Zero-Shot Ground Vehicle Navigation Enabled by Aerial-Based Digital Twin Scene Reconstructions

by

Desiree Jeanne Marika Fisker

A thesis submitted in conformity with the requirements
for the degree of Master of Applied Science

Institute for Aerospace Studies
University of Toronto

Zero-Shot Ground Vehicle Navigation Enabled by Aerial-Based Digital Twin Scene Reconstructions

Desiree Jeanne Marika Fisker
Master of Applied Science

Institute for Aerospace Studies
University of Toronto
2025

Abstract

Increased autonomy reduces risk and cost in unmanned ground vehicle operations, especially in remote or hazardous environments, and lowers operator burden. Virtual Teach and Repeat (VirT&R) is an extension of the Teach and Repeat (T&R) algorithm that enables GPS-denied, zero-shot navigation in previously untraversed environments. We use aerial imagery to build a simulation so an operator can define a route by virtually piloting a modelled robot. Along this route, we form a pose graph by associating submaps from a virtually generated point cloud, which enables a real robot to execute the mission in the physical environment. We validate VirT&R through multiple deployments across different sites, seasons, terrain types, and sensor configurations, under realistic operational constraints, and with multiple scene-reconstruction pipelines. On over 33 km of autonomous driving, VirT&R achieved smooth, reliable path-tracking with closed-loop performance comparable to conventional T&R—without requiring an operator to manually teach the route in situ.

Acknowledgements

I want to express my utmost gratitude to Tim Barfoot and Melissa Greeff for their mentorship and guidance throughout my time as a master's student. Tim's seasoned insight, and leadership in evaluating research topics and methodologies, as well as in deciding when to pivot or proceed, were invaluable. Melissa's detailed support and feedback during my research also greatly contributed to the further development of my problem solving and writing skills. Together, Tim and Melissa continually raised the bar to ensure research was conducted and presented at a high standard. This taught me many valuable lessons that are pertinent to success, the most important being to fail fast and iterate.

The Defence Research Department of Canada supported this work via grant funding and actively collaborated by mirroring the experiments conducted with the algorithm developed in this thesis at their testing site in Suffield, Alberta. Thank you to Jack Collier for organizing and facilitating the use of the Argobot and the CFB Suffield site for expanded testing of our work. I also want to thank my colleagues, Alec Krawciw and Sven Lilge, for sharing their expertise with me and for supporting me through many of the firsts in an academic career. Additionally, thank you to Alec, Sven, Hunter Ma, and Anthony Becca, for collaborating on publications and help with experiments. As well, I thank my colleagues and friends from UTIAS and the U of T Robotics Institute for their comradery. Furthermore, I could not have done all this without my close friends and housemates Timo, Julia, and Chris, who have seen me through thick and thin, and have always been there for me, thank you all. As well, I'd like to thank my parents Pauline and Chris for their unwavering love and support, and everyone at Skydive Toronto who I got to work and jump with over my years in Toronto for being amazing people and making sure my life was a good balance of work and play.

Lastly, I would like to thank the musical stylings of David Bowie, ABBA, Queen, Electric Light Orchestra, Fleetwood Mac, Radiohead, The Offspring, and others in those genres for keeping my spirits high during long nights spent gathering data for my experiments. I am so grateful that the two years I spent with ASRL will be part of the good old days I will get to look back on.

Contents

List of Tables

List of Figures

List of Acronyms

AGL	above ground level
CSV	comma separated value
DoF	degrees of freedom
DSM	Digital Surface Models
EXIF	Exchangeable Image File
FMCW	Frequency Modulated Continuous Wave
FoV	field of view
GCP	Ground Control Point
GNSS	Guidance Navigation Satellite System
GPS	Global Positioning System
GPU	Graphic Processing Unit
GS	Gaussian Splatting
ICP	Iterative Closest Point
LIDAR	Light Detection and Ranging
LPIPS	Learned Perceptual Image Patch Similarity
LT&R	Lidar Teach and Repeat
MLP	Multi-Layer Perceptron
MPC	Model Predictive Control
MVS	Multi View Stereo
NeRF	Neural Radiance Field
PnP	Perspective-n-Point
PoV	point of view
PTE	Path Tracking Error
PSNR	Peak Signal to Noise Ratio
RADAR	Radio Detection and Ranging
RANSAC	Random Sample Consensus
RGB	Red Green Blue
RMSE	Root Mean Squared Error
ROS	Robot Operating System
RT&R	Radar Teach and Repeat

