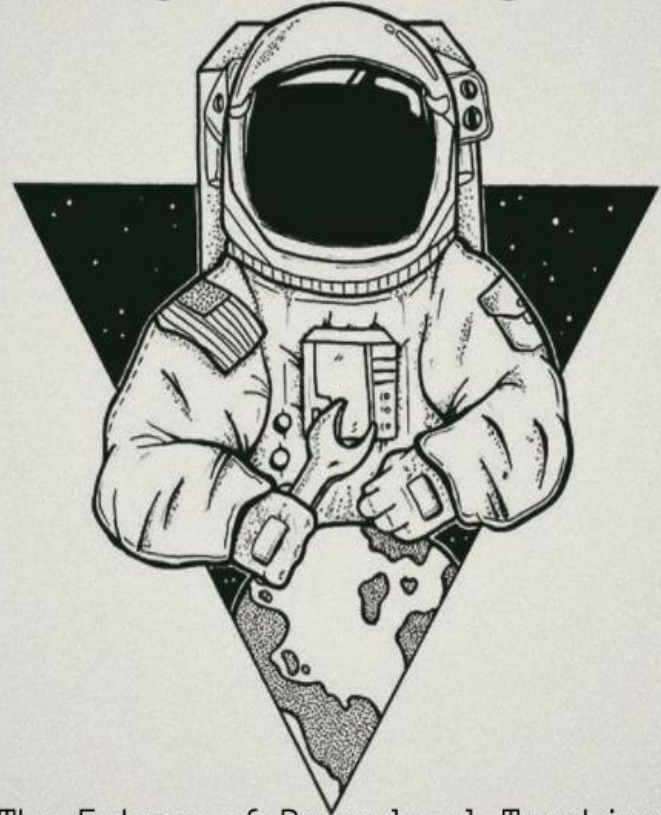


Project Argus

Final Presentation



Project Argus



The Future of Procedural Tracking

The Team

- Naimul Hoque



Team Leader

- Rishit Arora



Wireless Interface Design

- Abel Semma



User Interface Design

- Oles Bober



Model Training/Procedure Design

- Edwin Varela



Model Training/Procedure Design



Panoptes Argus

- Greek Mythology
- Argus Panoptes a Primordial Giant and Guardian
- Is described to be “all seeing” with multiple eyes
- “Followed by the Eyes of Argus”

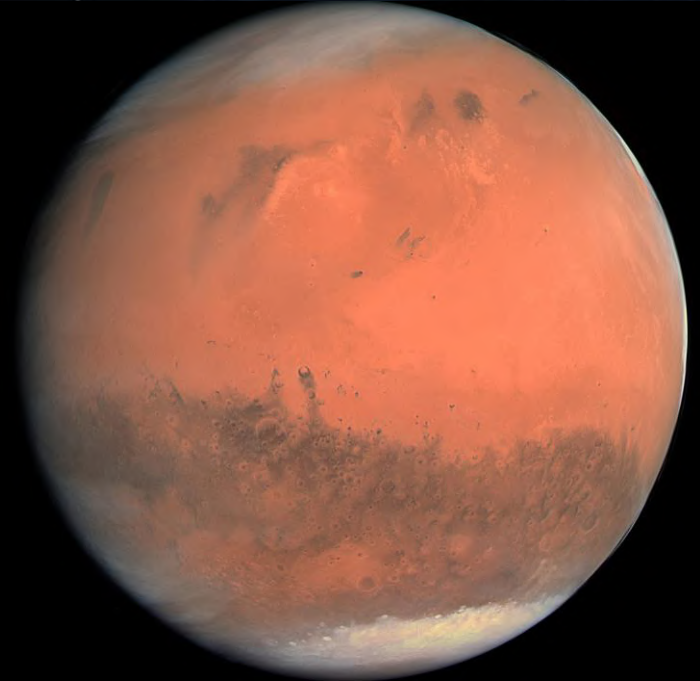


"And set a watcher upon her, great and strong Argus, who with four eyes looks every way. And the goddess stirred in him unwearying strength: sleep never fell upon his eyes; but he kept sure watch always".



