Autonomous Robots and Environmental Mapping

Mitchell Horning, Matthew Lin, Siddarth Srinivasan, and Simon Zou

UCLA Applied Math REU - Summer 2014

Mentors: Ke Yin and Matt Haberland
Project PIs: Andrea Bertozzi, Stan Osher, and Luminita Vese, UCLA
Rick Chartrand and Brendt Wohlberg, Los Alamos National Lab

Overview

- The problem
- The robots
- The algorithm
- Preliminary results
- Further steps

Motivating the Problem

- Underwater mapping with submarines
  - Cannot get an overhead view
  - Can only send data before and after travelling the entire path

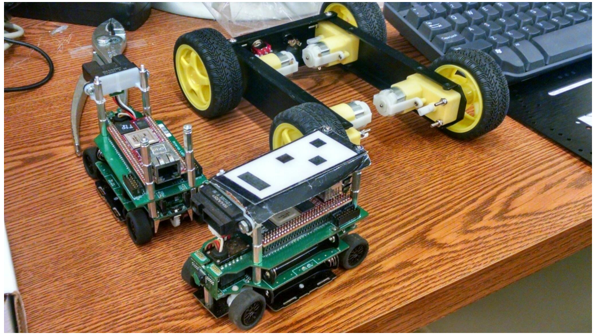
The Problem

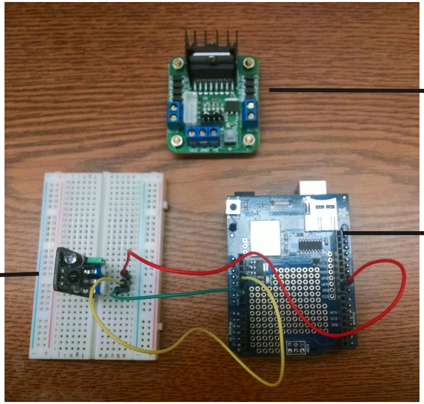
- Goal: Using simple robots and sensors to map an environment
- Constraints:
    - Limited bandwidth / Infrequent communication
    - Limited on board memory / computing
        - Too costly to sample every point in the environment
    - Want to minimize number of measurements taken

A Better Approach

- Sampling every point does not work well within the given constraints
- Compressed sensing
  - Recovering the signal despite undersampling
  - Travel a path and integrate along the path
    - Path hits many points, but the data consists of just the path’s start and end points and the single value found by integrating

