Project Proposal: Autonomous Search and Identification

Joe Howarth, Matthew Lee, and Kostas Tsiampouris

COMP50 - Autonomous Intelligent Robots

March 16, 2018

I. Introduction

In this project we aim to enable a TurtleBot to search a multi-room space for a specified object and then relay its position back to the operator. The location and object could be anything; however, we will focus on everyday objects in the upper floor of Halligan. This task represents a simplification of relevant tasks for autonomous robots such as disaster search and rescue, security patrolling, and industrial inspections. The autonomous search could itself be a full application for the robot or later work could build off this to achieve more complicated tasks such as object retrieval.

The main modules that will implement this aim are:

A. Planning a search pattern
B. Detecting an object within an image
C. Integrating high-level movement and camera tasks with the TurtleBot through the Robot Operating System (ROS) library

We will start by integrating an object detection model with the TurtleBot and attempting to recognize objects. We will then progressively increase the difficulty of the task by introducing movement and obstacles that the robot must navigate around to find the object. This strategy should hopefully lead to steady progress instead of building all the components out at once and testing at the end.

II. Related Works

Many past research projects have investigated algorithms and methods closely related to the modules presented for this project. Some selected related works have been presented below:

a. Image Recognition

In computer vision, assigning bounding boxes to objects within an image is known as object detection. This problem is significantly more difficult than image classification due to the possible presence of multiple objects within the image as well as the need to compute object locations and box offsets for each candidate. The current state of the art image classifiers using Convolutional Neural Network architectures such as InceptionNet[1], ResNet[2] or VGG[3] net achieve super-human accuracies on the ImageNet[4] dataset.

Due to this success, many object detection approaches focus on generating bounding box proposals, then feeding these sections of the images into a pre-trained image classifier. R-CNN[5],

---

[1] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. "Going deeper with convolutions." CoRR, abs/1409.4842, 2014.

[2] He, K., Zhang,X. ,Ren ,S. ,Sun ,J. :Deep residual learning for image recognition .In :CVPR. (2016)

[3] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *arXiv preprint arXiv:1409.1556* (2014).

[4] Deng, Jia, et al. "Imagenet: A large-scale hierarchical image database." *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009

[5] Girshick, Ross, et al. "Rich feature hierarchies for accurate object detection and semantic segmentation." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014.

Fast R-CNN[6] and Faster R-CNN[7] all use this approach and achieve high accuracy on large object detection datasets such as COCO or Pascal VOC, but suffer from low frames-per-second (FPS). This makes these methods problematic for real time detection such as autonomous search and localization.

Two promising alternatives are the YOLO[8] (You Only Look Once) architecture (Redmon et al) and the SSD[9] (Single Shot Detector) (Liu et al). Both only make one pass through a single convolutional network and are thus much faster. SSD can perform at state of the art accuracies while maintaining 60FPS on a single Titan X GPU.

b. Searching Algorithm

The problem of optimally searching a given room, modeled as a polygon with or without holes, is known within the field of computational geometry as the "watchman route problem" (WRP). It is a complication of the more famous "art gallery problem" or the "museum problem" that asks about the minimum number of stationary guards needed in order to observe an entire art gallery at once. Unlike the museum problem where observers are stationary, the watchman route problem asks about the shortest route required by a single observer to traverse through a polygon ensuring that every point within the polygon is visible from at least one point along the route.

This problem has been explored extensively and has been proven to be NP-hard for polygons with holes but solvable in polynomial-time for simpler polygons. In 1987, Chin and Ntafos from the University of Texas at Dallas proposed the first solution to the WRP given certain constraints; they proposed a O(n log log n) algorithm given a rectilinear polygon, or a polygon with linear edges and vertices at 90 or 270 degrees.[10] Many paper attempts to generalize this algorithm failed providing flawed algorithms with time-complexities of $O(n^3)$ and $O(n^2)$.[11] However, in 2003, Dror et al. successfully generalized the algorithm providing an optimal algorithm with a runtime of $O(n^3 \log n)$ for the fixed watchman route problem, where the starting is specified, and $O(n^4 \log n)$ for the floating watchman route problem, where the starting point *s* can be arbitrarily chosen.[12] In 2008 as part of the International Workshop on Robot Vision, Li and Klette furthered this research problem by utilizing a rubber band algorithm to approximate an improved bound of $\kappa(\varepsilon) \times O(kn)$ where k represents the number of essential cuts and ε is an accuracy constant.

