VLOAM
Robotic Localization and Mapping
16833

Fall 2024

Dan McGann

Slides adapted Montiel Abello and Eric Westman

Outline:

1. Motivation
2. Conceptual Overview
3. Visual Odometry
4. Laser Odometry
5. Things to think about

Outline:

1. **Motivation**
2. Conceptual Overview
3. Visual Odometry
4. Laser Odometry
5. Things to think about

Visual LIDAR Odometry and Mapping

- Homework 3: SLAM Solvers
  - Param `odoms` Odometry measurements between i and i+1 in the global coordinate system. Shape: (n_odom, 2).

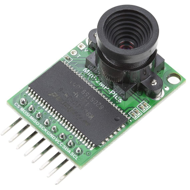
- How did we actually get these odometry measurements?

- Will they always be provided by an omniscient oracle like in the homework?

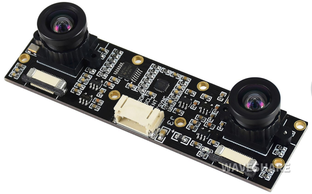
No. Enter VLOAM

Options for Sensors

- Odometry must be derived from raw data gathered by sensors on your robot



Monocular Camera

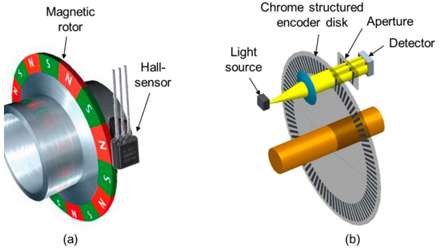


Stereo Camera

Depth Camera

LIDAR

GPS



Encoders

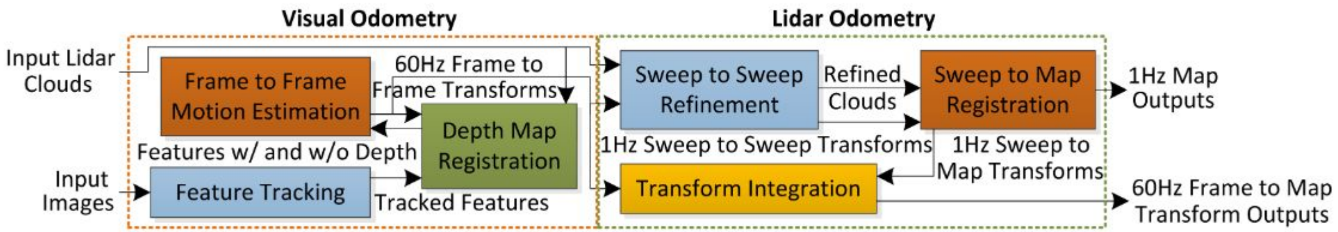
- Choice of sensors depends on (weight, size, cost, operation environment, desired accuracy, efficiency of available algorithms, etc).

Outline:

1. Motivation
2. **Conceptual Overview**
3. Visual Odometry
4. Laser Odometry
5. Things to think about

VLOAM Goal

- Given:
  - Sequence of high rate monocular Images $[I^0, I^1, ...I^t]$
  - Sequence of low rate raw LiDAR scans $[\mathcal{P}^0, \mathcal{P}^1, ...\mathcal{P}^t]$
- Estimate
  - The pose of the sensor $P^t \in SE(3)$ in a drifting frame
    - Assume calibration between LiDAR and camera
  - Implicit: estimate the odometry between consecutive poses