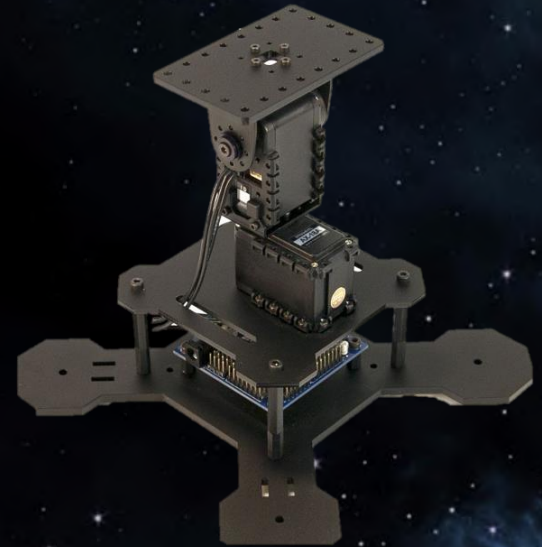
Problem Statement

- **Astronauts perform complicated procedures**
- **Assistance from ground control is required**
- **Major issue for future space exploration missions**
- **Communication delay and bandwidth limitations**



Solution: Procedural Tracking

- **Using Computer Vision and Machine Learning**
- **Tracking and logging procedures**
- **Assembling a Trossen PhantomX Robot Turret Kit**
- **Similar to procedures performed on the ISS**



Nvidia Jetson Nano

- **128-Core Maxwell GPU**
- **Quad-Core ARM A57 CPU**
- **4x USB 3.0, USB 2.0 Micro-USB**
- **Computer Vision/Machine Learning Applications**
- **Coded in Python**
- **Connects both cameras using CSI and WiFi**



Cameras

- **Raspberry Pi Camera**
 - **Sony IMX219 8-megapixel sensor**
 - **Capture video at 1080p30 resolution**
 - **Communicates using CSI Port**

- **AKASO EK7000**
 - **Uses RTSP Protocol to stream**
 - **Python is used for Jetson/AKASO communication**



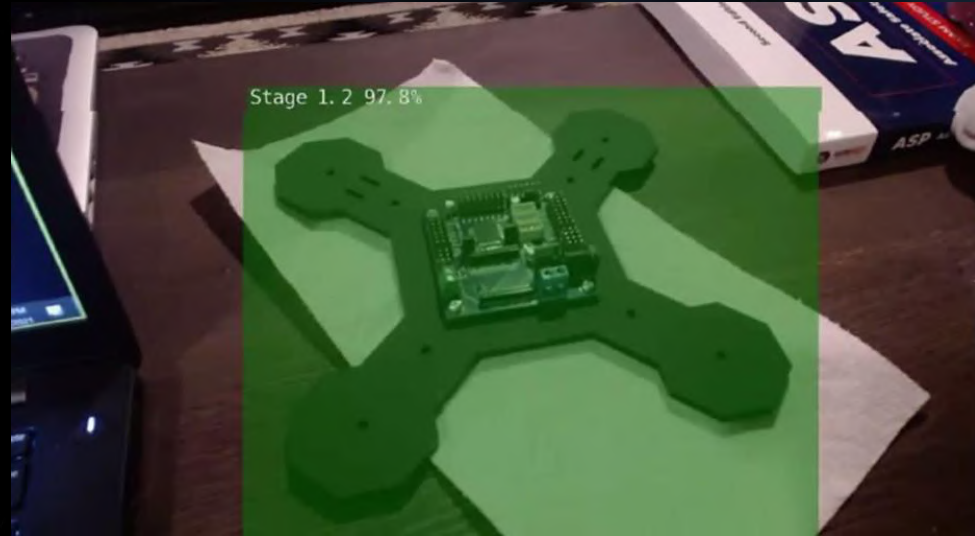
Workstation

- Workstation will be a flat surface with (preferably) white background, few to no objects
- Two camera options available: wireless on the user's body or wired on the workstation