The watchman route problem has been analyzed by many researcher and as the progression above shows, many optimal solution to the problem have been presented, rejected, and improved upon. In the context of the proposed project, an optimal solution to the WRP can

---

[6] Girshick, Ross. "Fast r-cnn." *arXiv preprint arXiv:1504.08083*(2015).

[7] Ren, Shaoqing, et al. "Faster r-cnn: Towards real-time object detection with region proposal networks." Advances in neural information processing systems. 2015.

[8] Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

[9] Liu, Wei, et al. "Ssd: Single shot multibox detector." *European conference on computer vision*. Springer, Cham, 2016.

[10] W. Chin and S. Ntafos. Optimum watchman routes. Information Processing Letters, 28:39–44, 1998.

[11] Dror, M., Efrat, A., Lubiw, A., Mitchell, J.S.B. Touring a sequence of polygons. In: Proceedings of the 35th ACM Symposium on Theory of Computing, pp. 473–482 (2003)

[12] F. Li, R. Klette, "An approximate algorithm for solving the watchman route problem", Proc. 2nd Int. Conf. Robot Vis., pp. 189-206, 2008.

be utilized to ensure that every point within the desired building, which can be represented as a set of edges with holes, is searched in the least amount of time.

c. Multi-robot search/communication

An alternative to the method above that will allow for the use of multiple robots utilizes the Lévy flight mechanism and the artificial potential method as proposed by Sutantyo et al. in their paper, "Multi-robot searching algorithm using Levy flight and artificial potential field." [13] Lévy flight is just an instance of a Brownian random walk, using Lévy probability distributions. The Brownian walk is very similar to the one that animals use to find food, when having no prior experience with their environment. Lévy probability distribution is very effective when targets are randomly distributed over large spaces, and its equation is shown below:

$$P_{\alpha,\gamma}(1) = \frac{1}{\pi}\int_{-\infty}^{\infty} e^{-\gamma q^{\alpha}} cos(ql)dq$$

The artificial potential method is another topic simulated from nature -- physics, in this case. In essence, when multiple robots are searching at the same time, each of them creates a potential field, which makes it less desirable for other robots to come nearby. This is similar with the way that positive charges act through space. It ensures that robots may cover the largest area possible in the least amount of time. It is expressed by the following gradient of potential, a Repulsion Force":

$$\underline{F}_{rep} = -\underline{\nabla} U_{rep}(q) = \begin{cases} k_{rep}\left(\frac{1}{\rho(q)} - \frac{1}{\rho_0}\right)\frac{\underline{q}-\underline{q_{neighbor}}}{\rho(q)}\frac{1}{\rho^2(q)} \text{if} \rho(q) \geq \rho_0 \\ 0 \text{if} \rho(q) < \rho_0 \end{cases}$$

With these two methods, we could make a multi-robot system efficient, by making each gradually and efficiently exploring their environments, without coming too close, causing overlap.

III. Technical Approach

a. Overall Project Outline

The goal of the project is to develop a robot and the associated algorithm to search through Halligan Hall and find a desired object. It will be implemented using a TurtleBot, running with the Robot Operating System library.[14,15] In addition to the base module of the TurtleBot, a camera will be mounted approximately 3 feet above the robot base to provide a better vantage point and a clearly, unobstructed view of nearby objects.

Overall, the project will work as follows: a map of the surrounding area and the current position will be given to the program, which will determine the points needed to have a full look of the current room. A traveling salesman algorithm will then follow, which will find the fastest

[13] Donny K. Sutantyo, Serge Kernbach, Valentin A. Nepomnyashchikh, Paul Levi, Multi-robot searching algorithm using Levy flight and artificial potential field, CoRR, abs/1108.5624, 2011.