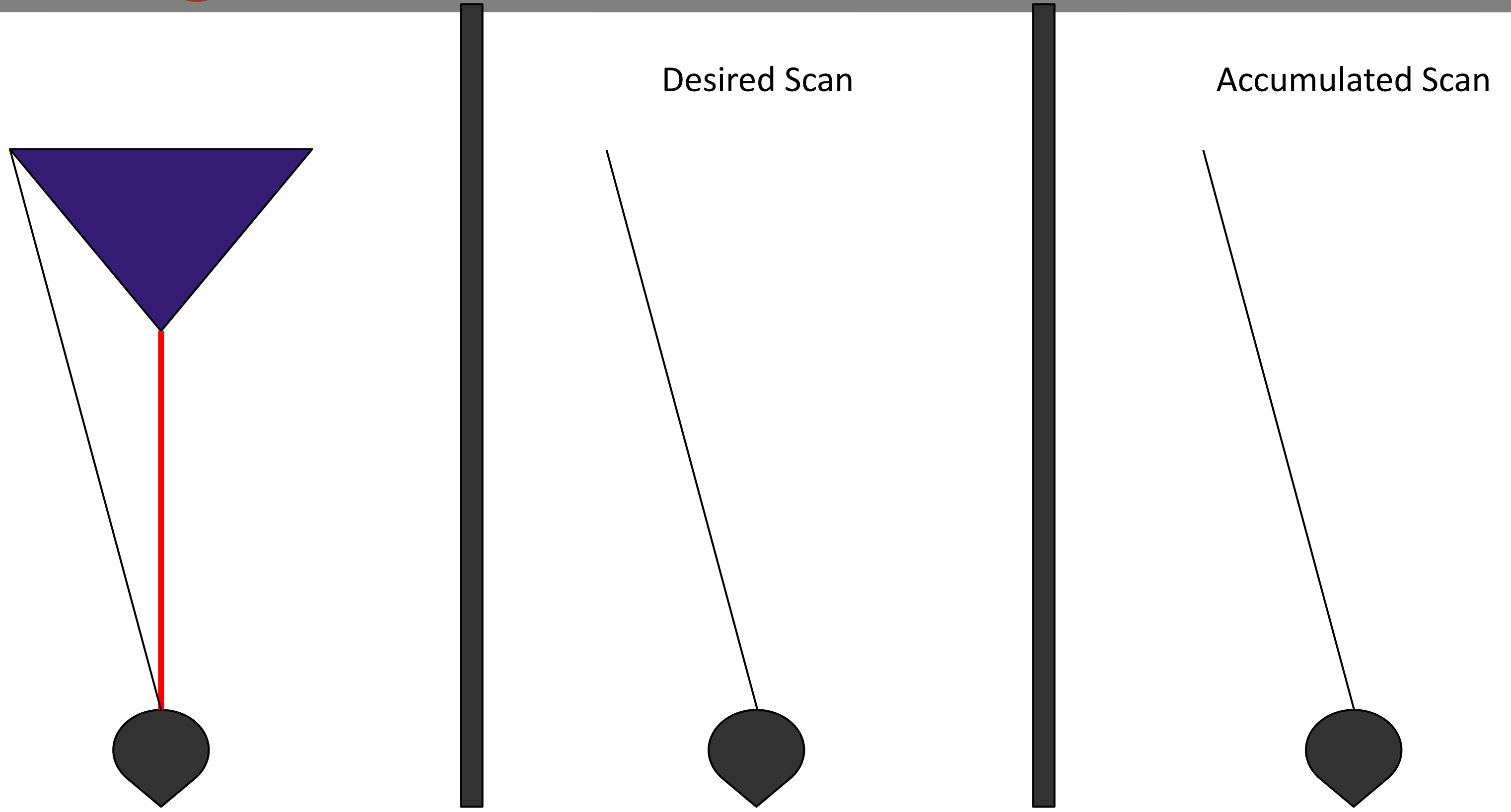
Challenge 1

- Monocular images suffer from scale ambiguity
- Given:
  - 2 consecutive images
  - Feature correspondences
  - Camera intrinsics
- We can only estimate the transform between camera centers up-to-scale!

