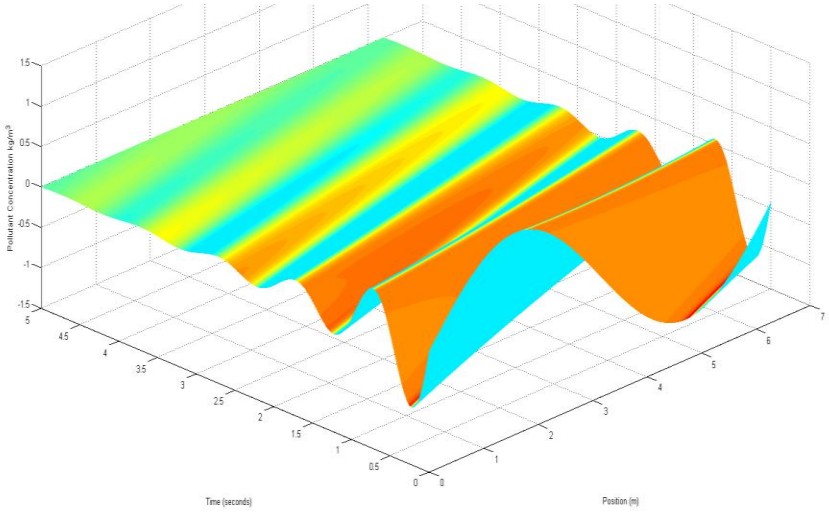
Real-time Landing Point Tracking and Relative Landing Point Vector Generation for Unmanned Aerial Vehicle Rooftop Landing Procedure Utilizing Computer Vision Methods

Student: Christian Howard
Mentor: Roland Brockers

Jet Propulsion Laboratory
California Institute of Technology

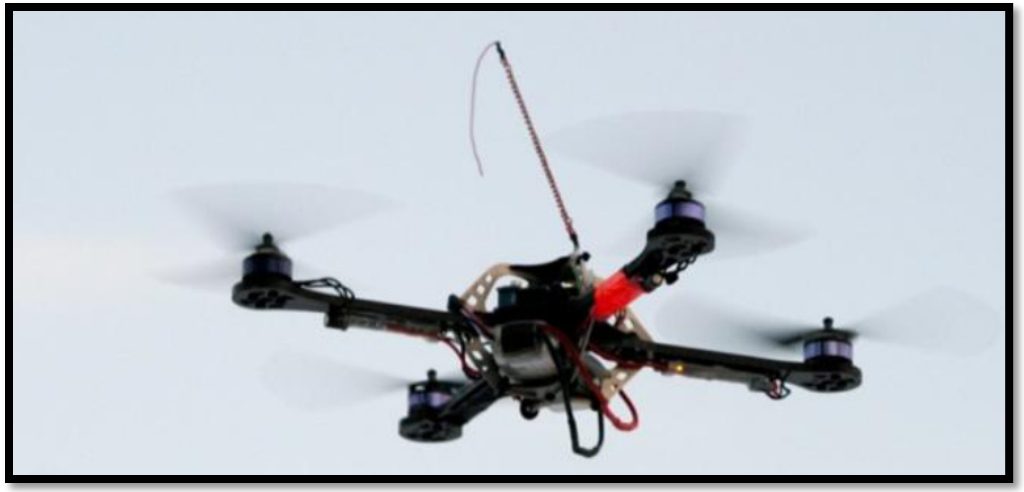
About Me

- University of Illinois at Urbana-Champaign
  - Aerospace Engineering Undergrad
- Interests
  - Computational Sciences
    - Computational Mechanics
    - Optimization
  - Dynamics/Controls
  - Robotics



The Project

- Autonomous Quadrotor
  - Develop autonomous rooftop landing software



- My role
  - Develop software to track a given landing point during descent
    - Helps control quadrotor accurately through descent

Implementation

- Software developed using ROS and C/C++
- Software utilized vision based methods
  - First tried using OpenCV, an open source computer vision library
  - Later migrated to using computer vision code developed here at JPL

Why?

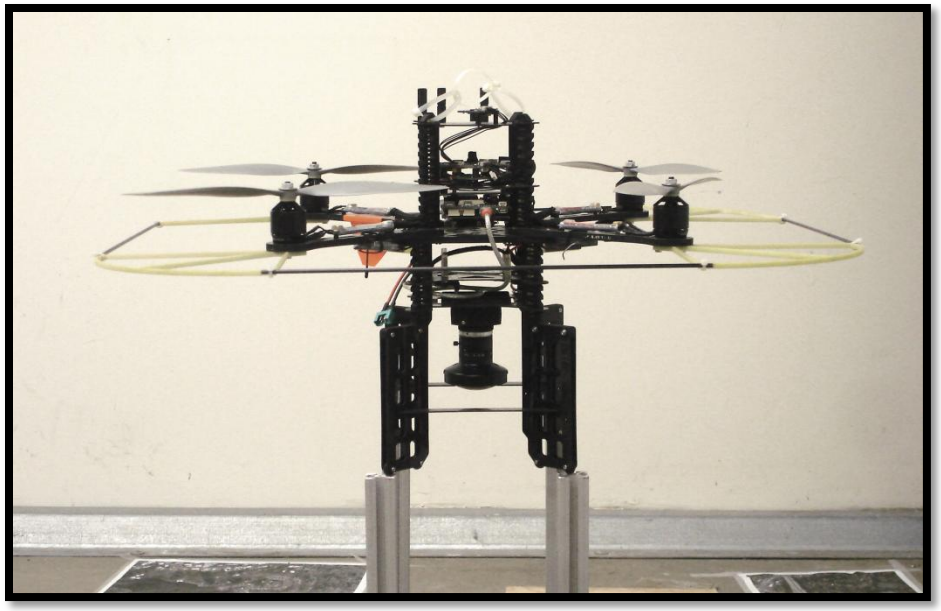
- Why autonomy?
  - Can help keep the UAV in air
  - Less work for pilot

Why?

- Why use vision based methods?
  - Pros
    - Cameras are light and energy efficient
    - Lots of useful data can be extracted
  - Cons
    - Can be computation heavy
    - Real-time algorithms are difficult to build

Why?

- Why use a quadrotor as the UAV?
  - Maneuverability
  - Hovering ability
  - Simple mechanically