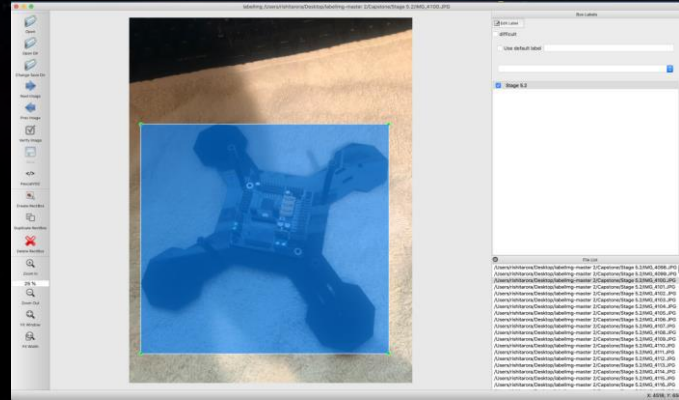
Design (Tracking)

- Procedure split into stages
- Detection done using jetson-inference
- Two object detection models
- Procedure moves on to next stage when the current stage is properly validated



Design (Object Detection Training)

- Both models were trained using jetson-inference
- Hundreds of images of each stage, part
- Bounding boxes with labellmg
- Google Colab was used for training to speed up the process



Design (Data Collection)

- Various data is collected during procedure
- Two .csv files exported at the end
 - **Stages.csv**
 - **Parts.csv**

