Marker Detection Failure Recovery

Project Plan

Abstract

Fiducial markers are commonly used in robotics for providing pose estimates. Their simplicity and known reliability make them popular in applications such as industrial robotics and space robotics. Classical computer vision algorithms used to detect these markers are subject to failure due to a variety of faults. These faults include the markers being partially out of view or occluded; the image being blurry, too dark, or too light; or the environment causing shadowing, glint, or glare. This paper presents a methodology to: (1) detect the marker under the previously specified faults; (2) estimate the pose of the marker; (3) classify the cause of failure of the classical computer vision algorithm; and lastly, (4) compute a closed-loop corrective action consisting of relative movement of the camera and adjustment in lighting from the robot to allow the marker to be detectable using the classical computer vision algorithm.

Methodology

1. Simulation Environment and Parameters

The marker detection, pose estimation model, and fault diagnosis model discussed later are both learning-based models, both of which will require on the order of millions of images. Thus, simulation will be used to generate synthetic images with parameterized control over the environmental conditions. Isaac Sim will be used to generate synthetic images and excite faults within the images. The Isaac Sim environment will consist of a planar fiducial marker with a randomized background, light sources, and objects. An ISAM environment will also be developed in Isaac Sim – this environment will only be used for testing and not training to avoid overfitting the model to the ISAM environment and to demonstrate generalization and robustness of the model trained on only randomized backgrounds. Lastly, the corrective action will be tested in a separate Isaac Sim environment in which a camera is mounted to a robotic arm with an illuminator. In this environment, a controller receiving feedback from the pose estimation model and fault diagnosis model will provide command signals to move the camera and change the brightness of the illuminator. This environment will be used to test the controller's ability to recovery from a failure of the classical computer vision based marker detection. The following table describes the adverse environment conditions to be studied (in priority order) as well as the corresponding simulation parameterized variables used to excite the faults.

Table 1: Marker detection faults and simulation variables to excite the faults

Fault Category	Fault	Simulation Variable
Pose	Not all corners in field of view	Camera pose
	Highly skewed angle	Camera pose
	Too close to view all corners	Camera pose
	To far to detect edges	Camera pose
Ambient Conditions	High/low brightness	Light source intensity
	Shadows	Objects between light source and marker (may or may not be within view)
	Background noise	Background images (Final: test with ISAM backgrounds)
Stray Light	External light source (Sun, Earth albedo)	Model Sun/Earth albedo light source, with varied direction, intensity
	Glare (large bright regions of reflected light)	Objects with varied shapes, poses, patterns, and optical properties
	Glint (small concentrated point of reflected light)	Objects with varied shapes, poses, patterns, and optical properties
	Lens flare (TBD if within scope)	Isaac Sim lens flare model
Blur	Focus blur (TBD if within scope)	Isaac Sim image blur model
	Motion blur (TBD if within scope)	Isaac Sim motion blur model

2. Marker Detection Model

A U-Net architecture will be used to perform segmentation of an image to classify each pixel as "marker" or "no marker". This model will be trained using only simulation data from the previously described Isaac Sim.

Figure 1: Input image with marker (left) and segmented image (right). The white pixels of the segmented image mark the pixel labeled as "marker."

3. Pose Estimation Model

A separate learning-based model will be used to perform marker pose estimation. This model will be trained using the same synthetic data previously described. The architecture of this model is TBD but its design will begin with open-source models used for similar tasks.

4. Fault Diagnosis Model

A separate U-Net model will be used to perform classification of the fault of the image. This model will output a probability distribution across all the specified faults listed in Table 1 as well as a “no fault detected” class. This model will first be trained to diagnose singular faults. Multiple simultaneous faults will likely fall out of scope of this work, but may be investigated in follow-up work.

5. Recovery Action

The recovery action consists of two types of actions – change of pose and change of illumination. If the detected fault is within the pose fault category, the recovery action is to perform visual servoing using the pose estimation model. If the detected fault is within the ambient conditions or stray light fault categories, the recovery action will consist of brightening or dimming the illuminator to compensate for the fault. The goal of this work is to demonstrate empirical convergence of failure recovery. Analytical proof of convergence will be out of scope of this work but may be investigated in follow-up work. Recovering from blur faults will be out of scope of this work.

6. Experiment

The marker detection, pose estimation, and fault diagnosis models will be validated in a real world environment where faults are independently excited. Each model will be evaluated on its accuracy in respectively performing detection, pose estimation, and fault diagnosis. Additionally, the covariance of the pose estimate will be empirically determined.