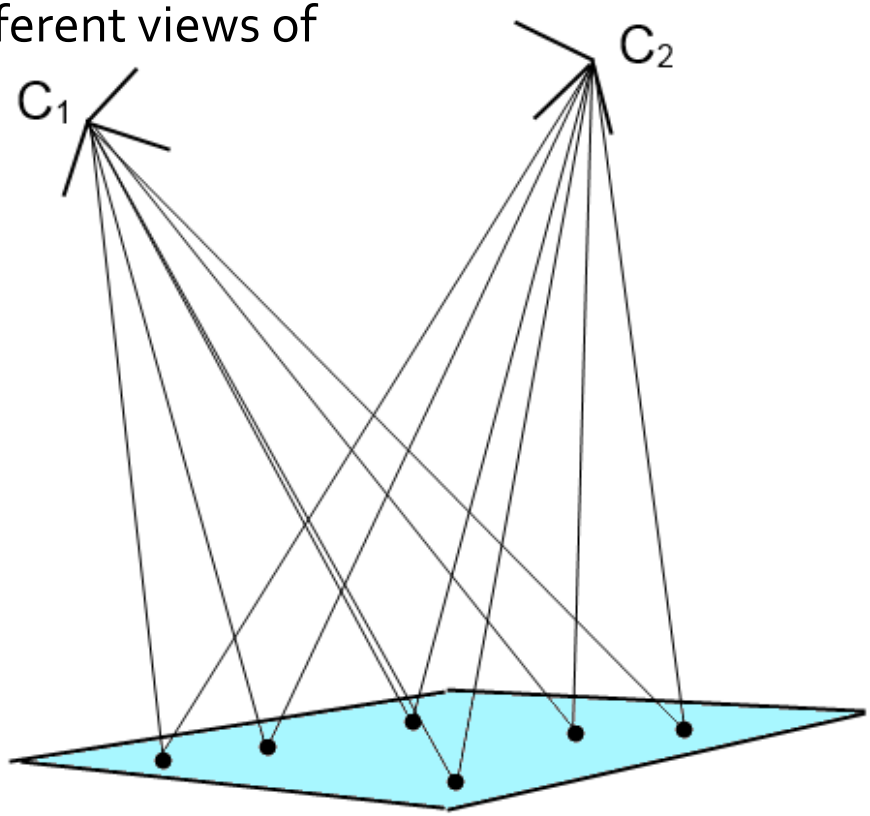


Approach ~ Tracking Landing Pose in Image

- Homography based tracking
  - Pros
    - Simple
    - Efficient once found
    - Can be used on all points on the plane it describes
  - Cons
    - Meant to represent features on a plane
    - Requires features matches to generate

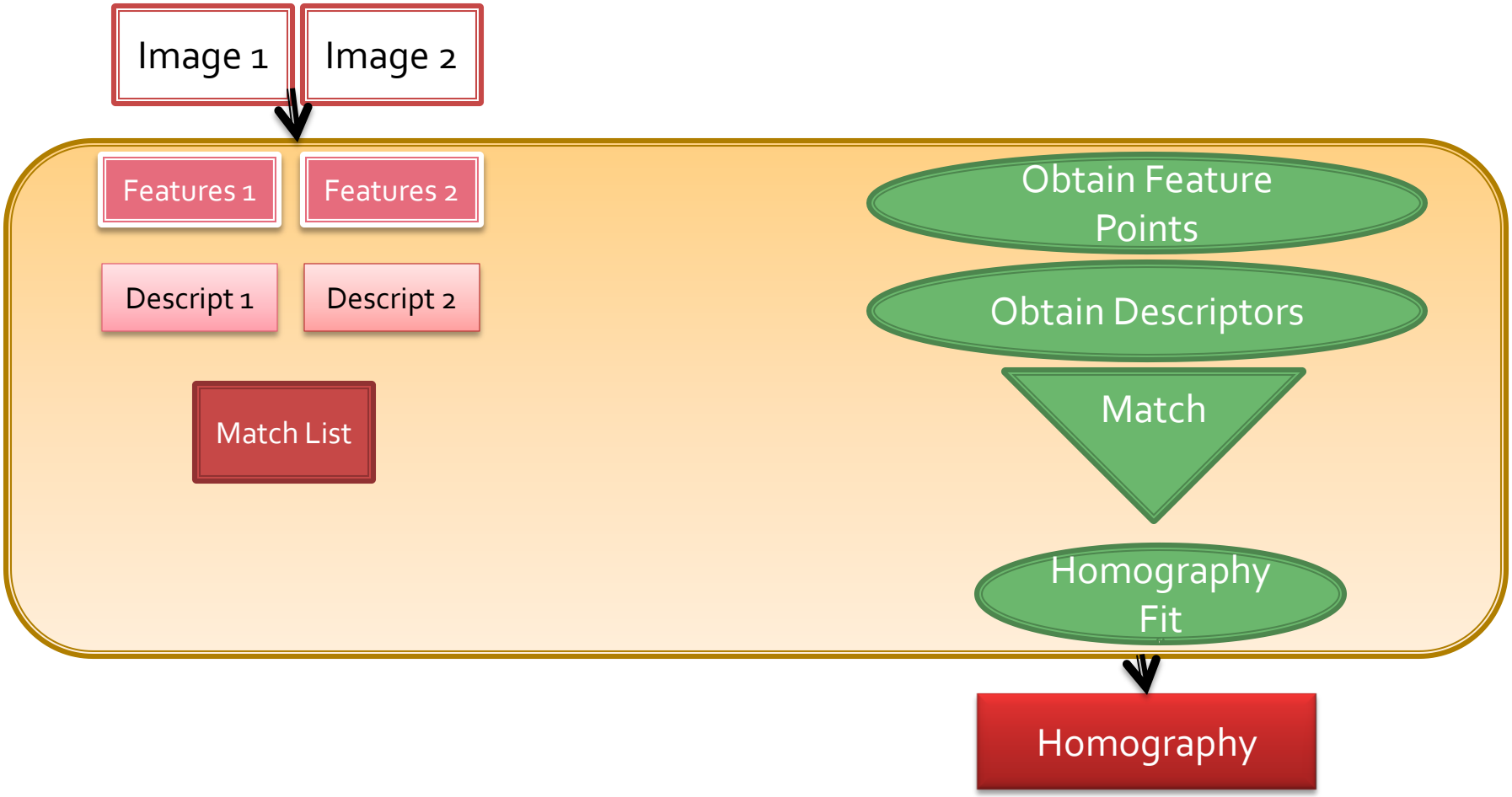
Homography based Tracking

- Definitions
  - What is a planar homography?
    - Point-to-point mapping between different views of a plane
  - What is a feature point?
    - A point with high detail
  - What is a descriptor?
    - A vector that holds local feature information



Homography based Tracking

- Homography Obtaining Process

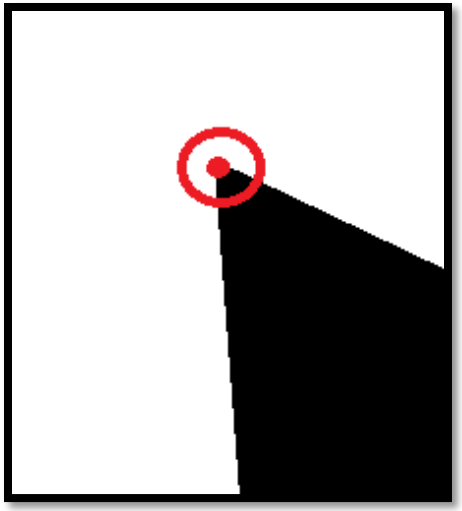


Feature Detectors and Descriptors

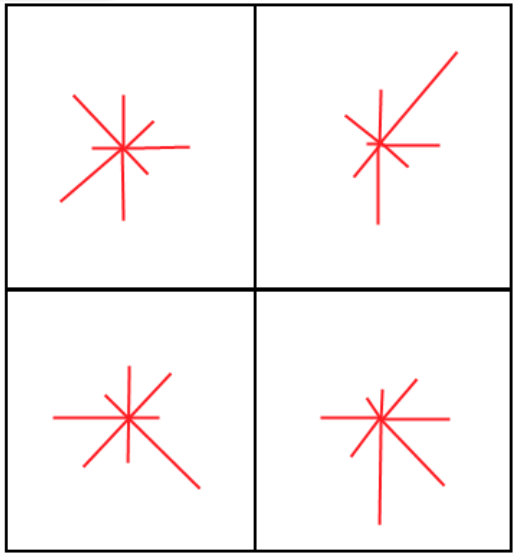
- Selection of these is very important
- Consequences
  - Robustness
  - Computational Efficiency