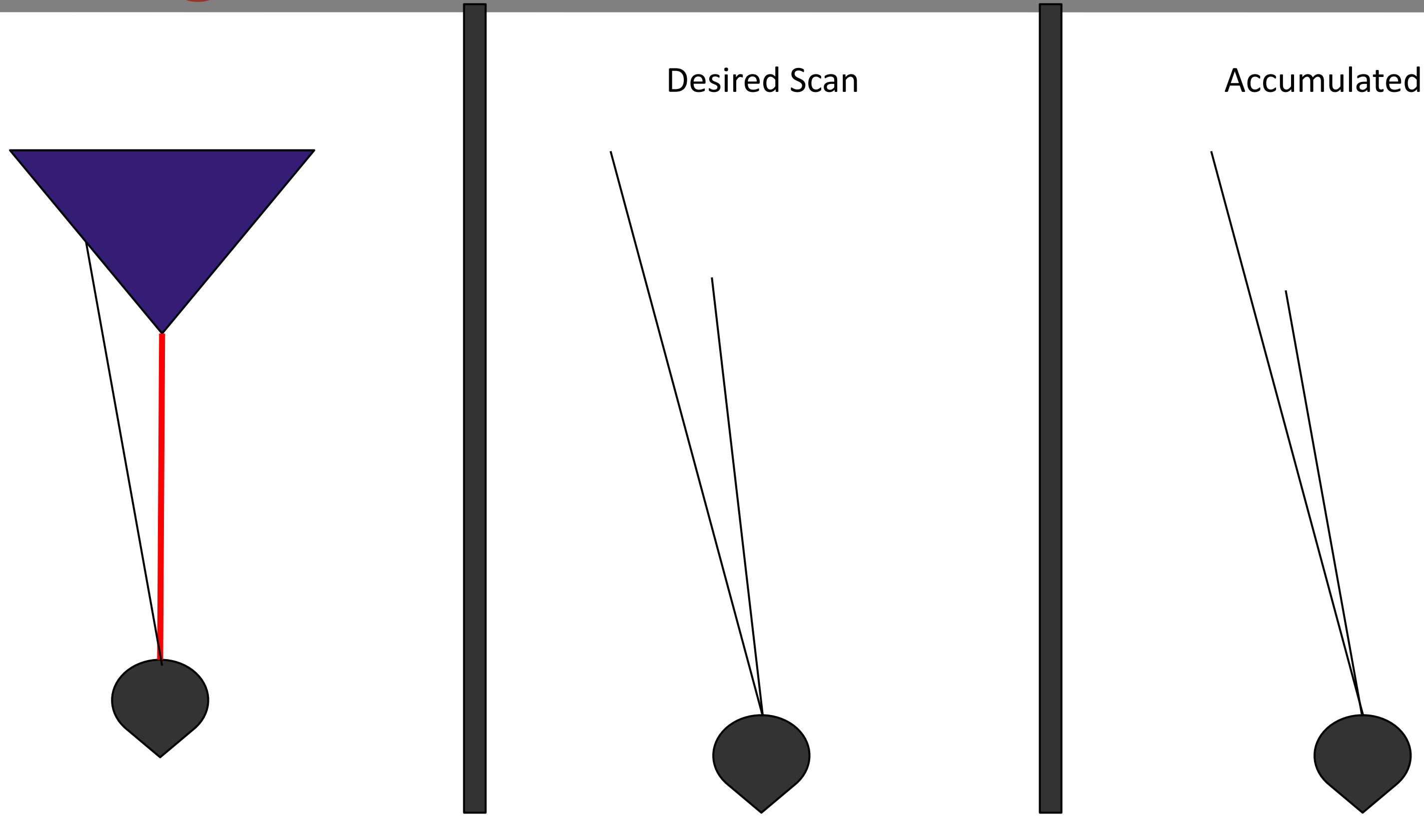
Challenge 2

- Raw LIDAR scans are very distorted
- Imagine a **LIDAR** moving towards an object along a path
- Motion may be complex and nonlinear