SIFT	Scale-Invariant Feature Transform
SLAM	Simultaneous Localization and Mapping
SfM	Structure-from-Motion
SSIM	Structural Similarity Index Measure
SVD	Singular Value Decomposition
TSDF	Truncated Signed Distance Function
T&R	Teach and Repeat
UAV	Unmanned Aerial Vehicle
UGV	Unmanned Ground Vehicle
URDF	Unified Robot Description Format
UTIAS	University of Toronto Institute for Aerospace Studies
UWB	ultra-wideband
VI-SLAM	Visual Inertial Simultaneous Localization and Mapping
VIO	Visual Inertial Odometry
VirT&R	Virtual Teach and Repeat
VirLT&R	Virtual Lidar Teach and Repeat
VirRT&R	Virtual Radar Teach and Repeat
VTRN	Visual Terrain Relative Navigation
VT&R	Visual Teach and Repeat
VT&R3	Visual Teach and Repeat 3

Notation

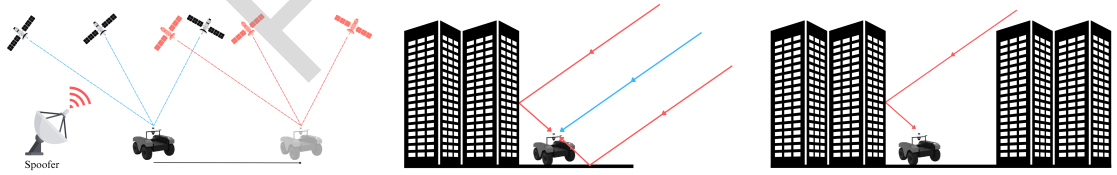
a	This font is used for quantities that are real scalars
$\mathbf{a}$	This font is used for quantities that are real column vectors
$\mathbf{A}$	This font is used for quantities that are real matrices
$\mathbf{1}$	The identity matrix
$\boldsymbol{\mathcal{F}}_a$	A vectrix representing a reference frame in three dimensions
$\underset{\rightarrow}{SE(3)}$	The special Euclidean group, a matrix Lie group used to represent poses
$\mathfrak{se}(3)$	The Lie algebra associated with $SE(3)$
$\mathbf{T}_{ba}$	A matrix in $SE(3)$ that transforms vectors from frame $\underset{\rightarrow}{\boldsymbol{\mathcal{F}}_a}$ to $\underset{\rightarrow}{\boldsymbol{\mathcal{F}}_b}$
$\exp(\cdot^\wedge)$	A Lie algebra operator mapping from $\mathfrak{se}(3)$ to $SE(3)$
$\ln(\cdot)^\vee$	A Lie algebra operator mapping from $SE(3)$ to $\mathfrak{se}(3)$

Chapter 1

Introduction

1.1 Motivation

The field of autonomous robot navigation is well-established, having produced numerous algorithms tailored to different sensing modalities and use cases, ranging from engineering and civil operations [1], to agricultural analysis [2], to humanitarian and emergency response [3]. A core capability that enables these operations is accurate and robust localization. In many cases, localization via route or waypoint following is facilitated by Global Navigation Satellite Systems (GNSS); however, connectivity is often subject to interruption or failure due to factors such as intentional jamming or spoofing, multipath effects, environmental interference, and accumulated errors. Thus, designing solutions that avoid these issues is of great interest, as many of the most valuable applications of autonomous vehicles—such as underground mining, military operations, and remote or extraterrestrial exploration—take place in environments where GNSS support is either unavailable or highly unreliable.



(a) Signal Jamming/Spoofing (b) Multipath Effects (c) Environmental Interference

Figure 1.1: A visualization of common issues faced when using GNSS in different scenarios.

Vision-based localization has been a prominent solution to replace or supplement GNSS, as cameras are small, lightweight, passive sensors unaffected by the factors that limit GNSS. Popular methods, such as Visual Inertial Odometry (VIO) [4], Visual Inertial Simultaneous Localization and Mapping (VI-SLAM) [5], and Visual Terrain Relative Navigation (VTRN) [6], rely on techniques derived from foundational concepts in state estimation, such as dead-reckoning and sensor-based pose estimation. While they can excel at providing scalable and robust real-time localization, these

tools are still limited by their perception of the world, which can be negatively impacted by common factors such as varying lighting conditions, weather, reflective surfaces, airborne particulates, viewpoint alterations, and subpar scene coverage or occlusions. Additionally, they are constrained by significant storage requirements, computational demands, odometry drift, error accumulation, and a limited field of view (FoV).