Feature Detectors and Descriptors

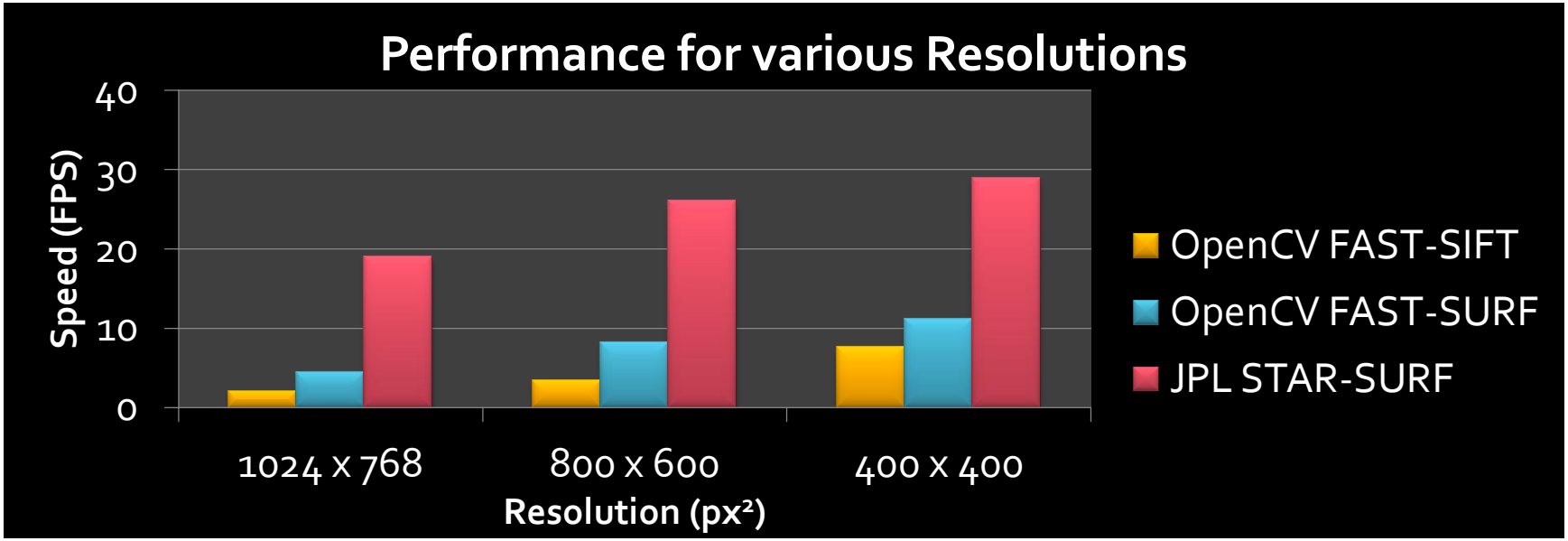
- Detector Options
  - FAST
  - STAR



- Descriptor Options
  - SIFT
  - SURF



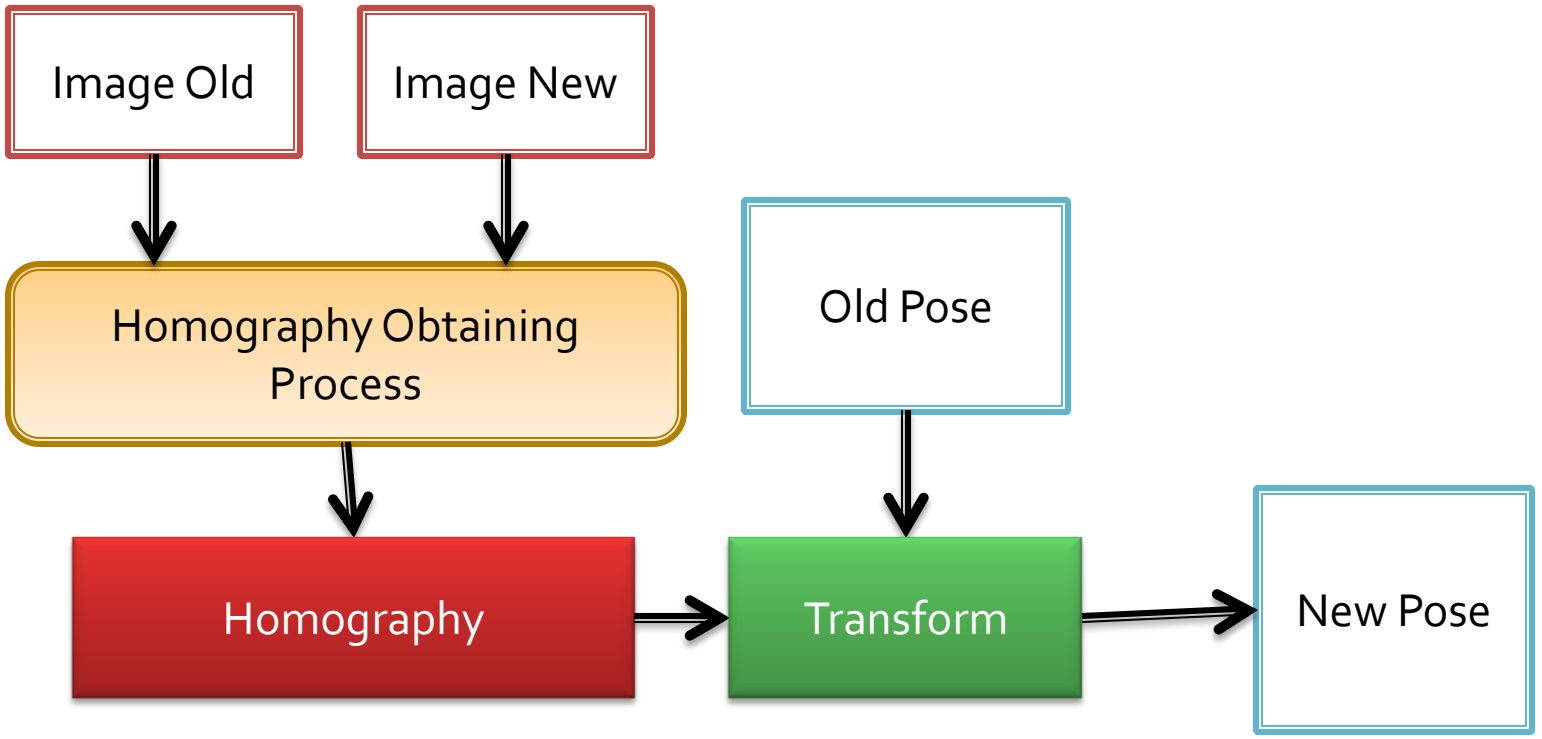
Performance Results



- OpenCV FAST-SIFT
  - Robust but slow
- OpenCV FAST-SIFT
  - Slow and not robust
- JPL STAR-SURF
  - Robust and fast