[14] https://www.turtlebot.com/

[15] http://www.ros.org/

way of moving between all these points. The TurtleBot will then proceed to move to each one of these locations and take pictures upon reaching the destination. The pictures will be simultaneously analyzed by computer vision software, which will give an estimation of the probability that the object is found. If this probability is above a certain threshold, the robot will navigate off the planned traveling salesman route to more closely analyze the suspected object. If, upon further inspection, the item is confirmed, the program will log the image and alert the user.

b. Image Recognition

The SSD architecture discussed in Section II.a provides the best combination of accurate object detection with performance considering the TurtleBot's currently limited computational power. While building and training a model based off this paper is possible, the TensorFlow Object Detection API[16] provides a polished interface to many models. The SSD and YOLO papers appear to be in beta at the moment; however, the Faster R-CNN may work well enough. This API outputs a bounding box and class confidence for objects in the image. The pertained models already support the classes of the COCO[17] dataset, so we may begin with an included object like an umbrella.

The Object Detection module will consist of receiving an image from the mounted camera, passing it through the TensorFlow API, then filtering out the found objects for the object in question above a certain confidence threshold. The module would then communicate to the higher level planning modules whether the object was in the current field-of-view and could proceed accordingly.

The module would be written in Python3 with Numpy and OpenCV3.0 for the data manipulation.

c. Movement Algorithm

While the watchman route problem discussed in Section II.b would provide an ideal solution to view every point within a robot's internal map, this method does not work in reality as the image processing techniques described above cannot occur fast enough to work on a live feed. Thus, we propose an alternative solution that follows a standard route but can enter a sidetracked mode as needed.

Standard Route:

The standard route for this robot consists of first solving the art gallery problem followed by the traveling salesman problem given the solution to the first problem.

The art gallery problem asks about the minimum number of guards required to observe an entire gallery; mathematically, this problem can be restated as follows: given a set of points S, what is the minimum size of S such that there exists, between every point and a member of S, some line segment that does not intersect an edge of the polygon. For this project, the algorithm described by Couto, de Rezende, and de Souza will be implemented and used to precompute a solution to the art gallery problem.[18] The algorithm described by Couto et al. solves the problem

---

[16] https://github.com/tensorflow/models/tree/master/research/object_detection

[17] Lin, Tsung-Yi, et al. "Microsoft coco: Common objects in context." *European conference on computer vision*. Springer, Cham, 2014.

[18] Couto, M.C., de Rezende, P.J., de Souza, C.C.: An exact algorithm for minimizing vertex guards on art galleries. International Transactions in Operational Research 18(4), 425–448 (2011)

for polygons without holes, which when considering Halligan Hall, is a valid assumption. However, for more complex buildings and rooms, this may not be a valid assumption to make.

After solving the art gallery problem, there will be a set points that provide coverage of the entire map. Given the starting point of the robot and the set of points, the traveling salesman problem can be solved. The traveling salesman problem asks about the shortest possible route such that each point in the set is visited once. While the solution to the traveling salesman problem is ideal, any reasonable solution is acceptable as the rooms in Halligan are fairly simple and not many vertices will be generated in the first place.

Thus, the robot now has a baseline route to follow that provides the robot with full coverage of the room.

Sidetracked Mode:

A sidetracked mode helps to prevent a situation where the traveling salesman route encourages the robot to go right despite something resembling the desired object seen to the left. The sidetracked mode will trigger and the robot will remain in this mode while the image recognition model suspects that the desired object is seen.

In the sidetracked mode, the robot will wander off the standard route to more closely explore items off the path. To do so, the robot will travel one meter at a time in the direction of the seen object to get a better view of the suspected object. If the object is determined to not be the desired object, the robot will return to the standard route and continue its path.

IV. Evaluation Plan

The proposed project is designed in a way such that it can be developed incrementally. Success will be defined as the robot being able to achieve the goals of each of the incremental stages at a rate greater than 80%. The incremental stages for success are detailed below:

The robot is able to locate the desired object...

1. That is in its direct line of sight given a map of the room
2. In an empty room given a map of the room
3. In a room with minor obstacles (boxes on ground that prevent robot movement but not the object detecting camera) given a map
4. In a room with obstacles blocking its direct line of site given a map
5. Given a map with multiple rooms and obstacles

Multiple robots are able to search together to find an object...

6. In an empty room
7. In a room with many obstacles
8. Given a map with multiple rooms and obstacles

In addition to above, a web interface could be built that allows a user to see where the robots are, to track what regions have been searched, and to view a live video feed to confirm/reject objects as being detected.

Many of the stages above “stretch-goals” that may or may not be achieved. However, they provide an challenge that will be attempted as time permits.