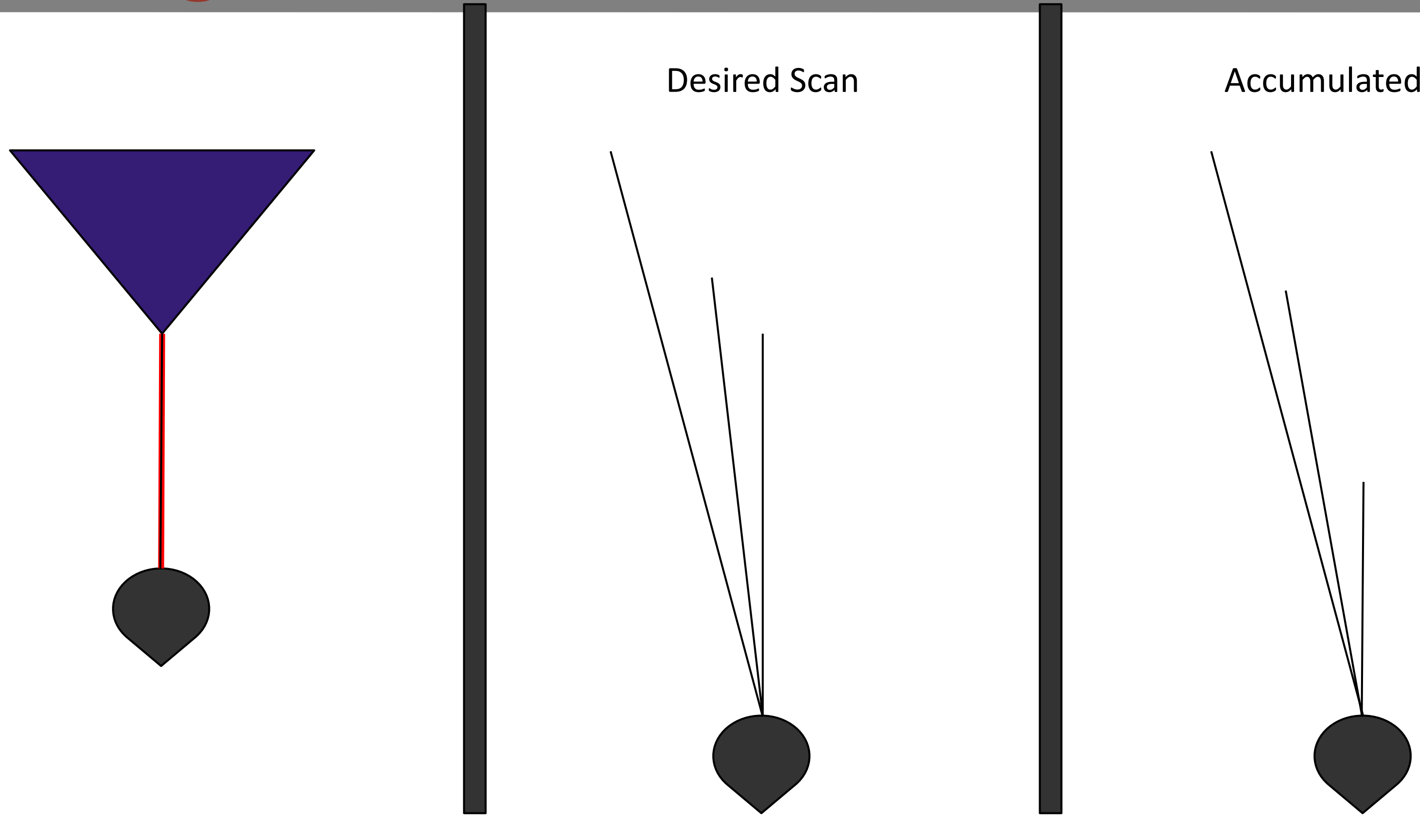
Challenge 2



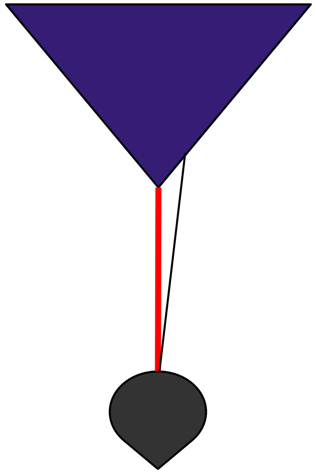
Challenge 2



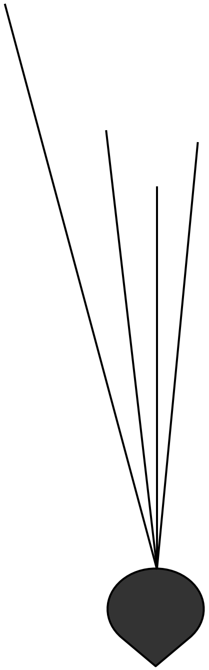
Challenge 2



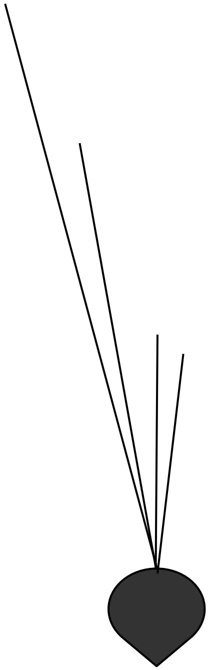
Challenge 2



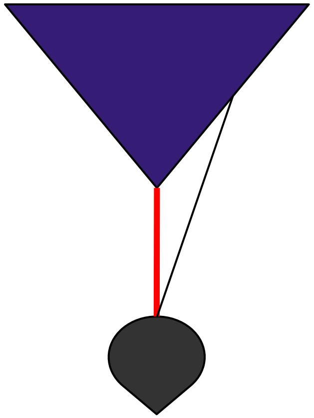
Desired Scan



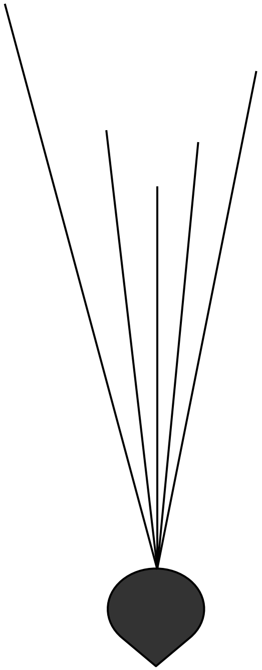
Accumulated Scan



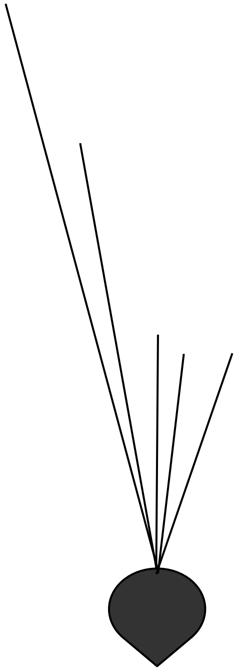
Challenge 2



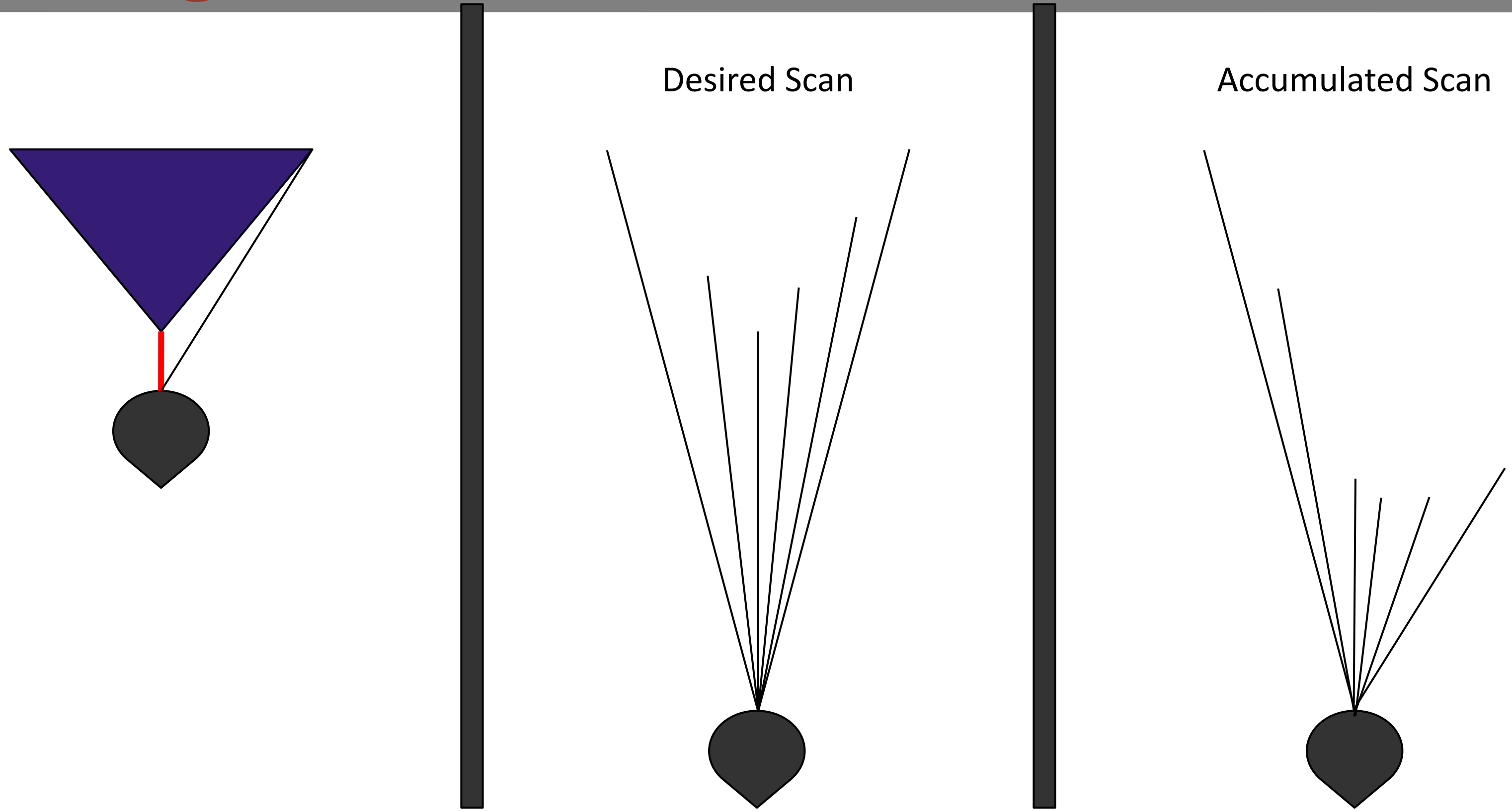
Desired Scan



Accumulated Scan



Challenge 2

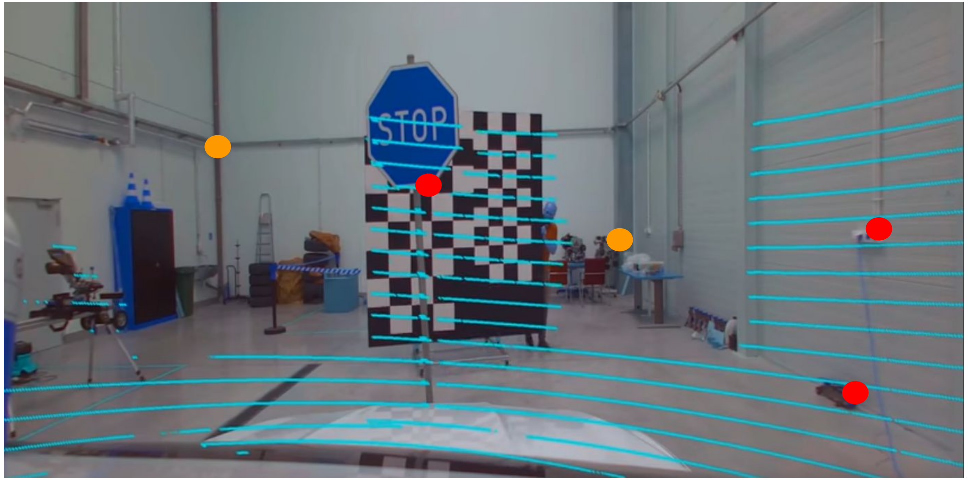


Outline:

1. Motivation
2. Conceptual Overview
3. **Visual Odometry**
4. Laser Odometry
5. Things to think about

VLOAM Visual Odometry