Homography based Tracking

- Tracking Process



Not so fast…

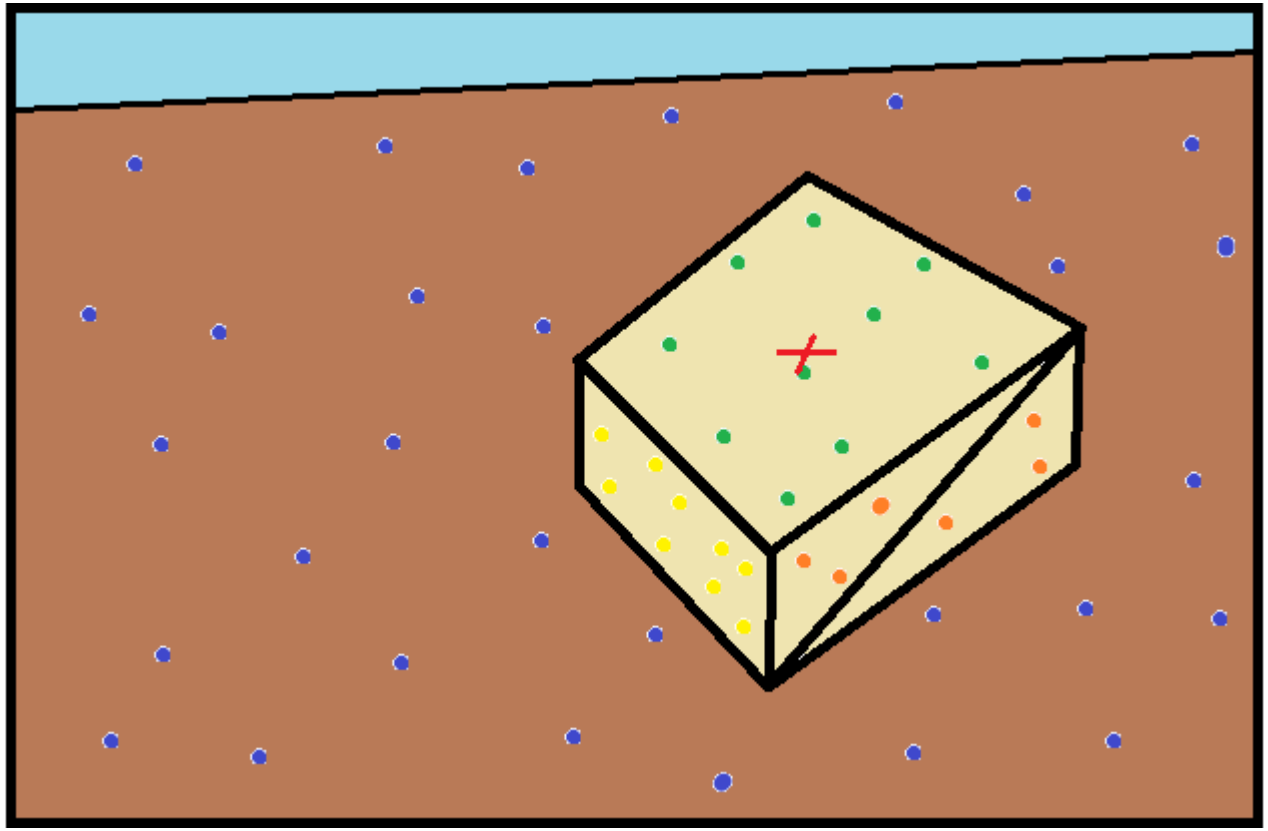
- Problems
  - Scene may have many planes
  - If no filtering of features is done, homography can be wrong
  - Can cause landing location drift
- Solution
  - Find smart way to get rid of outlier features

Methods for Outlier Removal

- RANSAC Homography Calculation
  1. Picks random set of four points and finds homography
  2. Finds number of inliers for found homography
  3. Iterates through 1. and 2. **N** times, picking the homography with the most inliers
  4. Outputs best homography

Methods for Outlier Removal

- But RANSAC is not sufficient

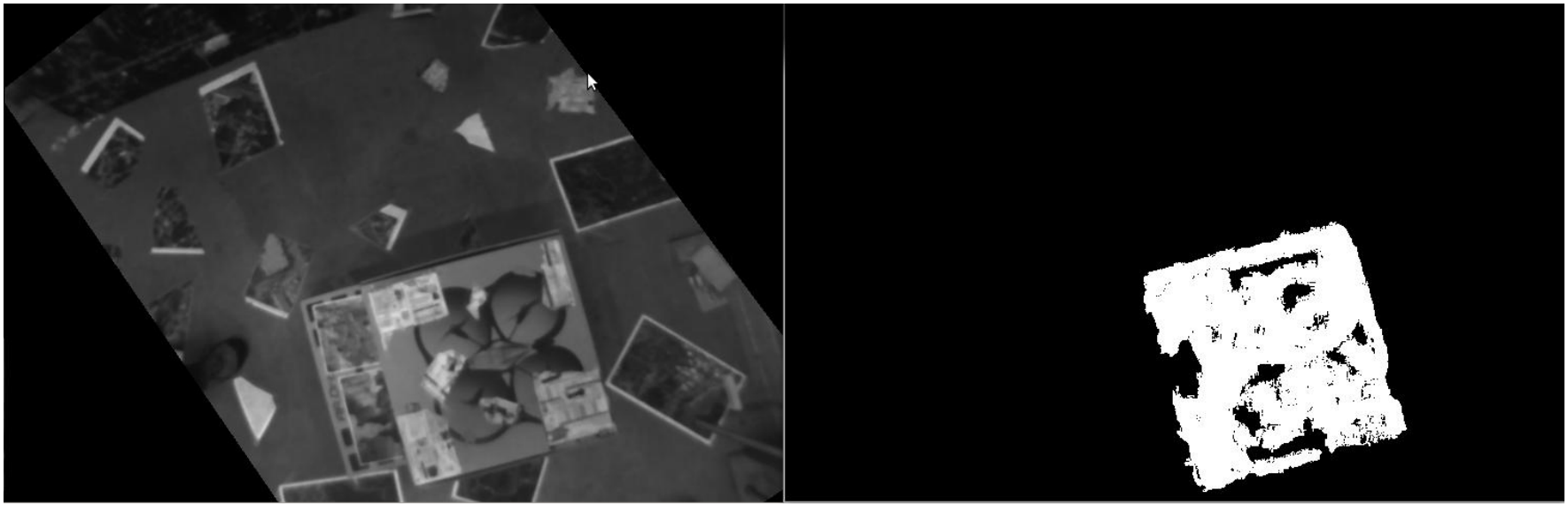


Methods for Outlier Removal

- Filter features using landing plane map
  - Pros
    - Accurate if a dense landing map
  - Cons
    - Map is often full of holes

