Contract No:

This document was prepared in conjunction with work accomplished under Contract No. DE-AC09-08SR22470 with the U.S. Department of Energy (DOE) Office of Environmental Management (EM).

Disclaimer:

SRNL-STI-2020-00359
Revision 0

Gamma-Ray Raster Imaging with Robotic Data Collection

T. Aucott *, W. Wells*, M. Siddiqi**

*Savannah River National Laboratory (SRNL),
Aiken, SC, USA, william.wells@srnl.doe.gov
**University of Florida, Gainesville, Florida, USA, mhsiddiqi@ufl.edu

ABSTRACT

In order to create gamma-ray images, some form of collimation is required. The foremost imaging techniques either require physical collimation or restrictive algorithms. Physically collimated approaches result in a limited field of view. An alternative imaging capability for characterizing and imaging radioactive materials in situ has been developed. This approach uses a robotic-mounted gamma-ray detector which can move around an area of interest, sampling the space at an extremely high frequency. The detector was calibrated in three dimensions as a function of energy, distance, and angle. A Bayesian particle filter was implemented to localize and quantify a radioactive source in a search arena. An informed guidance algorithm was also developed to guide the robot's search path to obtain more accurate information on the source. The use of a robotic mount allows data collection for extended periods of time unattended, and it eliminates uncertainties in positioning typically introduced by personnel.

Keywords: gamma-ray, raster, imaging, bayesian, robotic

1 INTRODUCTION

Holdup characterization involves determining mass, shape, location, and composition of radioactive isotopes. gamma-ray assays and images are a key tool for holdup characterization in a nuclear facility. Images can be used to create radiation maps, which can be used for establishing procedures, assist decontamination and aid radiological control. Heavy collimators are typically used to perform gamma-ray Assays. The collimated detectors are resource intensive requiring hours of continuous operation. Operators may introduce uncertainty into the process by providing imprecise prior data. Alleviating these limitations this work utilizes a robotic acquisition system along with a small gamma-ray detector to sample an area at high frequency in order to create an image. This approach is:

- Lightweight: requiring little to no lead shielding
- Autonomous: requiring minimal operator time and input
- Precise: relying on camera, LIDAR, and software to map out the space.

This work integrates state-of-the-art robotics developments to improve the acquisition of gamma-ray data. An informed guidance algorithm is used to gather the most information about the radiation distribution while operating under the set of constraints given by the dynamics of the robot and other requirements such as minimizing the overall mission time.

There are existing approaches which use directional sensors [1, 2] to gain further information about where a radiation source might be. In this work, an omnidirectional gamma sensor is utilized, meaning that the origination of the radiation when measuring at a location is not known. This greatly lowers the cost of the equipment needed but increases the complexity of localization. The use of Gaussian process regression (GPR) and an associated utility function has been proposed by several authors [3, 4]. This project also explored the use of a Bayesian particle filter [5, 6] to localize and quantify the source.

2 METHOD

Experiments were carried out on a Turtlebot3 Waffle from ROBOTIS. The Turtlebot3 is equipped with an OpenCR1.0 control board with a 9-axis Inertial Measurement Unit (IMU), an Intel Joule 570x (single board computer), 2 Dynamixel XM430-W210-T networked motor actuators with contactless absolute encoders, an LDS-01 2D real time light detection and ranging (LiDAR) system with 360-degree sensing, and an Intel RealSense R200 3D camera. (shown in Figure 1).

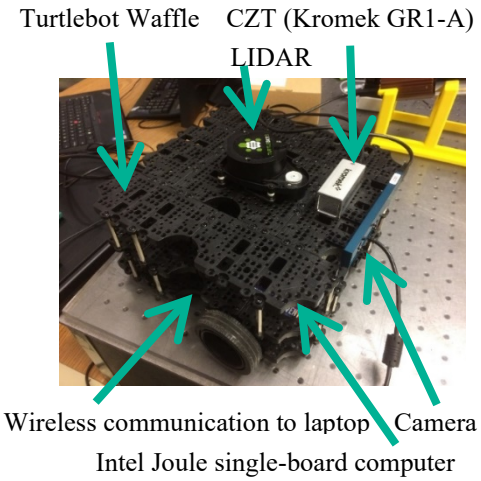

Figure 1: The Turtlebot3 Waffle robot, with important components and sensors

2.1 Robot Setup

A Robot Operating System (ROS) compatible platform was chosen for rapid deployment and sensor testing. ROS Kinetic running on Ubuntu Xenial and Ubuntu Core on the controlling station and the Intel Joule respectively are the platforms that were selected for this work. A user interface was developed to give the user robotic acquisition system start and stop control as well as parameter adjustment capabilities.