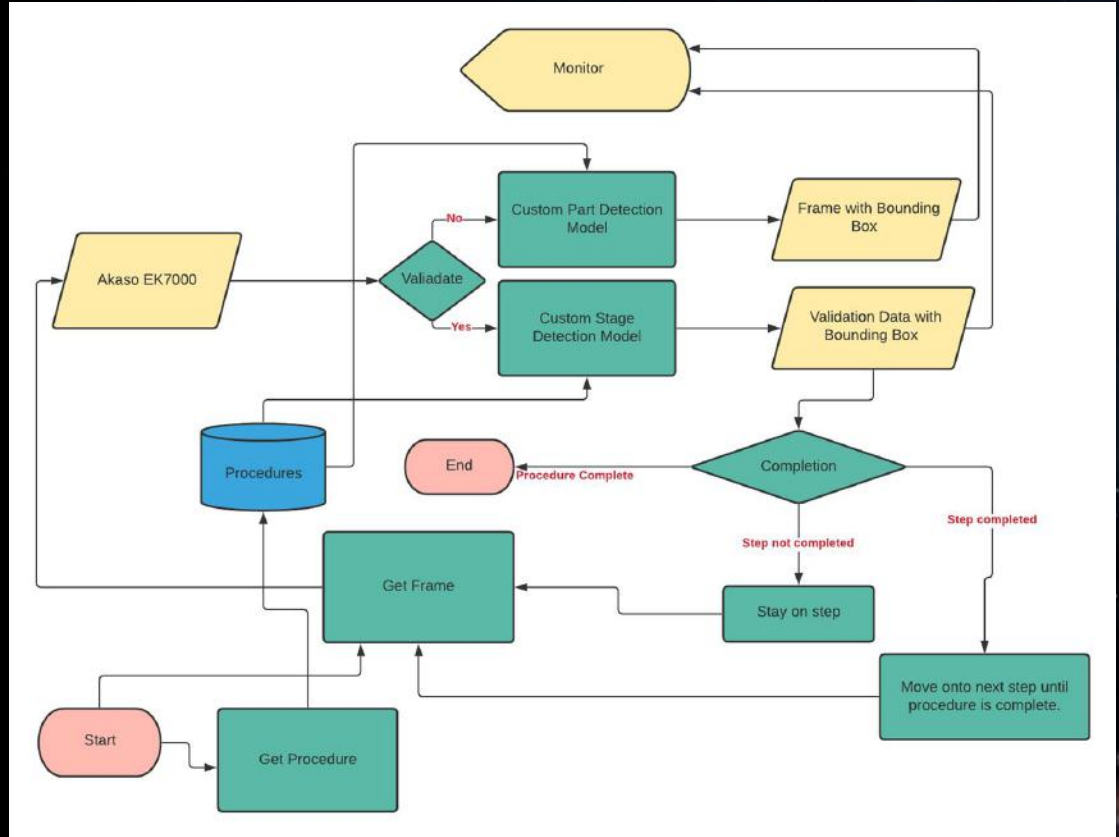


Design (Wireless Camera Interface)

- **WiFi Interface was accessed using Python CV2**
- **Connect to AKASO WiFi network on the Jetson**
 - **Each AKASO camera has a unique IP Address for camera feed access**

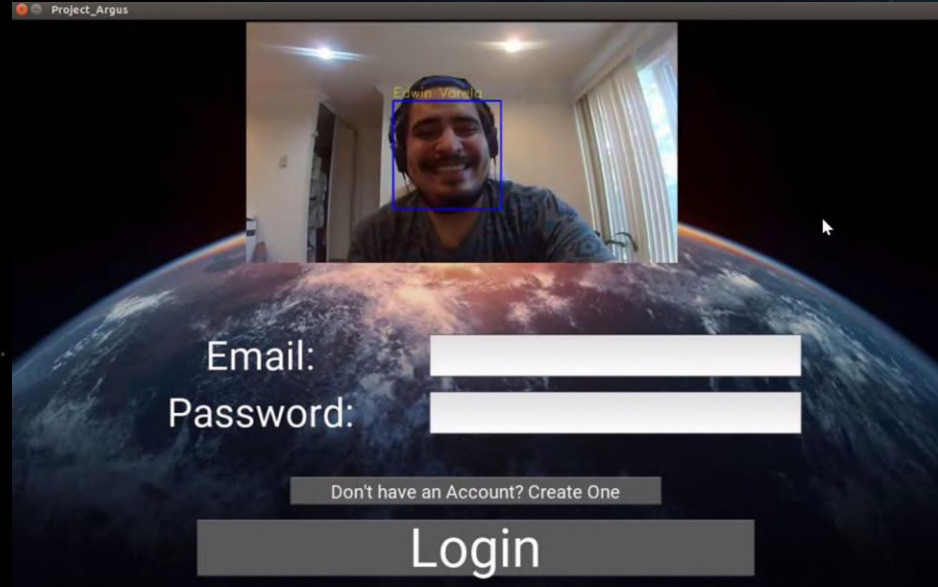


Software Flow



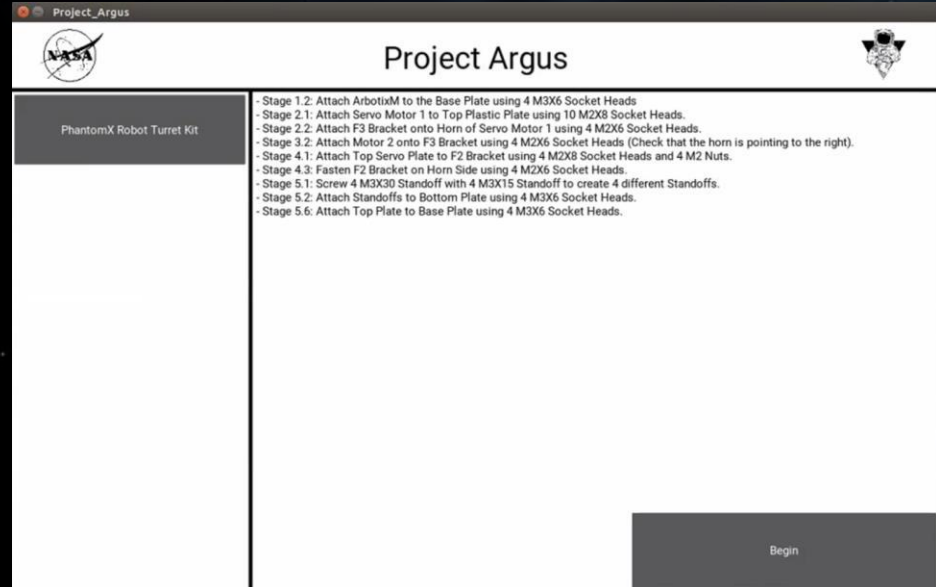
User Interface (Login Screen)