As technology advances, many sensors are becoming more readily accessible, increasing the research effort towards exploiting the benefits of different modalities. Supplementing a passive sensor, such as a camera, with active sensors such as Lidar or Radar may increase the complexity of a navigation pipeline, but it introduces the idea of creating a framework that uses different sensors to overcome environmental constraints such as lighting or weather to enhance autonomous capabilities. A summary of various sensor pros and cons can be found in Table 1.1.

Table 1.1: A comparison of common sensors used for mapping and localization on autonomous vehicles.

	Vision	**Lidar**	**Radar**
Pros	✓ Works regardless of most geometries ✓ Human interpretable	✓ Accurate ✓ Dense pointcloud	✓ Multiple returns per measurement ✓ Weather ✓ Long range
Cons	✗ Weather/Lighting ✗ Appearance changes	✗ Weather	✗ Noisy ✗ Sparse

The Visual Teach and Repeat 3 (VT&R3)[1] [7] framework provides a complete autonomy stack and addresses many of the mentioned practical challenges by simplifying navigation tasks into path-following routines. Through a learning phase (the *teach pass*) where a robot is manually piloted in an environment to generate a pose graph (the *map*) embedded with rich sensor data, and an autonomous route-following phase (the *repeat pass*) where the robot uses live sensor data to localize to the map, highly precise navigation is achieved. This approach has been demonstrated successfully on UGV platforms equipped with a 3D Lidar [8–10], a Radar [11], and Red Green Blue (RGB) vision sensors [7], as well as on UAV platforms using RGB vision sensors [12, 13]. T&R does have an inherent operational drawback; it requires a human operator to manually drive the robot through the environment during the teach pass to create the pose graph map while ensuring route traversability. This requirement becomes impractical or even dangerous in remote, hazardous, or inaccessible use case scenarios, such as facility inspections, battlefields, mining operations, or extraterrestrial exploration.

To overcome this limitation, we propose Virtual Teach and Repeat (VirT&R), visualized in Figure 1.2. Our algorithm modifies the teach pass of the existing T&R framework and leverages the affordable and lightweight camera sensors commonly found on commercial drones to capture de-

[1] https://github.com/utiasASRL/vtr3

tailed aerial images of outdoor environments. Advanced reconstruction tools then use these images to create realistic simulations and localization data of the captured scene, which can be customized for different sensor modalities. Figure 1.3 shows a generated mesh and localization layer data for Lidar and Radar-based VirT&R .

Figure 1.2: We present an architecture for navigating previously untraversed environments that uses 3D reconstruction tools to simulate the environment and create high-fidelity localization data from aerial imagery of the target environment. Shown here are the main steps of the framework, dubbed Virtual Teach and Repeat. The three steps are: 1. A scene reconstruction based on aerial imagery, a virtual driving simulation for virtual path definition, and the execution of an online mission with T&R.

We explore both learned and classical reconstruction methods to elucidate their respective strengths and limitations. This comparison further informs optimal tool selection for real-world navigation scenarios, ensuring that the most detailed and realistic data is used for reliable localization, regardless of lighting or environmental conditions, and enhances both operational safety and cost-effectiveness for unmanned missions through the use of multiple modalities and vehicle platforms.

1.2 Contributions

The primary contribution of this thesis is a new method for conducting the teach pass. Our novel architecture replaces the existing manually piloted teach pass with an offline, virtual, high-fidelity pilot simulation that allows a user to pilot their UGV through the environment to define a virtual path, which is then populated with localization data also obtained from the detailed 3D reconstruction created with the previously captured aerial imagery. This notably expands the operational functionality and usefulness of the Teach and Repeat framework in various ways by combining multiple platforms [14, 15] and bridging the gap between sensor modalities [15–17] to exploit their respective advantages. To our knowledge, this represents the first successful demonstration of zero-shot, closed-loop autonomous driving in outdoor environments using aerial scene reconstructions paired with ground-based Lidar or Radar localization. We also conducted several small studies throughout the evolution of VirT&R to verify that the most suitable tools, parameters, metrics, and evaluation

methods were used.