The Robots

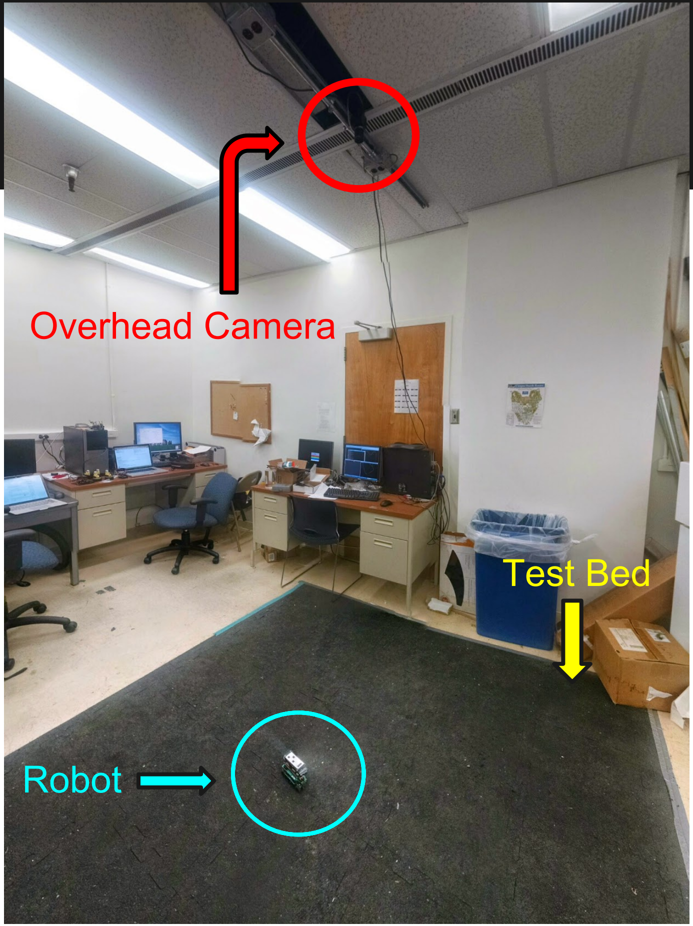




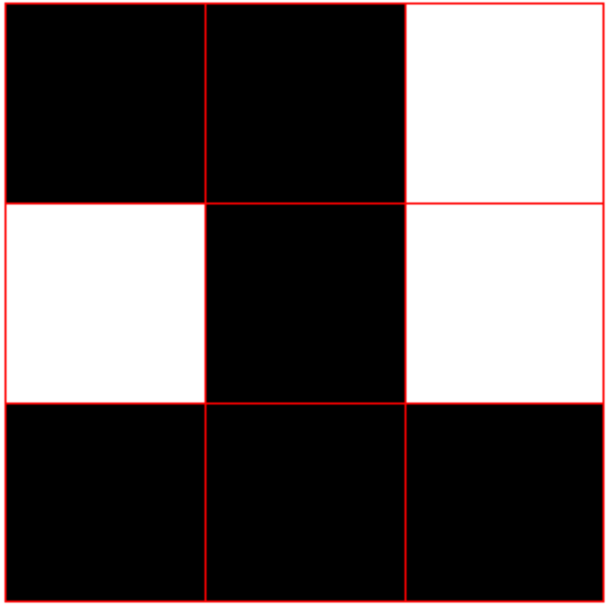
Motor Controller

Arduino with AdaFruit WiFi Shield

Reflectance Sensor

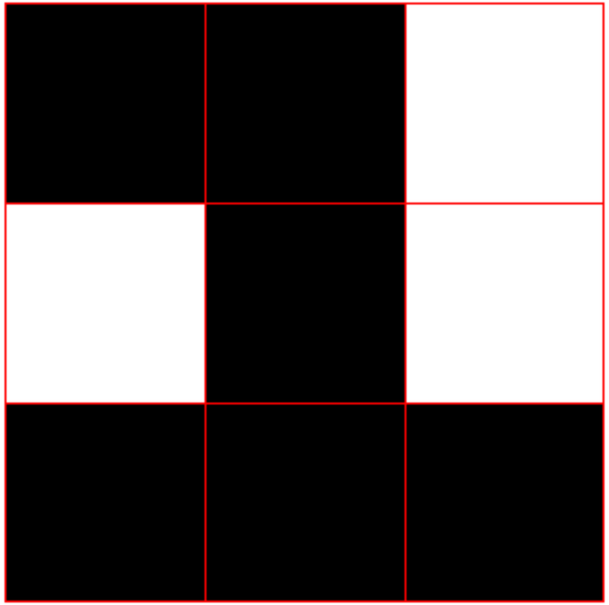


The Algorithm



Let u be the image we wish to reconstruct. In this example, consider this 3x3 binary image (black=0, white=1).