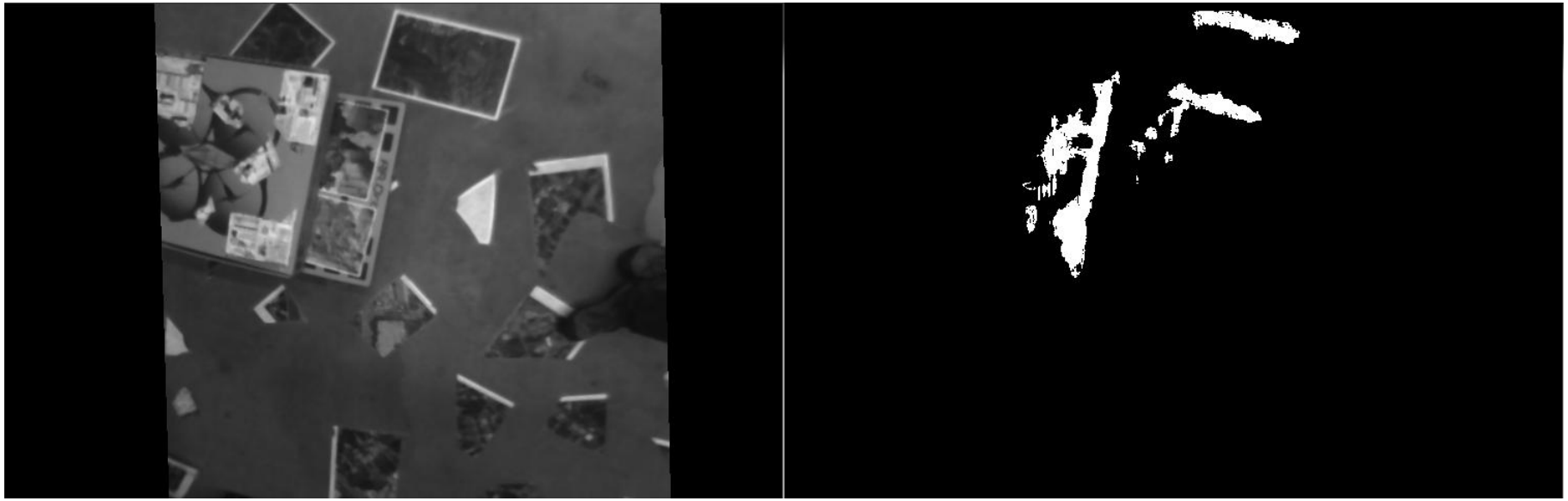
Methods for Outlier Removal

- Good Landing Map



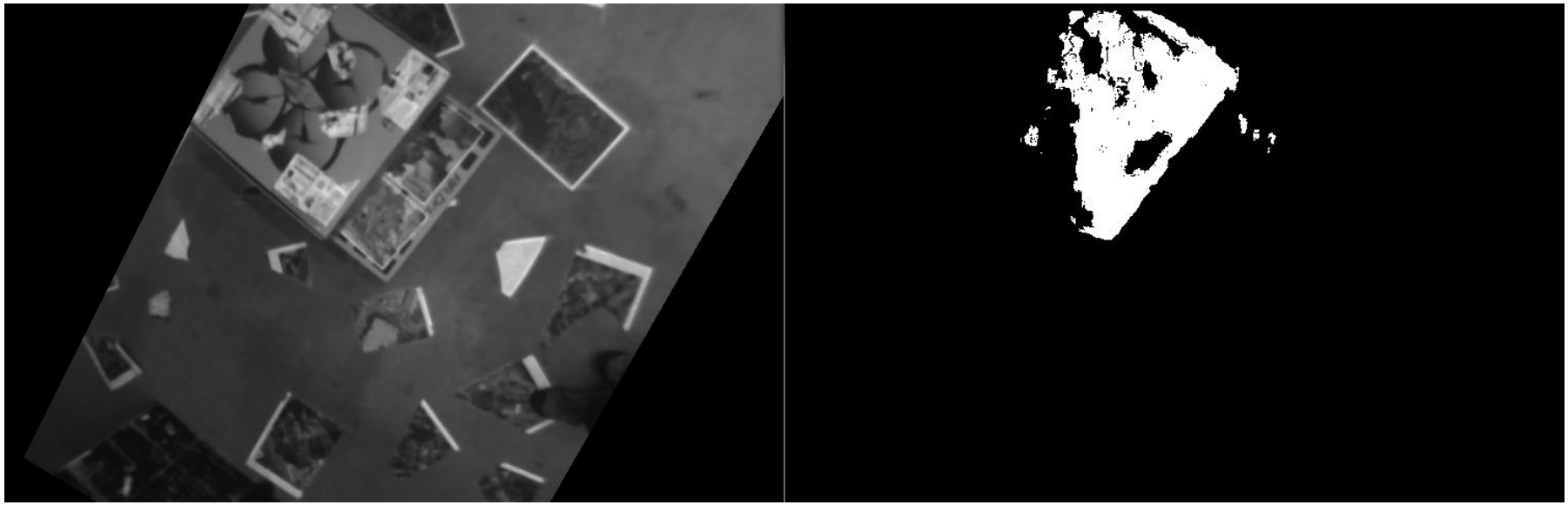
Methods for Outlier Removal

- Bad Landing Map



Methods for Outlier Removal

- Partial Landing Map = Bad Landing Map

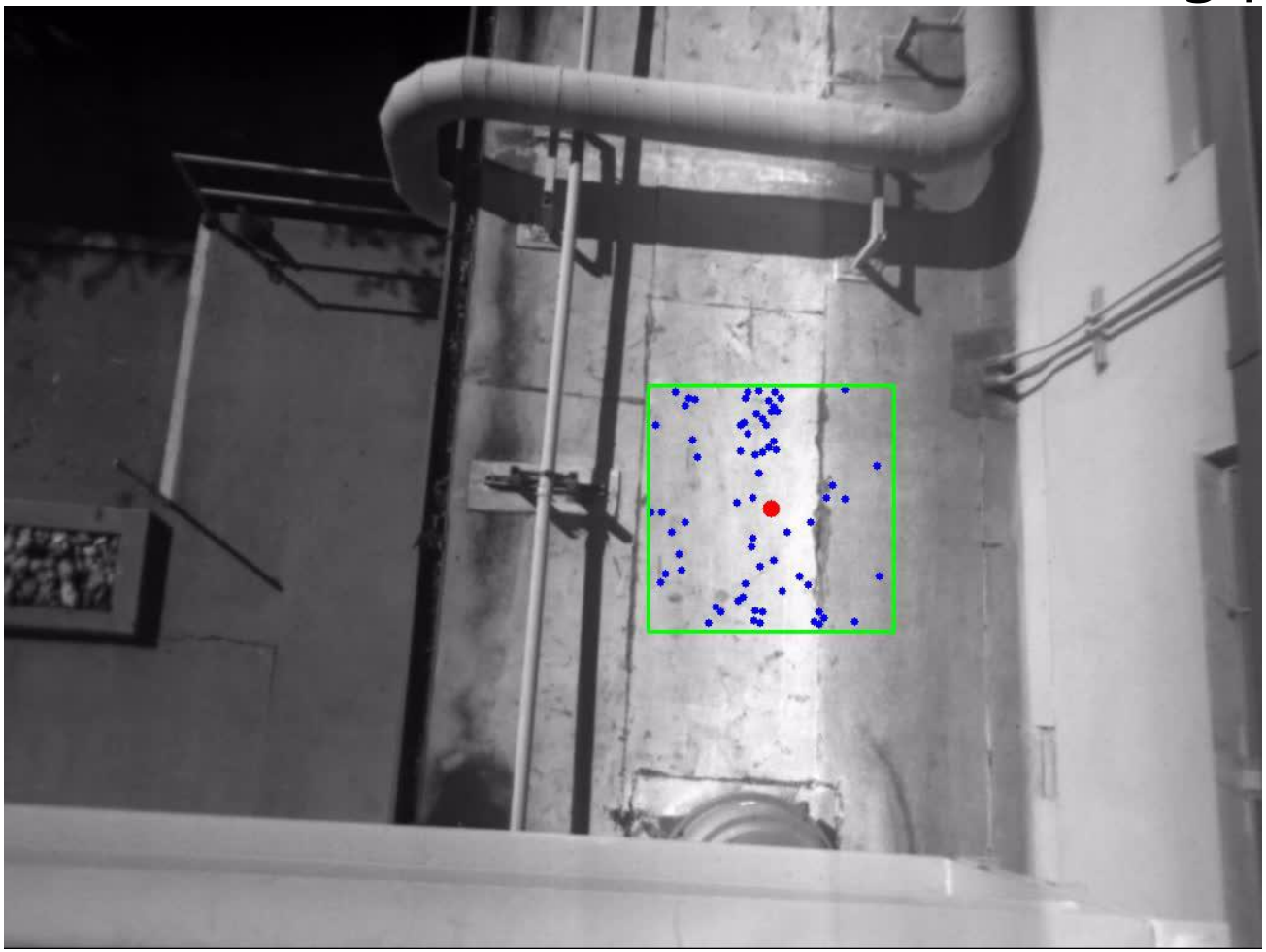


Methods for Outlier Removal

- Use features within ROI around landing point
  - Pros
    - Accurate if enough points present
  - Cons
    - Lack of texture around landing point
    - Not always enough points around ROI
    - Size of ROI depends on height from plane