- User Interface was designed using KivyMD
- First screen allows user to login, create account
- Face detection login



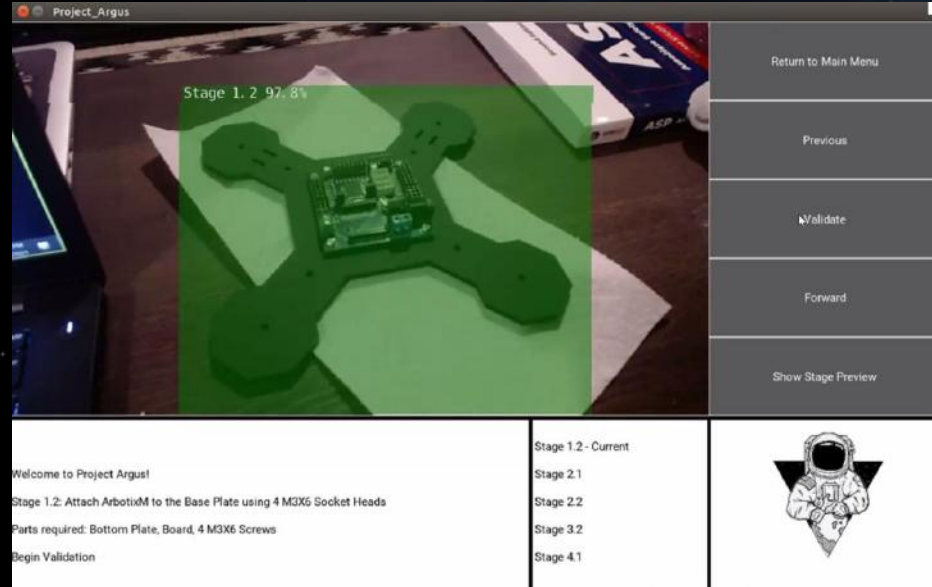
User Interface (Main Menu)

- List of procedures
- When clicked, list of instructions for procedure appear



User Interface (Procedure Screen)

- **Top left: camera with bounding boxes**
- **Top right: various buttons**
 - **Return to Main Menu**
 - **Previous**
 - **Validate**
 - **Forward**
 - **Show Stage Preview**
- **Bottom left: text output**
- **Bottom middle: checklist**



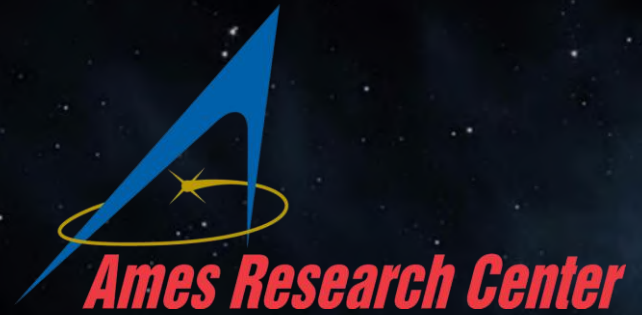
Acknowledgements

A huge thank you to:

- Dr. Yogananda Irukupalli
- John Karasinski
- Melodie Yashar
- Boning Dong
- Trenton Rochelle

Without the advice, knowledge, and inspiration from each and every one of you, we wouldn't have been able to complete this project!

An additional thanks to UCSB for making this opportunity possible!



Questions?