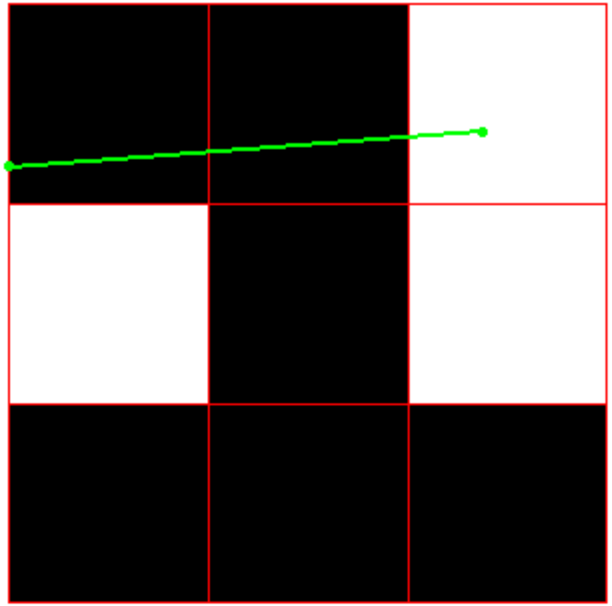
The Algorithm



Let u be the image we wish to reconstruct. In this example, consider this 3x3 binary image (black=0, white=1).

Given a set of paths and the corresponding path integrals over the pixel values of u, denoted g, how do we reconstruct the original image?

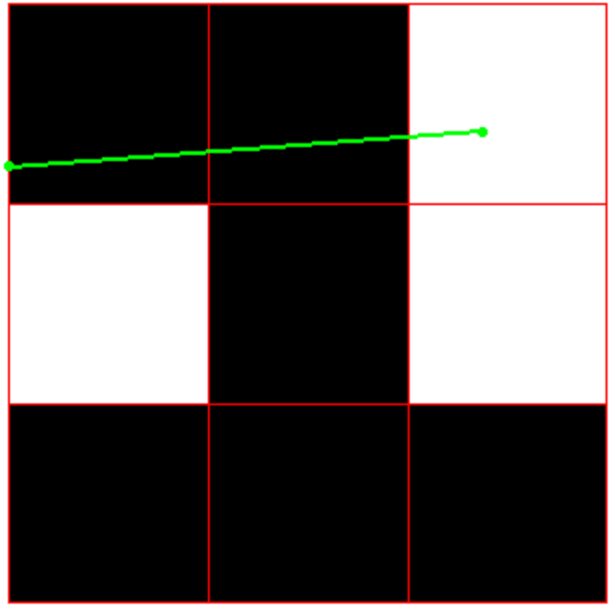
The Algorithm



Let *u* be the image we wish to reconstruct. In this example, consider this 3x3 binary image (black=0, white=1).

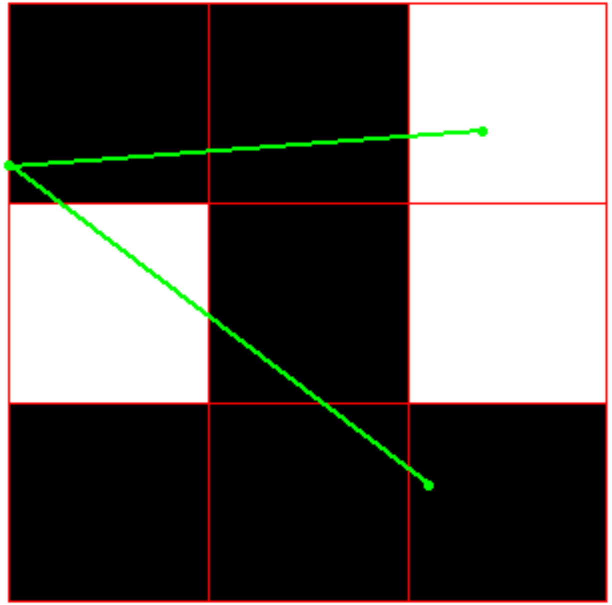
Given a set of paths and the corresponding path integrals over the pixel values of *u*, denoted *g*, how do we reconstruct the original image?