Thus, we present the following contributions as a part of the development of the Virtual T&R framework and the methodology used to assess its performance in varied terrains, utilizing traditional Lidar and Radar T&R [9, 11] as comparative baselines:

- A novel Virtual Teach and Repeat framework, integrated with the T&R codebase[2] via a published repository[3] that provides the following new capabilities:
  a. Navigate through an environment previously untraversed by either a human or the UGV autonomously without GPS assistance;
  b. Conduct multiple teach passes in a desired environment with only one aerial survey into said environment for image capture, as opposed to manually piloting a new path in the physical environment each time a new route is to be defined;
  c. Interactively and explore the target environment in simulation from multiple perspectives ahead of virtually piloting the UGV to determine the path.
- An experimental methodology for quantitatively comparing the accuracy and repeatability of VirT&R across the sim-to-real gap with real-world benchmarks using physical markings;
- Rigorous evaluation of the proposed architecture in various environments and seasons, with Lidar and Radar sensors, that consisted of the following sub-studies:
  a. An ablation study to determine optimal hyperparameters for the learned scene reconstruction method;
  b. An analysis of the ability to assess VirT&R performance with pseudo-GPS data created for virtual teach passes;
  c. A comparison of classical and learned 3D reconstruction methods.

The remainder of this thesis is organized as follows. Chapter 2 presents a detailed background of the fields and related work relevant to this thesis. Chapters 3 and 4 describe the methodology, implementation, and evaluation methods of the VirT&R pipeline. The various experiments and results are outlined in Chapters 5 and 6. Finally, Chapter 7 presents an analysis of our findings, outlines future work, and highlights the lessons learned.

[2] https://github.com/utiasASRL/vtr3
[3] https://github.com/desifisker/virtual_teach_vtr_wrapper

1.2.1 Associated Material

Publications

Portions of the technical work and writing presented in this thesis were submitted for publication with help from co-authors; however, all written content in this work was produced independently.

- Fisker D, Krawciw A, Lilge S, Greeff M, and Barfoot T D. "UAV See, UGV Do: Aerial Imagery and Virtual Teach Enabling Zero-Shot Ground Vehicle Repeat". In Proceedings of the IEEE/RSJ International Conference on Intelligent Robot Systems (IROS), to appear. Hangzhou, China, 19-25 October 2025.

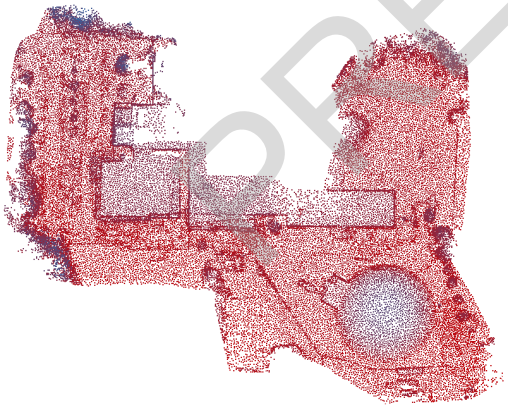
Code

The code for the vtr_virtual_teach package that interfaces with the existing T&R codebase, along with a dockerized setup of the required supporting software and custom scripts for VirT&R, is available at https://github.com/desifisker/virtual_teach_vtr_wrapper, and the official T&R codebase is available at https://github.com/utiasASRL/vtr3.

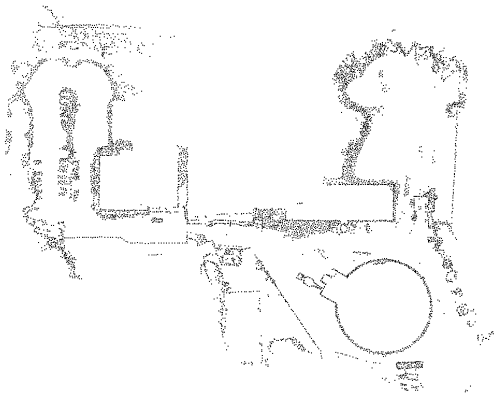
Video

- Virtual Teach and Repeat https://www.youtube.com/watch?v=0C8iAYP3atM

(a) Photo-Textured Triangular Mesh Exported from Pix4D



(b) Virtual Lidar Point Cloud Created from Pix4D Point Cloud and Coloured by Elevation in the Z Direction



(c) Virtual Radar Point Cloud Created from Pix4D Point Cloud

Figure 1.3: Visualization of the high-fidelity photo-textured mesh, virtual Lidar point cloud, and virtual Radar scan extracted point cloud data products produced from the reconstruction step in the VirT&R pipeline that enable virtual teach map creation.