- Assume that we have depth information for some image features



Cyan: LiDAR points

Red: Image features w/ depth

Orange: Image features wo/ depth

- This assumption holds given we have a constructed metric map and we know the camera pose relative to map for the last frame

VLOAM Visual Odometry: Input + Goal

- New image, features: $I^k, \{ {}^S\bar{X}_i^k \}_i$
  - We cannot know depth, because new image may have been taken anywhere

- Previous image, features: $I^{k-1}, \{ {}^S\bar{X}_i^{k-1} \}_i \cup \{ {}^S X_j^{k-1} \}_j$
  - Some features with known depth and some with unknown depth

- **GOAL**: Solve the following for known feature correspondences
  - Features = Harris Corners, Correspondences = KLT Tracking

$$ {}^S X_i^k = R\, {}^S X_i^{k-1} + T $$

Carnegie Mellon
THE ROBOTICS INSTITUTE

VLOAM Visual Odometry: Math

- Relationship below allows us to define

$$^{S}X_i^k = R\ ^{S}X_i^{k-1} + T$$

- 2 Nonlinear equations
  - For correspondences with known depth in frame k-1

$$({}^{S}\bar{z}_i^k \boldsymbol{R}_1 - {}^{S}\bar{x}_i^k \boldsymbol{R}_3)\,{}^{S}\boldsymbol{X}_i^{k-1} + {}^{S}\bar{z}_i^k T_1 - {}^{S}\bar{x}_i^k T_3 = 0,$$
$$({}^{S}\bar{z}_i^k \boldsymbol{R}_2 - {}^{S}\bar{y}_i^k \boldsymbol{R}_3)\,{}^{S}\boldsymbol{X}_i^{k-1} + {}^{S}\bar{z}_i^k T_2 - {}^{S}\bar{y}_i^k T_3 = 0.$$

