: Collaborative robots servicing multi-floor buildings," in *IROS*, 2012.

[4] "Aethon TUG," http://www.aethon.com/tug/.

[5] "Savioke SaviOne," http://www.savioke.com.

[6] "Vecna QC Bot," http://www.vecna.com/product/qc-bot-base-model/.

[7] G. Kim, W. Chung, K.-R. Kim, M. Kim, S. Han, and R. H. Shinn, "The autonomous tour-guide robot jinny," in *IROS*, 2004.

[8] R. Philippsen and R. Siegwart, "Smooth and efficient obstacle avoidance for a tour guide robot," in *ICRA*, 2003.

[9] S. Thrun, M. Bennewitz, W. Burgard, A. Cremers, F. Dellaert, D. Fox, D. Hahnel, C. Rosenberg, N. Roy, J. Schulte, *et al.*, "MINERVA: A second-generation museum tour-guide robot," in *ICRA*, 1999.

[10] A. Ohya, Y. Nagumo, and Y. Gibo, "Intelligent escort robot moving together with human-methods for human position recognition," in *Int. Conference on Soft Computing and Intelligent Systems*, 2002.

[11] M. K. Lee, S. Kiesler, and J. Forlizzi, "Receptionist or information kiosk: how do people talk with a robot?" in *ACM Conference on Computer Supported Cooperative Work*, 2010.

[12] T. Kanda, M. Shiomi, Z. Miyashita, H. Ishiguro, and N. Hagita, "An affective guide robot in a shopping mall," in *HRI*, 2009.

[13] J. Park and G. J. Kim, "Robots with projectors: an alternative to anthropomorphic hri," in *HRI*, 2009.

[14] M. Lorbach, S. Hofer, and O. Brock, "Prior-assisted propagation of spatial information for object search," in *IROS*, 2014.

[15] A. Aydemir, A. Pronobis, M. Gobelbecker, and P. Jensfelt, "Active visual object search in unknown environments using uncertain semantics," *IEEE Transactions on Robotics*, 2013.

[16] M. Samadi, T. Kollar, and M. M. Veloso, "Using the web to interactively learn to find objects." in *AAAI*, 2012.

[17] L. Kunze, M. Beetz, M. Saito, H. Azuma, K. Okada, and M. Inaba, "Searching objects in large-scale indoor environments: A decision-theoretic approach," in *ICRA*, 2012.

[18] D. Joho and W. Burgard, "Searching for objects: Combining multiple cues to object locations using a maximum entropy model," in *ICRA*, 2010.

[19] T. Kollar and N. Roy, "Utilizing object-object and object-scene context when planning to find things," in *ICRA*, 2009.

[20] S. Tellex, T. Kollar, S. Dickerson, M. R. Walter, A. G. Banerjee, S. J. Teller, and N. Roy, "Understanding natural language commands for robotic navigation and mobile manipulation." in *AAAI*, 2011.

[21] C. Matuszek, D. Fox, and K. Koscher, "Following directions using statistical machine translation," in *HRI*, 2010.

[22] J. Fasola and M. J. Mataric, "Interpreting instruction sequences in spatial language discourse with pragmatics towards natural human-robot interaction," in *ICRA*, 2014.

[23] V. Perera and M. Veloso, "Handling complex commands as service robot task requests," in *IJCAI*, 2015.

[24] A. Jacoff, "Search and rescue robotics," in *Springer Handbook of Robotics*, 2008.

[25] W. F. Truszkowski, M. G. Hinchey, J. L. Rash, and C. A. Rouff, "Autonomous and autonomic systems: A paradigm for future space exploration missions," *Trans. on Systems, Man, and Cybernetics*, 2006.

[26] J. Casper and R. Murphy, "Human-robot interactions during the robot-assisted urban search and rescue response at the world trade center," *Trans. on Systems, Man, and Cybernetics*, 2003.

[27] R. Bachmayer, S. Humphris, D. Fornari, C. Van Dover, J. Howland, A. Bowen, R. Elder, T. Crook, D. Gleason, W. Sellers, *et al.*, "Oceanographic research using remotely operated underwater robotic vehicles: Exploration of hydrothermal vent sites on the mid-atlantic ridge at 37 north 32 west," *Marine Technology Society Journal*, 1998.

[28] G. A. Hollinger and G. S. Sukhatme, "Sampling-based robotic information gathering algorithms," *IJRR*, 2014.

[29] J. Van Den Berg, S. Patil, and R. Alterovitz, "Motion planning under uncertainty using iterative local optimization in belief space," *IJRR*, 2012.

[30] J. Velez, G. Hemann, A. S. Huang, I. Posner, and N. Roy, "Planning to perceive: Exploiting mobility for robust object detection." in *International Conference on Automated Planning and Scheduling*, 2011.

[31] T. H. Chung, G. A. Hollinger, and V. Isler, "Search and pursuit-evasion in mobile robotics," *Autonomous Robots*, 2011.

[32] G. Grisetti, C. Stachniss, and W. Burgard, "Improved techniques for grid mapping with Rao-Blackwellized particle filters," *IEEE Transactions on Robotics*, 2007.

[33] A. Pronobis and P. Jensfelt, "Large-scale semantic mapping and reasoning with heterogeneous modalities," in *ICRA*, 2012.

[34] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: An efficient probabilistic 3D mapping framework based on octrees," *Autonomous Robots*, 2013.

[35] E. Marder-Eppstein, E. Berger, T. Foote, B. Gerkey, and K. Konolige, "The office marathon," in *ICRA*, 2010.

[36] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky, "The Stanford CoreNLP natural language processing toolkit," in *ACL: System Demonstrations*, 2014.