Chapter 2

Related Works

2.1 State Estimation Overview

This section introduces foundational concepts and conventions used throughout this thesis for camera pose and state estimation, point cloud alignment, and rigid-body transformation theory. These topics form the basis for our various mapping, reconstruction, and localization modules, as well as Teach and Repeat itself.

2.1.1 Three-Dimensional Geometry and Localization

Vehicles that translate and rotate have three degrees of freedom (DoFs) for rotation and three degrees of freedom for translation. Understanding these DoFs is crucial, as we deal with multiple types of moving vehicles, in addition to their sensor measurements. We refer to their six DoF configuration as a *pose*, which consists of both position and orientation. This section introduces the mathematical tools used to represent poses and transformations throughout Chapters 3, 4, 5, and 6 where we describe the VirT&R framework, explain how we obtain path-tracking error, detail our attempt at creating pseudo-GPS for evaluation, as well as the results of the pipeline. For detailed derivations, refer to Chapter 7 of State Estimation for Robotics [18].

A Lie group is a set of elements with an operation that combines any two elements to form a third in the same set. It is also a differentiable manifold where the elements in a matrix Lie group are matrices themselves. Rotations are represented using the special orthogonal group:

$$\mathrm{SO}(3) = \left\{ \mathbf{C} \in \mathbb{R}^{3\times 3} \,\middle|\, \mathbf{C}\mathbf{C}^T = \mathbf{1}, \det(\mathbf{C}) = 1 \right\}. \tag{2.1}$$

Rigid-body transformations are represented in the special Euclidean group:

$$\mathrm{SE}(3) = \left\{ \mathbf{T} = \begin{bmatrix} \mathbf{C} & \mathbf{r} \\ \mathbf{0}^T & 1 \end{bmatrix} \in \mathbb{R}^{4\times 4} \,\middle|\, \mathbf{C} \in \mathrm{SO}(3), \mathbf{r} \in \mathbb{R}^3 \right\}. \tag{2.2}$$

Equations 2.1 and 2.2 must be true while satisfying the required group axioms: closure, asso-

ciativity, identity, and invertibility.

Transformations between coordinate frames in this work follow the robotics convention, where a transformation $\mathbf{T}_{ab} \in \mathrm{SE}(3)$ transforms a homogeneous point expressed in frame $\boldsymbol{\mathcal{F}}_b$ to frame $\boldsymbol{\mathcal{F}}_a$ such that $\mathbf{r}_a = \mathbf{T}_{ab}\mathbf{r}_b$.

We define the following frames, commonly referred to throughout this work:

- $\boldsymbol{\mathcal{F}}_{\mathrm{i}}$: The fixed inertial frame (world frame);
- $\boldsymbol{\mathcal{F}}_{\mathrm{m}}$: The map frame (the pose graph map);
- $\boldsymbol{\mathcal{F}}_v$: The vertex frame, a local pose within the map;
- $\boldsymbol{\mathcal{F}}_r$: The robot frame, located at the robot's base;
- $\boldsymbol{\mathcal{F}}_s$: The sensor frame (e.g., Lidar or Radar), rigidly mounted on the robot.

In T&R, a trajectory is encoded in a pose graph as a series of temporally ordered vertices, each associated with a timestamp and a relative transformation (an *edge*) connecting to its predecessor. These relative transforms are elements of $\mathrm{SE}(3)$ and are defined as transformations from the previous pose to the current pose ($\mathbf{T}_{ab}$), all resolved in the local map frame $\boldsymbol{\mathcal{F}}_m$. A visualization of a pose graph can be seen in Figure 2.1

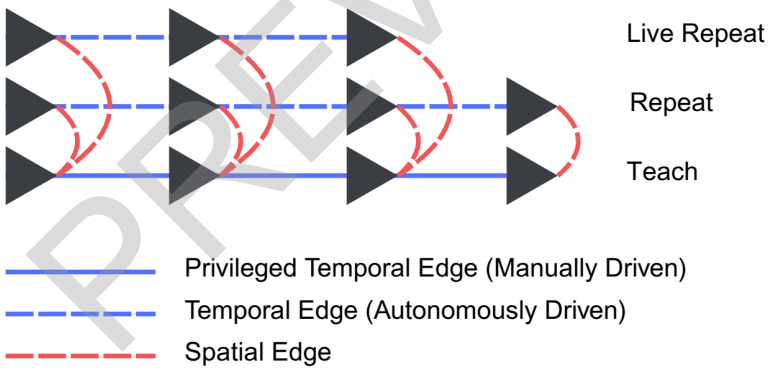


Figure 2.1: Visualization of the pose graph structure used in T&R. Shown here are the Privileged Edge (teach pass), a repeat Temporal Edge localizing to the Privileged Edge, and a subsequent repeat Temporal Edge in progress, also localizing to the Privileged Edge.