The onboard USB hub was upgraded to a hub with additional USB ports for attaching the gamma-ray detector and peripherals that aided in initial software configuration as well as additional programming. A 3 cell 11.1V Li-Po battery provides power to the Turtlebot3. The sensors and actuators that are included with the Turtlebot3 are accessed/controlled through ROS installed on the Intel Joule. A wireless card is included on the Intel Joule, which enables data to be received/transmitted between the Turtlebot3 and a controlling station. The Intel RealSense camera was lowered to the middle waffle platform to properly visualize obstacles in the test area including chairs. The gamma-ray detector is mounted on the middle waffle platform facing the opposite direction of the Intel RealSense camera.

2.2 Detector Setup

A gamma-ray detection system, a cadmium zinc telluride (CZT) semiconductor detector from Kromek, is mounted to an autonomous robotic platform. The CZT provides excellent energy resolution in an extremely small package, keeping the payload light. The detector is controlled by an Intel Joule single-board computer, which in turn talks wirelessly to a laptop.

The Kromek GR1-A splits the detected spectrum into 4096 energy band channels. The detector was calibrated in three dimensions as a function of energy, distance, and angle using a Ho-166m calibration source. Ho-166m is a radioactive source that produces a multitude of gamma rays at a wide range of energies. The source was placed on a grid around the detector. For each position on the grid the distance (r) and angle (θ) to the detector are known. The entire range of energies (E) is detected at each position. The detector efficiency (ε) is a function of those three terms as shown in equation 1. Constants C_1-C_4 were fit by the programming language R. C_0 was set by normalizing the fall-off distance.

$$\varepsilon = \frac{C_0}{r^2}\exp\left\{C_1 + C_2\ln\left(\frac{E}{E_0}\right) + C_3\ln^2\left(\frac{E}{E_0}\right) + C_4\left(\frac{\theta}{\theta_0}\right)^3\right\} \quad (1)$$

2.3 Data Acquisition

Gamma-ray spectra are saved along with LIDAR scans and positioning information. The Kromek GR1-A included a device driver and a C++ accessible library.

A user interface was developed on the controlling station using QT and C++ shown in Figure 2. A clickable executable was created to start the user interface. The executable is a shell script. The user interface allows starting and stopping of the autonomous robotic system. Modifiable system parameters include radiation color coded threshold values (T_1, T_2, T_3), sampling time, and the size of squares used to depict the detected radiation readings overlaid on the Simultaneous Localization and Mapping (SLAM) generated real time area map.

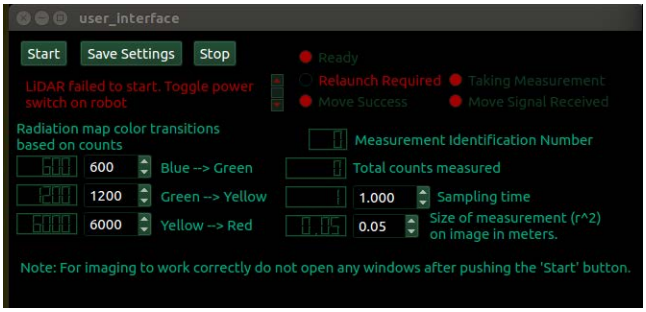

Figure 2: User Interface for starting and stopping robotic system as well as adjusting system parameters

An auto start shell script on the Intel Joule starts ROS on operating system startup. Upon startup connections to sensors and actuators are established as well as AMCL (Adaptive Monte Carlo Localization), a probabilistic localization system that performs SLAM for an autonomous vehicle restricted to movement in 2D space. After modification of a parameter to use the real-time generated map, AMCL performs actual *simultaneous* localization and

mapping; the default configuration requires one to perform mapping of a space prior to utilizing localization.

Once the user presses the start button on the user interface, the robot simultaneously starts recording measurements every 1 second from the CZT detector as well as enables the guidance algorithm. The total count (C_t) value from the CZT detector at time t is used along with the radiation color coded threshold values to add a radiation marker to the SLAM generated real time map of the area. Table 1 displays the radiation marker color coding scheme.