The Algorithm



$$0.98u_{1,1} + 1.00u_{1,2} + 0.38u_{1,3} = 0.38$$

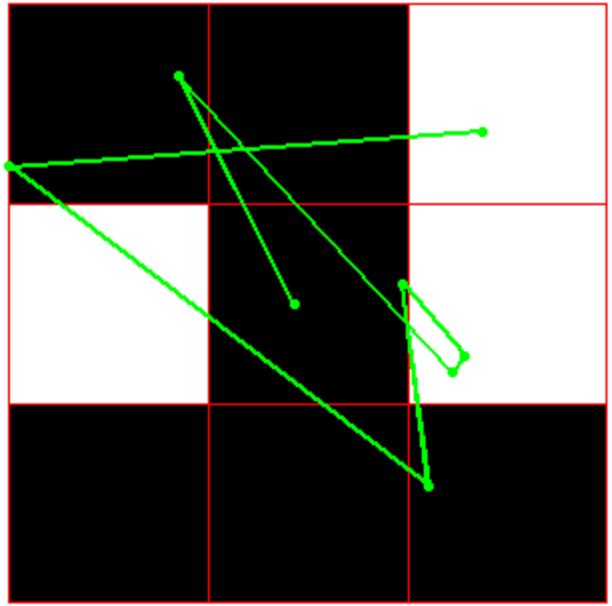
The Algorithm



$$0.98u_{1,1} + 1.00u_{1,2} + 0.38u_{1,3} = 0.38$$

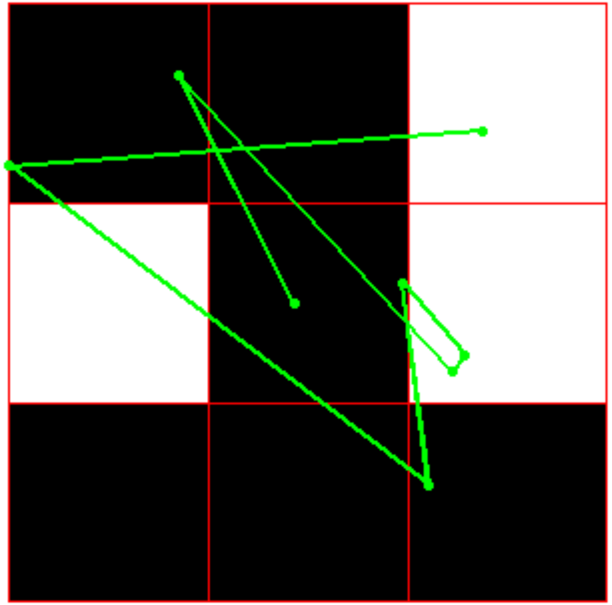
$$0.29u_{1,1} + 0.94u_{2,1} + 0.71u_{2,2} + 0.55u_{3,2} + 0.14u_{33} = 0.94$$

The Algorithm



$$
\begin{aligned}
0.98u_{1,1} + 1.00u_{1,2} + 0.38u_{1,3} &= 0.38 \\
0.29u_{1,1} + 0.94u_{2,1} + 0.71u_{2,2} + 0.55u_{3,2} + 0.14u_{33} &= 0.94 \\
\vdots \qquad\qquad &\quad \vdots \\
c_{m,1}u_{1,1} + c_{m,2}u_{2,1} + \cdots + c_{m,9}u_{3,3} &= g_m \\
\vdots \qquad\qquad &\quad \vdots \\
0.32u_{1,1} + 0.39u_{1,2} + 0.57u_{2,2} &= 0.
\end{aligned}
$$

The Algorithm



$$
\begin{aligned}
0.98u_{1,1} + 1.00u_{1,2} + 0.38u_{1,3} &= 0.38 \\
0.29u_{1,1} + 0.94u_{2,1} + 0.71u_{2,2} + 0.55u_{3,2} + 0.14u_{33} &= 0.94 \\
\vdots \qquad & \quad \vdots \\
c_{m,1}u_{1,1} + c_{m,2}u_{2,1} + \cdots + c_{m,9}u_{3,3} &= g_m \\
\vdots \qquad & \quad \vdots \\
0.32u_{1,1} + 0.39u_{1,2} + 0.57u_{2,2} &= 0.
\end{aligned}
$$