The pose graph in VirT&R is built from a list of time-stamped $\mathrm{SE}(3)$ transformation matrices obtained from the relative motion defined by driving a UGV model in a high-fidelity simulation of the environment as opposed to driving it in the real world as in traditional T&R. The first transformation is set to be identity to define the origin, and subsequent transformations (assumed to be perfect odometry) are used as edges to yield each vertex's pose in the map frame. These transformations are expressed as robot-to-robot transforms and can be resolved in the world frame.

At each vertex, a submap point cloud is optionally created to save data taken by onboard sensors, which creates a representation of the environment at that instant. These submaps, created during a traditional or virtual teach, are what is used by the T&R when running a repeat pass on a vehicle to

perform localization. Submaps are made by transforming the virtually generated environment point cloud into the current sensor frame (where the sensor is currently located in the virtual point cloud of the environment) and taking a small cropping of it. To do this, the environment point cloud is re-based using the known starting position in the world frame such that the first vertex becomes the origin (i.e., $\mathbf{T}_{v,0} = \mathbf{I}$) in the map frame, and the remaining transformations are compounded relative to this new origin.

The submaps are then saved to the map in the local vertex frames, allowing us to maintain the topometric map representation and recover our position using that information. The full sensor-to-map transformation is computed as

$$\mathbf{T}_{s,m} = \mathbf{T}_{s,r}\mathbf{T}_{r,v}\mathbf{T}_{v,m}, \tag{2.3}$$

where:

- $\mathbf{T}_{s,r}$ is the fixed extrinsic calibration between sensor and robot base;
- $\mathbf{T}_{r,v}$ is identity, since each vertex pose is defined in the robot frame;
- $\mathbf{T}_{v,m}$ is the absolute pose of the current vertex with respect to the map.

Each vertex also stores a pointer to the most recent vertex with a submap (that is Identity if a submap is located at the vertex in question), which are created based on spatial and rotational thresholds (e.g., 1.5 m or 30° of motion). The relative transformation between the current vertex and the current submap at a prior vertex is computed as follows:

$$\mathbf{T}_{\text{submap vertex,current vertex}} = \mathbf{T}_{\text{current vertex},m}\mathbf{T}^{-1}_{\text{submap vertex},m}. \tag{2.4}$$

This relative pose is saved and used later to help initialize a live scan against the stored submap during localization. Given a set of 3D landmark observations $\{\mathbf{p}_v\}$ from a local vertex submap frame $\boldsymbol{\mathcal{F}}_v$ at time t_k, and corresponding landmarks $\{\mathbf{p}_s\}$ from a live sensor frame $\boldsymbol{\mathcal{F}}_s$ at time t_{k+1}, we seek the transformation $\mathbf{T}_{k+1,k} \in \mathrm{SE}(3)$ that best aligns them.

For both the vertex and sensor frames, we have M measurements of the set of landmark points P given in the respective frames $\mathbf{r}_v^{p_i v}$ and $\mathbf{r}_s^{p_i s}$, where $i = 1 \dots M$. The goal is to find the rotation matrix, $\mathbf{C}_{vs}$, and translation, $\mathbf{r}_{vs}$, that will align the two sets of points. We also define w_i as the scalar weights for each point.

Following the derivation in State Estimation for Robotics [18], we can formulate the alignment as a least-squares problem over rotation $\mathbf{C} \in \mathrm{SO}(3)$ and translation $\mathbf{r} \in \mathbb{R}^3$, minimizing

$$J(\mathbf{C}, \mathbf{r}) = \frac{1}{2}\sum_{i=1}^{M} w_i \|\mathbf{y}_i - \mathbf{C}(\mathbf{p}_i - \mathbf{r})\|^2, \tag{2.5}$$

under the constraint $\mathbf{C} \in \mathrm{SO}(3)$. A closed-form solution can be obtained by computing weighted centroids, forming a cross-covariance matrix, and solving for $\mathbf{C}$ using Singular Value Decompo-