Color	**C_t**
blue	$[0, T_1)$
green	$[T_1, T_2)$
yellow	$[T_2, T_3)$
red	$[T_3, \infty)$

Table 1: Radiation marker color coding scheme.

Pressing the start button executes a shell script on the controlling station that starts RVIZ (a ROS visualization tool), starts recording data logs, and starts taking screenshots of the controlling station display every 130 seconds using FFmpeg.

2.4 Guidance Algorithm

The guidance algorithm, referred to as Adaptive Walk, uses an informed pseudo random walk. The Cartesian axes are oriented with the +y axis extending forward from the robot's initial position and the +x axis extending to the right side of the robot when the system starts. The position the robot occupies at startup is the map origin. Target $[x_{t+1}, y_{t+1}]$, actual $[x_t, y_t]$, and previous $[x_{t-1}, y_{t-1}]$ positions are tracked. All initial positions are set to $[x_0, y_0] = [0, 0]$. Actual position $[x_t, y_t]$ is updated from the odometry measurement. Target position is updated as follows:

$$[x_{t+1}, y_{t+1}] = [n \cos \theta_{t+1}, n \sin \theta_{t+1}] \quad (2)$$

θ_{t+1} is the target directional angle from the robot's actual position to the target position oriented with respect to the origin at time step t+1. $\theta_{t+1} \epsilon S = \{\pi/2, \pi/4, 0, -\pi/4, -\pi/2, -3\pi/4, \pi, 3\pi/4\}$. In the experiments n = 0.3048 (metric equivalent of 1 foot) was used. The value $p_{min} = 0.0762$ is used to determine if the navigation layer path planner is able to find a path to the target position. While updating the target position, the difference between the position from the odometry (IMU and wheel encoders) and the previous actual position is computed,

$$[\Delta x, \Delta y] = [x_t - x_{t-1}, y_t - y_{t-1}]. \quad (3)$$

θ_{t+1} is updated in accordance with the following function:

$$\theta_1 = \text{Random } s \epsilon S$$
$$\theta_{t+1} = \text{Random } s \epsilon S, \text{ if } [|\Delta x|, |\Delta y|] \geq [p_{min}, p_{min}] \Lambda C_t \leq C_{t-1}$$
$$\theta_{t+1} = \theta_t, \text{ if } [|\Delta x|, |\Delta y|] \geq [p_{min}, p_{min}] \Lambda C_t > C_{t-1}$$
$$\theta_{t+1} = \theta_t - \frac{\pi}{4}; \theta_{t+1} \epsilon S, \text{ if } [|\Delta x|, |\Delta y|] < [p_{min}, p_{min}] \quad (4)$$

The initial direction of movement (θ_1) is chosen randomly from the set S. If the navigation layer path planner is unable to find a path to the target position, the next sequential $s \epsilon S$ is chosen as the new target angle. If successive detector count readings are increasing, the direction of targeted motion remains the same, otherwise a random $s \epsilon S$ is chosen as the new target angle.

Prior to sending commanded $[x_{t+1}, y_{t+1}]$ values to the robot $[x_{t-1}, y_{t-1}]$ is updated as $[x_t, y_t]$.

2.5 Experiment Environment

The test area consisted of an enclosure created by 2 connected North State Industries Superyards in a random shape. Sealed button sources were placed in this test enclosure along with the robotic acquisition system.

Figure 3: Testing environment with 2 sealed button sources.

3 RESULTS

The CZT detector was tested with a Cs-137 10 µCi source, placed in a large paint can-sized container for improved response from the robot's LIDAR to help prevent collision. The can was placed at a location of (1.58 m, -1.1 m) relative to the starting point of the robot, which is treated as the origin (0 m, 0 m). A few main points about the test should be noted:

- The robot covered only half of the total quartered-off area in which it could survey,
- The robot, for most of the test, never moved behind the source (i.e. it was always to one side of the source

during the test; a full 360-degree coverage was not obtained, only 180 degrees),

- The source was located at the center of the test area.