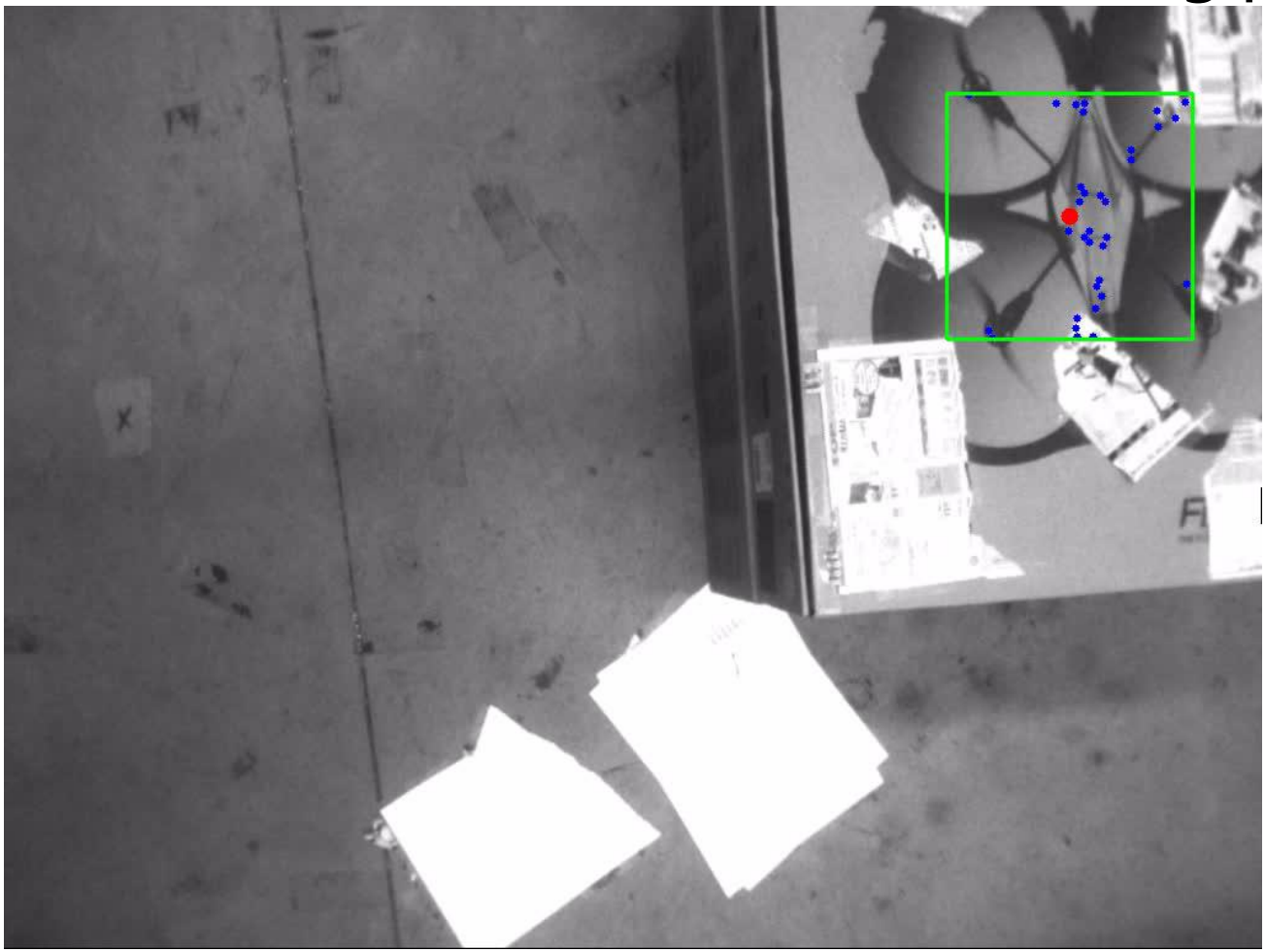
Methods for Outlier Removal

- Use features within ROI around landing point



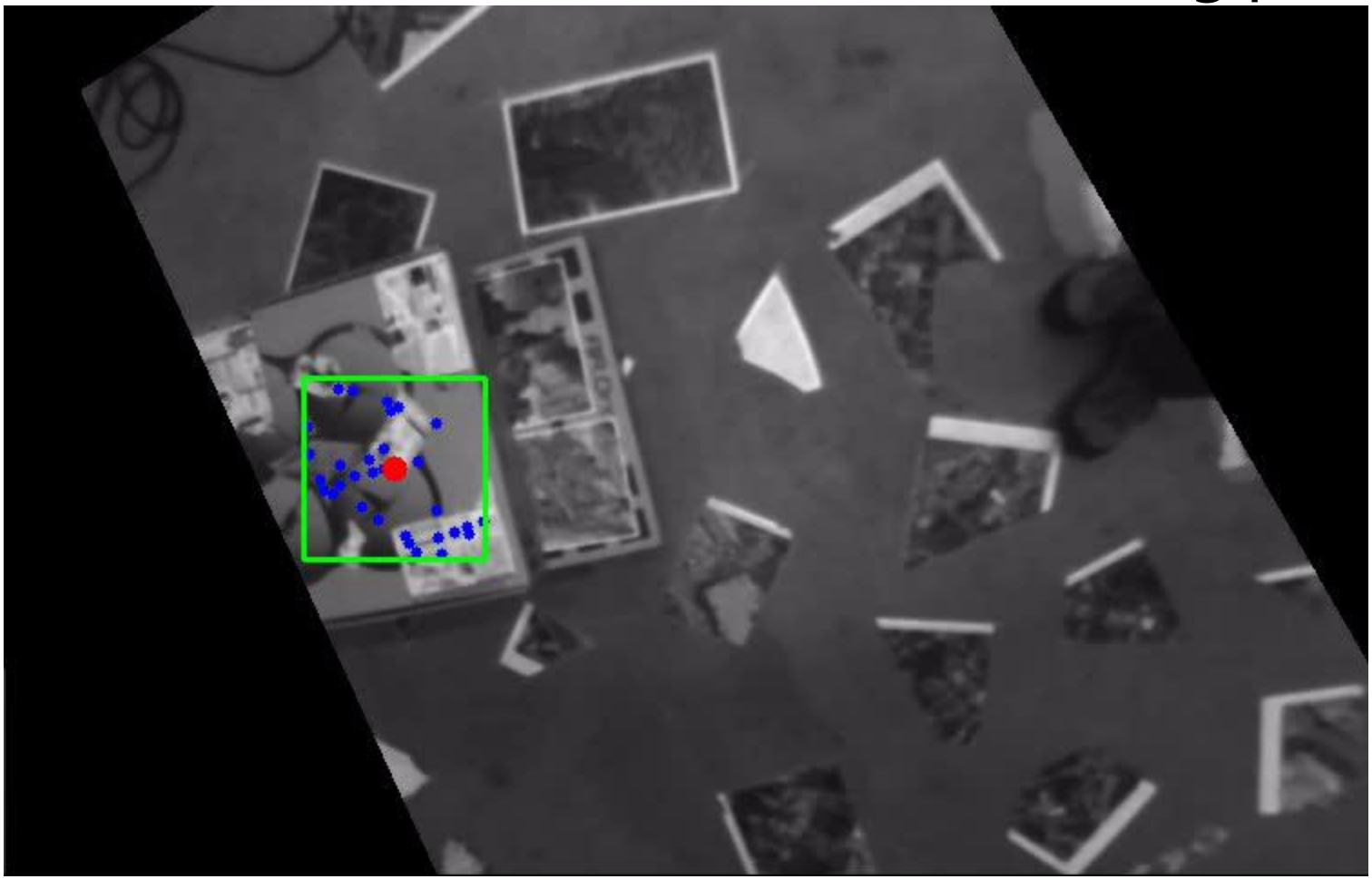
Methods for Outlier Removal

- Use features within ROI around landing point



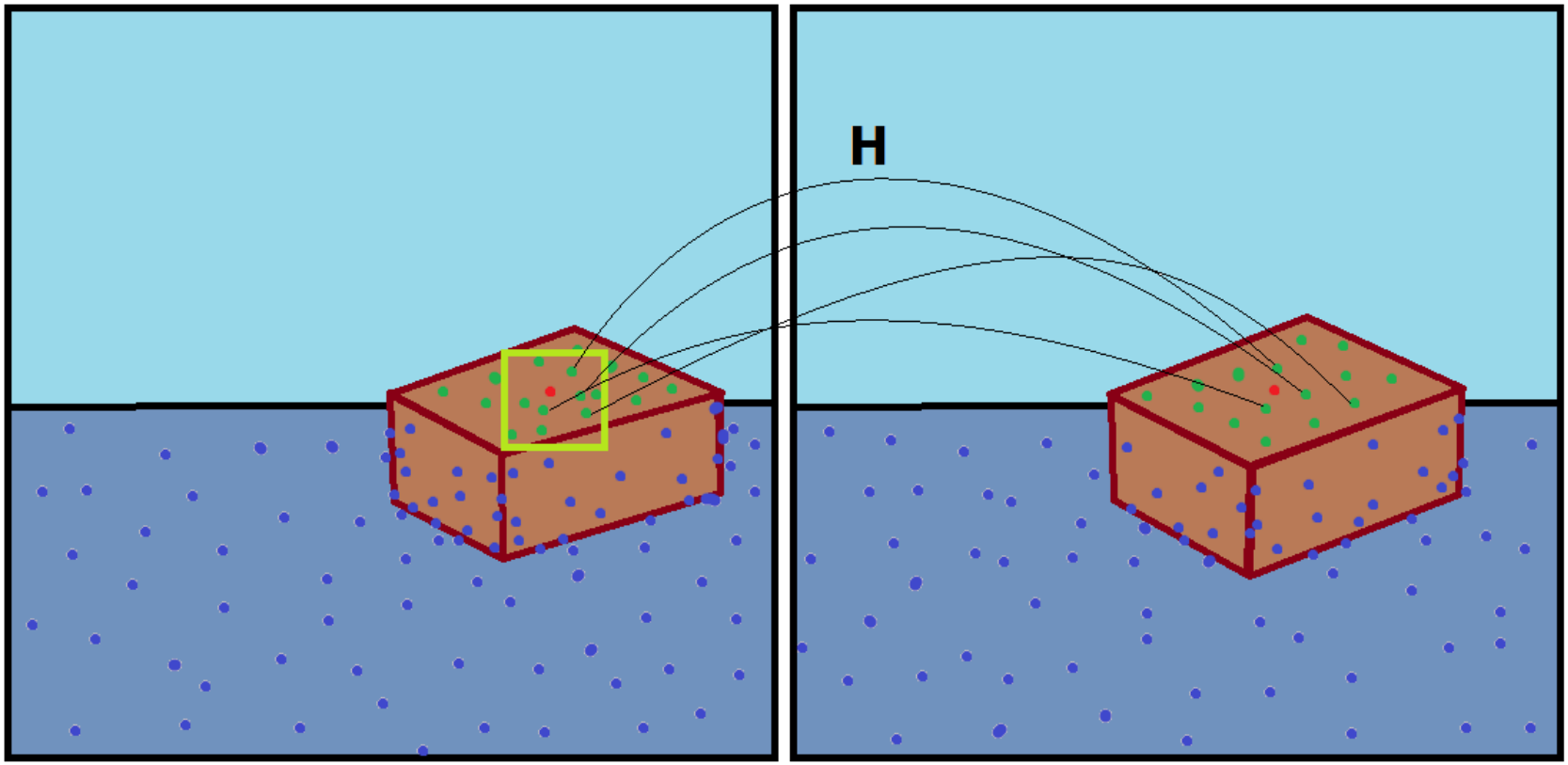
Methods for Outlier Removal

- Use features within ROI around landing point



Methods for Outlier Removal

- Use homography to find features of plane



Methods for Outlier Removal

- Use homography to find features of plane
  - For first image pair, use ROI around landing point to find homography
  - Map old matched points in the image using homography
  - Find error between these mapped values and the new matched points
  - Keep features with very small error