$$Au = g$$

Reconstruction

$$\min_{u} ||Au - g||^2$$

Reconstruction

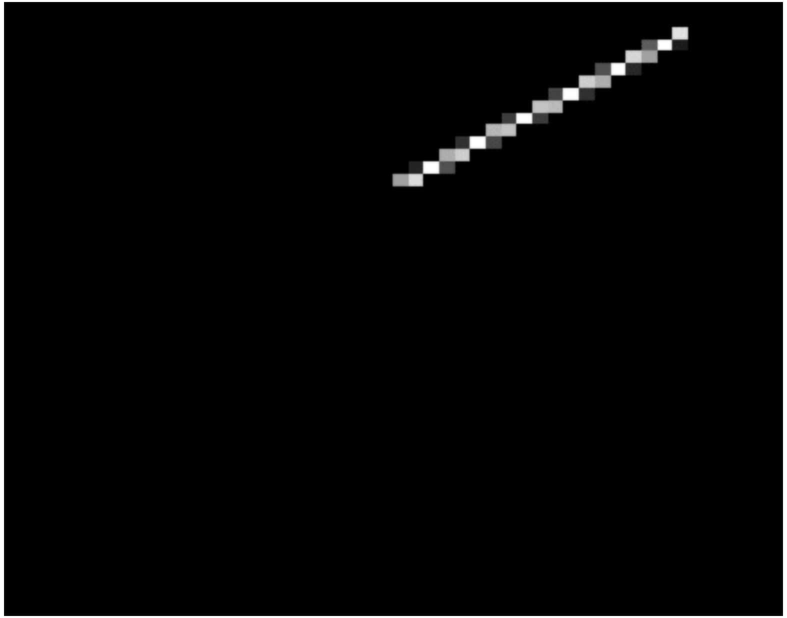
$$\min_{u} ||Au - g||^2 + |u|_1$$

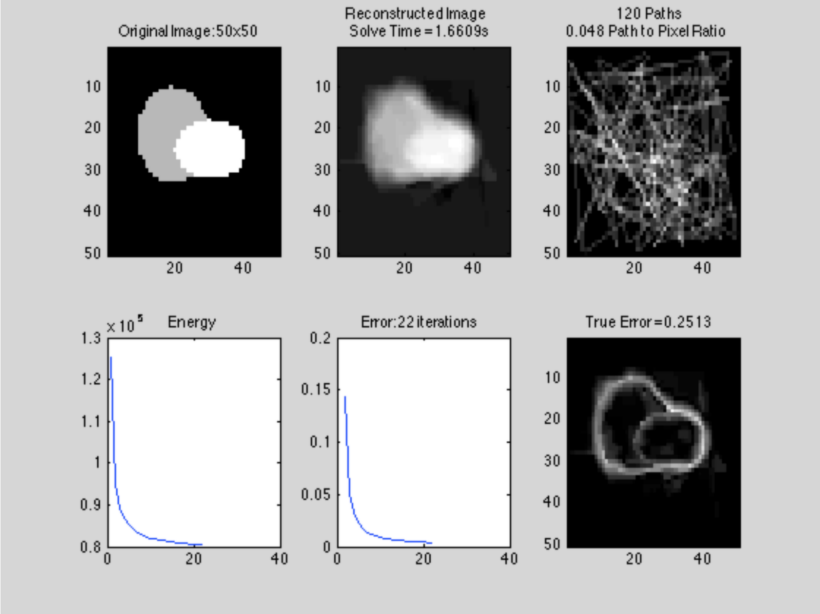
Reconstruction

$$\min_{u} ||Au - g||^2 + |u|_1 + |\nabla u|_1$$

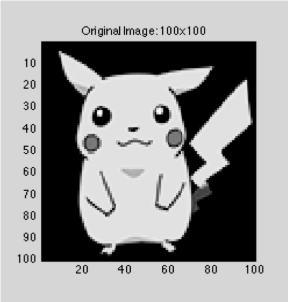
- We solve this optimization problem using an algorithm called Split Bregman
- This works because we assume the image is sparse
- Time complexity is proportional to the size of A (the number of pixels times the number of paths).