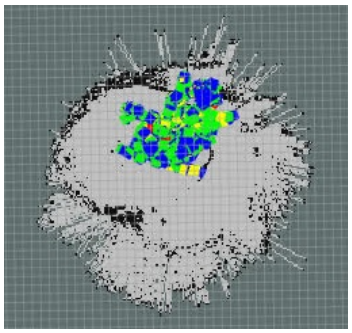

Figure 4: Raster image with a sole source

3.1 Data Post Processing

A Bayesian particle filter was implemented in MATLAB on the logged data that was converted from ROS .bag files to .csv files. Position, orientation, and radiation count data were pulled from the .csv files and processed as such: position data on the millisecond level was averaged to generate an average position per second, angle of orientation was converted from standard quaternion to angle (in degrees), and radiation counts were arranged into a set of spectra with 4096 channels.

The detector calibration was used as the input to the Bayesian particle filter. A particle filter is a form of maximum likelihood estimation (MLE) which utilizes a large, finite number of hypotheses to approximate an unknown source distribution. At the core of the algorithm is Bayes' theorem, using an array of measurement and hypothesis vectors. For some set of $k = 1{:}K$ measurements being constantly updated, each measurement $\boldsymbol{M}_k = (x_k, y_k, \theta_k, \lambda_k)$, and a set of $n = 1{:}N$ hypotheses, each hypothesis $\boldsymbol{H}_n = (\hat{x}_n, \hat{y}_n, \hat{A}_n, w_n)$, Bayes' theorem can be applied to derive the following relation:

$$P(\boldsymbol{H}_{1:N}|\boldsymbol{M}_{1:K}) \propto \mathcal{L}(\mathbf{M}_K|\mathbf{H}_{1:N})P(\boldsymbol{H}_{1:N}|\boldsymbol{M}_{1:K-1}) \quad (5)$$

Which can be simply rendered as: posterior ∝ likelihood · prior.

The Bayesian particle filter algorithm ran for a total of 140 times on a data set of the Cs-137 10 μCi source test, each time with a different distribution of particles and positions selected randomly within the test area. The predicted x and y-coordinates and predicted activity, A, of each test was recorded. Because of the low efficiency of the CZT detector, the entire gamma-ray spectrum (not just the unscattered photopeak) was used for the analysis. As a result, although activity is predicted, it is dominated by down-scatter in the environment and did not accurately determine the source activity. The distribution of particles with position and activity is shown in Figure 5. The mean (μ ± 1σ) of the x and y-coordinates are: 1.76 ± 0.437 meters and -0.809 ± 0.851 meters. Inside the circled area the mean is: [1.86 ± 0.210, -1.19 ± 0.296] meters.

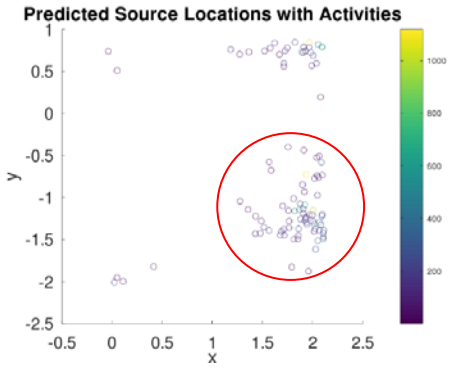

Figure 5: Predicted distribution of likely source locations

REFERENCES

[1] K. Vetter, R. Barnowksi, A. Haefner, T. H. Y. Joshi, R. Pavlovsky, and B. J. Quiter, "A Gamma-Ray imaging for nuclear security and safety : Towards 3-D gamma-ray vision," Nucl. Inst. Methods Phys. Res. A, vol. 878, pp. 159–168, 2018.

[2] M. Lee, M. Hanczor, J. Chu, Z. He, N. Michael, and R. Whittaker, "3-D Volumetric Gamma-ray Imaging and Source Localization with a Mobile Robot," Waste Management Symposia 2018.

[3] R. Marchant and F. Ramos, "Bayesian Optimization for informative continuous path planning," Proc. - IEEE Int. Conf. Robot. Autom., pp. 6136–6143, 2014.

[4] G. Hitz, A. Gotovos, F. Pomerleau, M. É. Garneau, C. Pradalier, A. Krause, and R. Y. Siegwart, "Fully autonomous focused exploration for robotic environmental monitoring," Proc. - IEEE Int. Conf. Robot. Autom., pp. 2658–2664, 2014.

[5] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*, MIT Press, Aug. 2005

[6] J. Towler, B. Krawiec, and K. Kochersberger, "Terrain and Radiation Mapping in Post-Disaster Environments Using an Autonomous Helicopter," Remote Sensing, vol. 4, p. 1995-2015