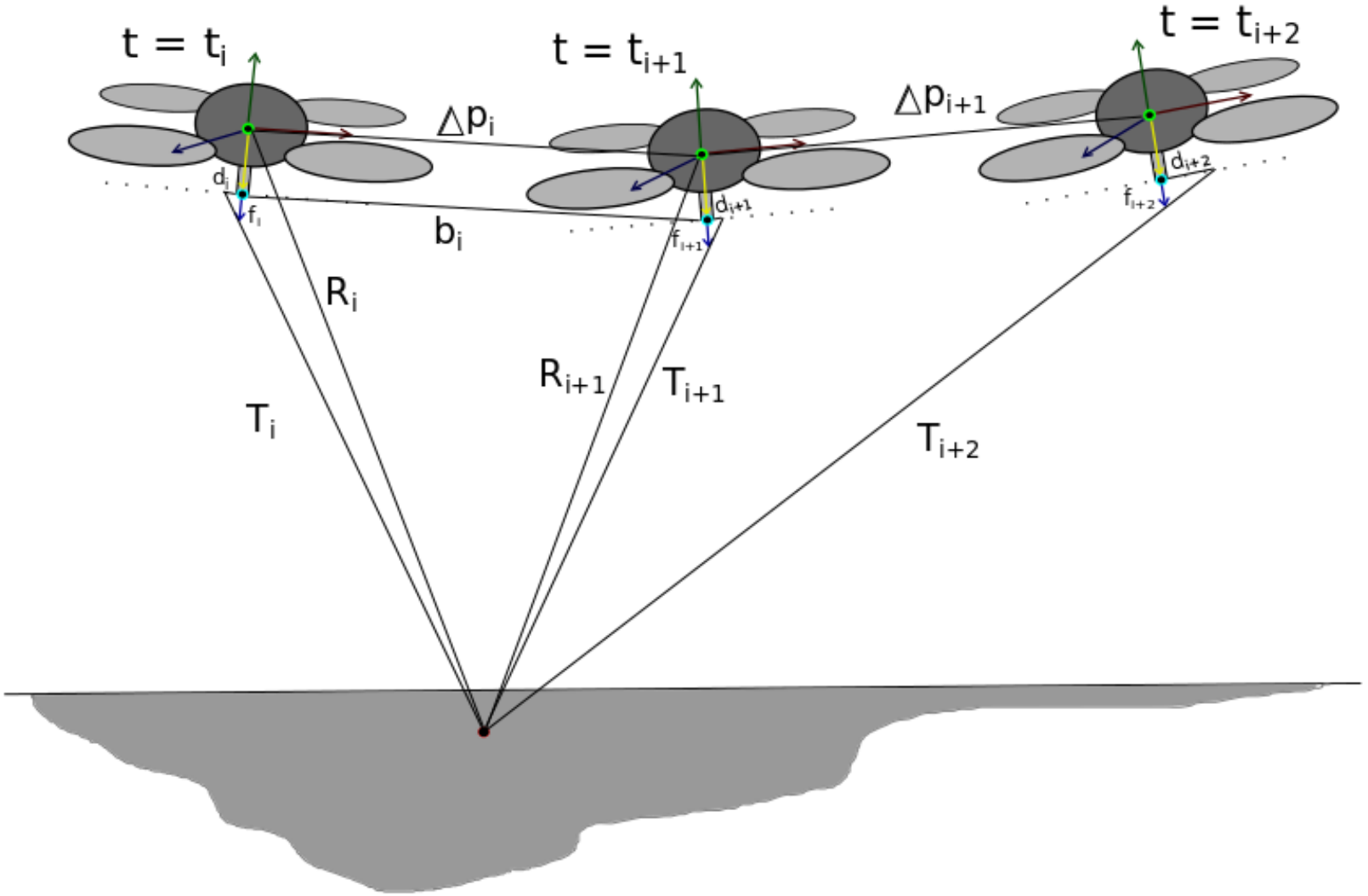
Methods for Outlier Removal

- Initial Results

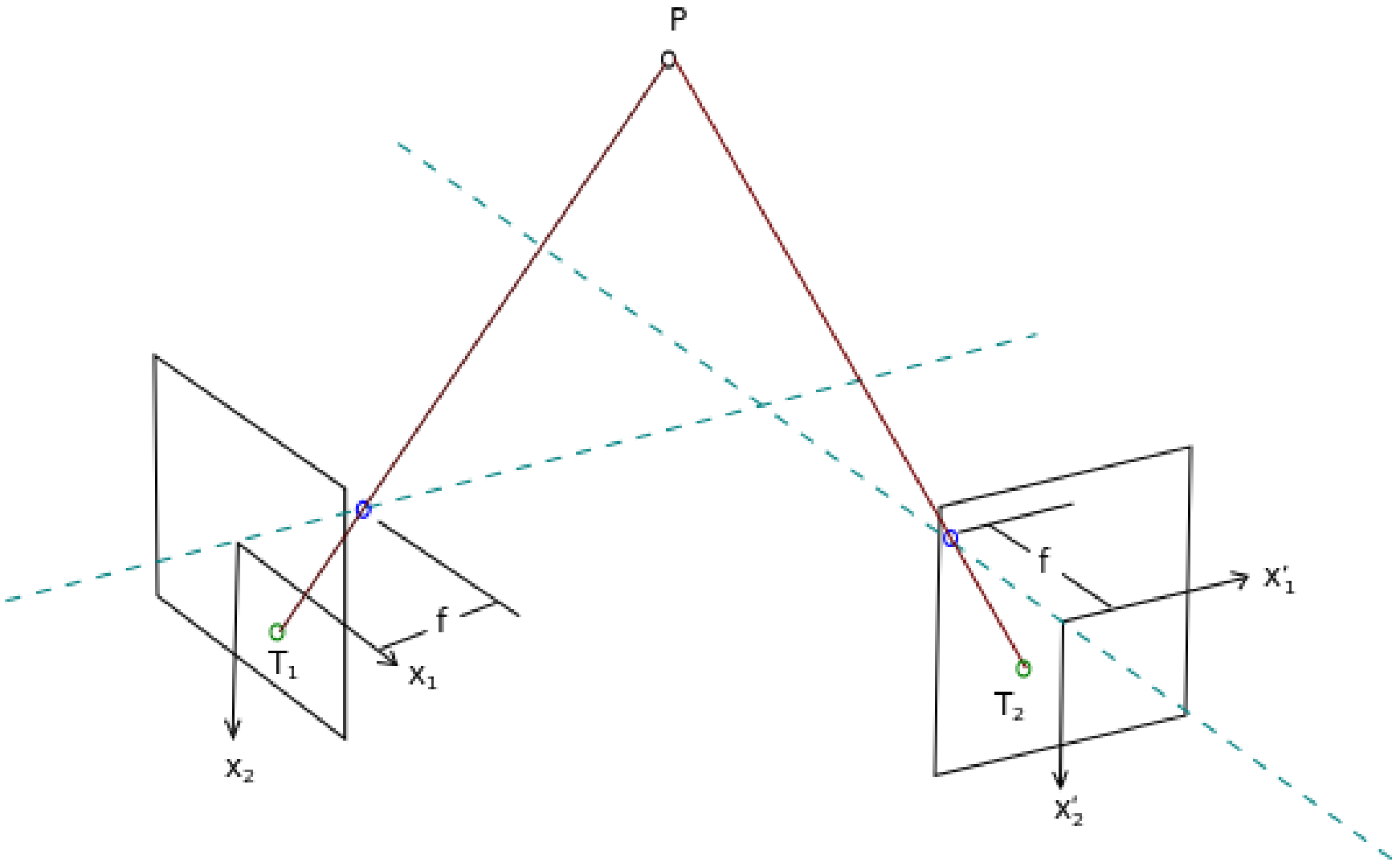
Approach ~ 3-D Vector Generation

- Uses pose estimation and simple pinhole camera model
  - Pros
    - Efficient to calculate
  - Cons
    - Pose estimate is noisy, so error exists

Approach ~ 3-D Vector Generation



Approach ~ 3-D Vector Generation



Summary of Results

- Working tracking code utilizes JPL code for:
  - STAR feature point detection
  - SURF descriptor generation
- OpenCV Library robustness and efficiency not sufficient, using JPL code instead
- Three dimensional vector generation
  - Needs more testing

Future Work

- Finish development of homography based outlier removal
- Test 3-D vector generation software on real data set or quadrotor

Bibliography

- Brockers,R., Susca,S., Zhu, D., Matthies,L., *Fully Self-Contained Vision-Aided Navigation and Landing of a Micro Air Vehicle Independent from External Sensor Inputs*, Proc. Of SPIE 8387, 2012.
- Brockers, R. *Feature Detection and Matching – ME/CS 132a Lecture Notes*, 2013.
- Brockers, R., *Stereo Vision – ME/CS 132a Lecture Notes*, 2013
- Cheng, Y., *Tracking and Outlier Detection – ME/CS 132a Lecture Notes*, 2013
- Szeliski, R., *Computer Vision: Algorithms and Applications*, September 3, 2010
- http://en.wikipedia.org/, *Pinhole Camera Model*, 2013
- http://en.wikipedia.org/, *Triangulation (Computer Vision)*, 2013
- Belongie,S., UCSD, *CSE 252B: Planar Scenes and Homography*, 2013
- Collins, R., *CSE 486 Lecture 16: Planar Homographies*, 2012.

Thanks

- Roland Brockers
- Martin Humenberger
- NASA USRP
- Jet Propulsion Lab & Caltech
- Intern co-workers

Questions?