Project Schedule

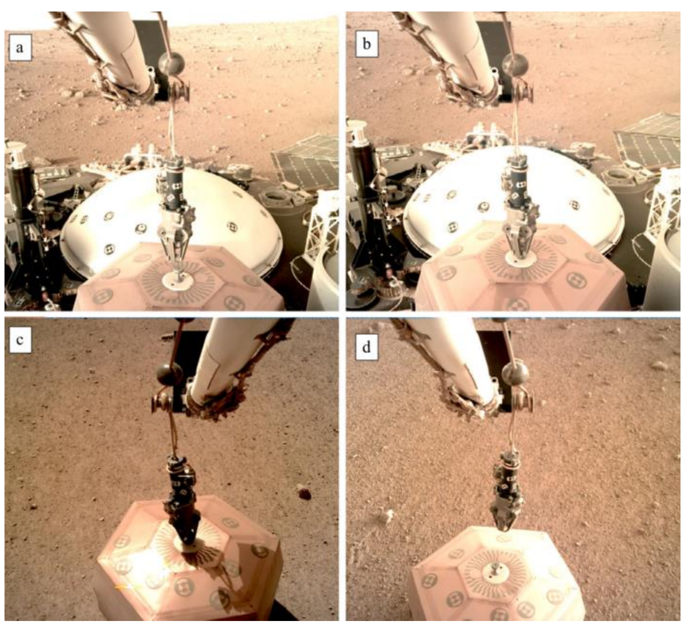
Category	**Tasks**	**Target Completion Date**
Simulation Environment	Generate images with varying poses and backgrounds	Complete
	Generate images with varying lighting conditions	Feb 3, 2025
	Generate images with varying objects and blur	Feb 3, 2025
	Create ISAM environment	Mar 3, 2025
	Dynamic environment with robotic arm and illuminators	Mar 31, 2025

Marker Detection Model	Model trained on varying poses and backgrounds	Complete
	Model trained on varying poses, backgrounds, lighting conditions, objects, and blur	Feb 10, 2025
Pose Estimation Model	Design model architecture	Feb 17, 2025
	Model trained on varying poses, backgrounds, lighting conditions, objects, and blur	Feb 24, 2025
Fault Diagnosis Model	Automate ground truth classification	March 10, 2025
	Model trained on varying poses, backgrounds, lighting conditions, objects, and blur	March 31, 2025
Recovery Action	Pose recovery action controller	April 7, 2025
	Illumination recovery action controller	April 21, 2025
Experiments	Marker detection, pose estimation, and fault diagnosis models testing	May 5, 2025
	Recovery action testing	May 19, 2025

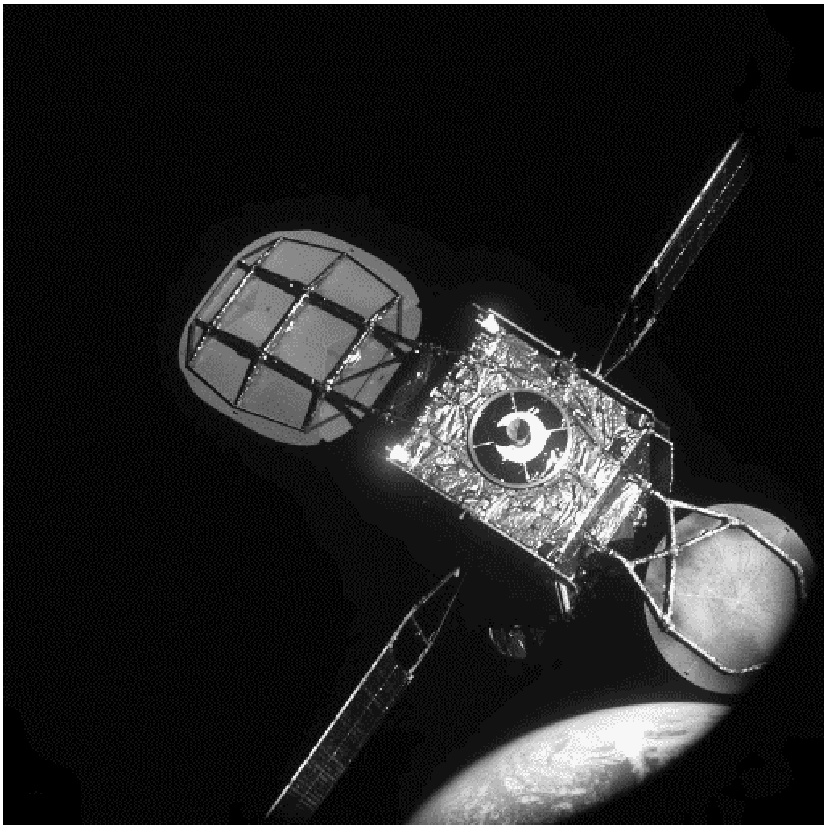
Notes:

- Project schedule is tentative and depends on workload of USC research
- This method primarily studies how machine learning can be used to provide dissimilar redundancy when classical computer vision fails under adverse conditions. Future work will study the failure modes of the machine learning method itself.
- Target forum for publication is TBD. Two possibilities are RA-L (no fixed deadline) and ICRA (September 2025).

Example Images



Source: https://link.springer.com/article/10.1007/s11214-023-00964-0



Source: https://www.intelsat.com/resources/blog/in-orbit-mission-success-extending-the-life-of-intelsat-10-02/