Preliminary / Simulated Results

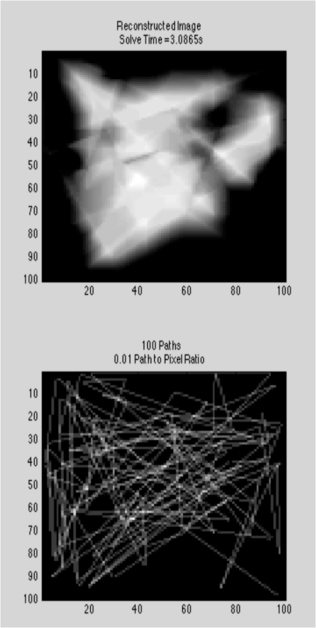




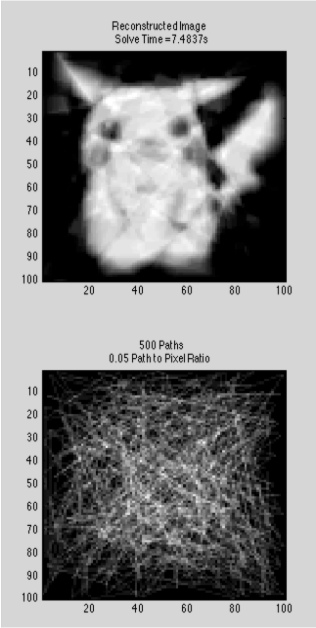
Preliminary / Simulated Results



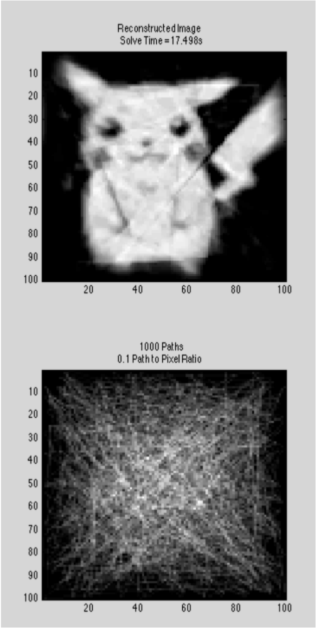
Original Image - 100x100
(10,000 pixels/unknowns)



100 Paths



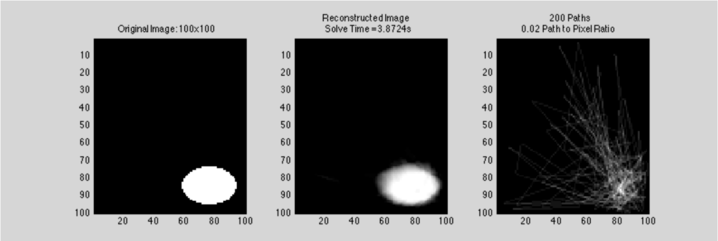
500 Paths

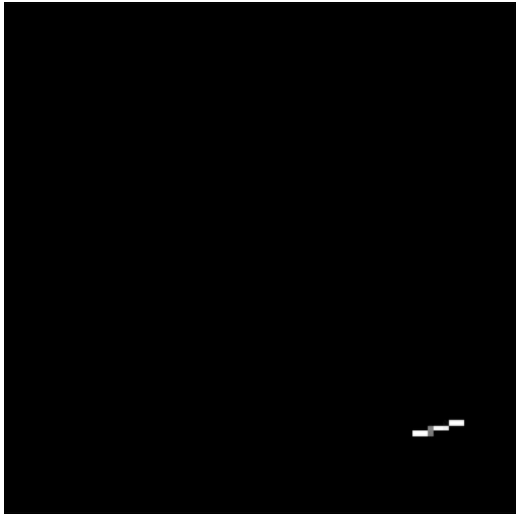


1000 Paths

Further Steps

- Coarse Reconstruction + Adaptive pathing
- Accounting for noise
- Collect data with robots





Summary / Conclusion

- Goal is to use robots to map an environment subject to certain constraints
- Take measurements by integrating along a path, then solve the underdetermined system with compressed sensing
- Questions?