- 1 Nonlinear equations
  - For correspondences with UNKNOWN depth in frame k-1

$$\begin{bmatrix} -{}^{S}\bar{y}_i^k T_3 + {}^{S}\bar{z}_i^k T_2 \\ {}^{S}\bar{x}_i^k T_3 - {}^{S}\bar{z}_i^k T_1 \\ -{}^{S}\bar{x}_i^k T_2 + {}^{S}\bar{y}_i^k T_1 \end{bmatrix} \mathbf{R}\ {}^{S}\bar{\boldsymbol{X}}_i^{k-1} = 0.$$

* Special manipulations eliminate unknown variables (i.e. depth)

VLOAM Visual Odometry: Outcomes

- With N nonlinear equations, we can solve for unknown parameters:
  - T=[dx, dy, dz], R = ExpMap([a, b, c]^)
  - Recall: “Nonlinear Optimization” (Hw3)
  - Recall: “Rotations and Manifolds” (L14)

- Partial knowledge of depth allows recovery of this transform with known scale!

- Provides High Rate Odometry
  - Low-ish accuracy -> Non-trivial drift over time

Outline:

1. Motivation
2. Conceptual Overview
3. Visual Odometry
4. **Laser Odometry**
5. Things to think about

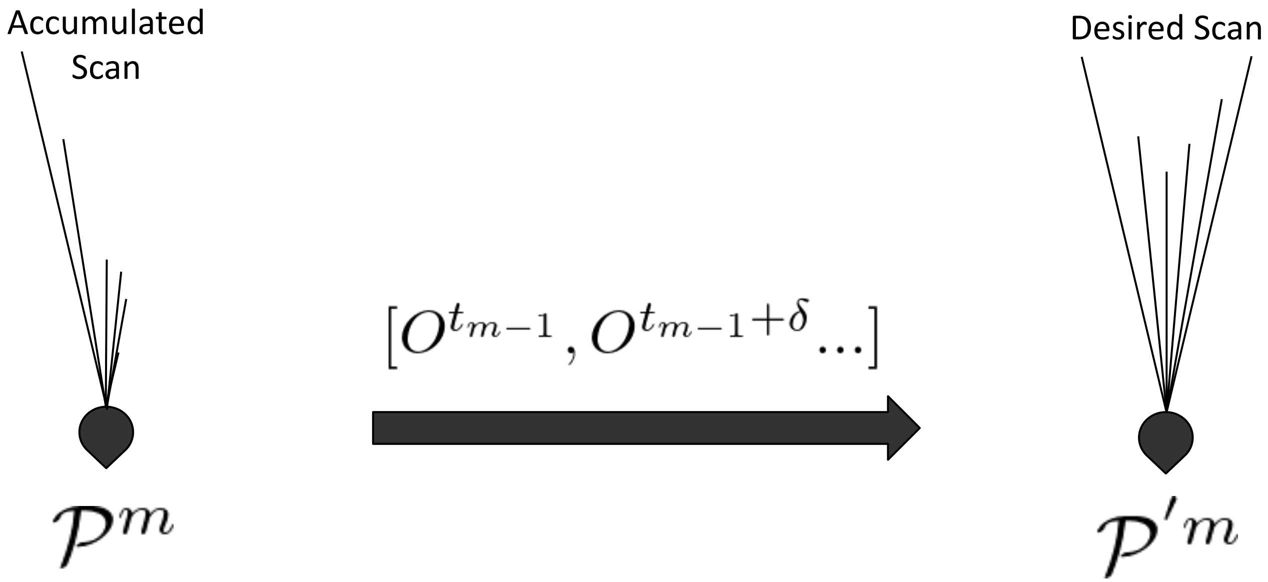
VLOAM LiDAR Odometry: Inputs + Goal

- New Raw LiDAR Scan: $\mathcal{P}^m$
  - Distorted due to motion of sensor while scan was occurring
- Previous LiDAR Scan: $\mathcal{P}''^{m-1}$
  - Undistorted!
- Odometry Sequence for period $\mathcal{P}^m$ was taken: $[O^{t_{m-1}}, O^{t_{m-1}+\delta} \ldots]$
  - From our visual odometry before
- Metric Map of points: $\mathcal{Q}^{m-1}$

- Goal: Refine estimate of relative pose between t^{m-1}, t^m

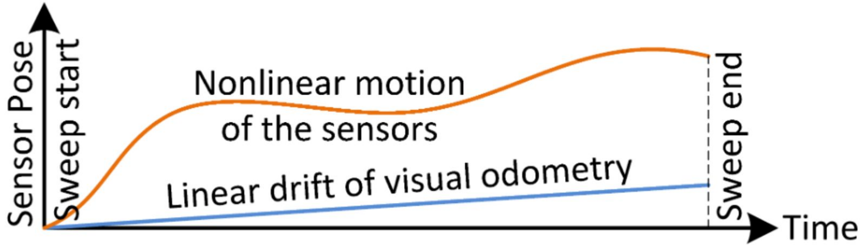
VLOAM LiDAR Odometry: Undistort

- Undistort $\mathcal{P}^m$ using the sequence of odometry from VO



VLOAM LiDAR Odometry: Undistort

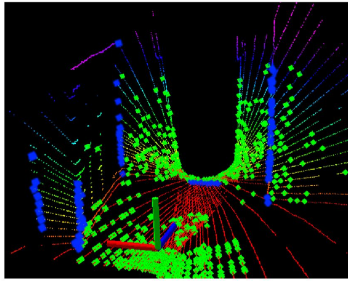
- Just using VO is still inaccurate!
  - VO drifts over the period in which the scan was taken
  - If we assume this drift can be described by constant velocity motion, then we can estimate it!



- Ex.

VLOAM LiDAR Odometry: Undistort

1. We first transform $\mathcal{P}'^{m}$ into the frame of $\mathcal{P}''^{m-1}$ (let's call this A)
   a. Ensures that points are directly comparable



2. We extract and match geometric features from the two scans
   a. Corners, and planes

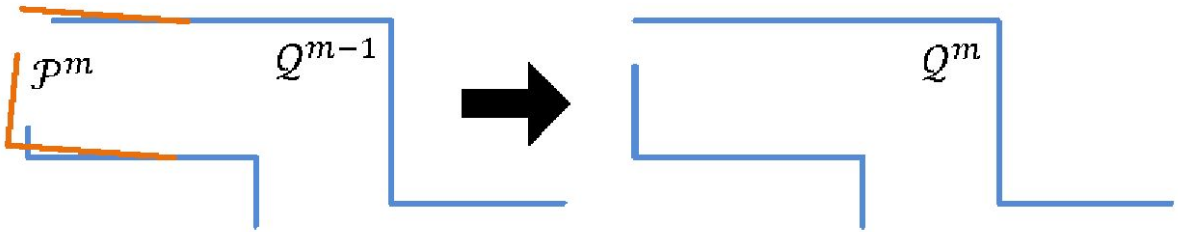
3. For each correspondence (indexed by i) we defined a distance
   - Where $T'_i = T'(t_i - t^m)/(t^{m+1} - t^m)$ and $T' = [\partial x, \partial y, \partial z, \partial a, \partial b, \partial c]$

$$f({}^{S}X_i^{m-1},\ {}^{S}X_i^{m},\ T'_i) = d_i$$

4. Nonlinear Optimization provides T'
   a. Used to further undistort $\mathcal{P}'^{m}$ into $\mathcal{P}''^{m}$

VLOAM LiDAR Odometry: Register

- Finally, we recover our goal (a pose relative to the map) by registering $\mathcal{P}''^m$ to Q^m to get P^m
- Registration accomplished with Iterative Closest Feature (ICF)
  - ICF is similar to Iterative Closest Point (Hw4)
  - ICF initialized by transform from VO + Undistort



Questions?

Things to think about

- i.e. Discussion questions / Further exploration
- How to linearize functions in VO w.r.t. [x,y,z,a,b,c]?
  - Real-time Depth Enhanced Monocular Odometry (2014)
- How do we perform matching of geometric features?
  - LOAM: Lidar Odometry and Mapping in Real-time (2014)

Things to think about

- Will our Map(Q) and poses in the map frame drift over time?
- If it does drift would that drift be correctable?
- Can we think of any ways to improve VLOAM's accuracy?