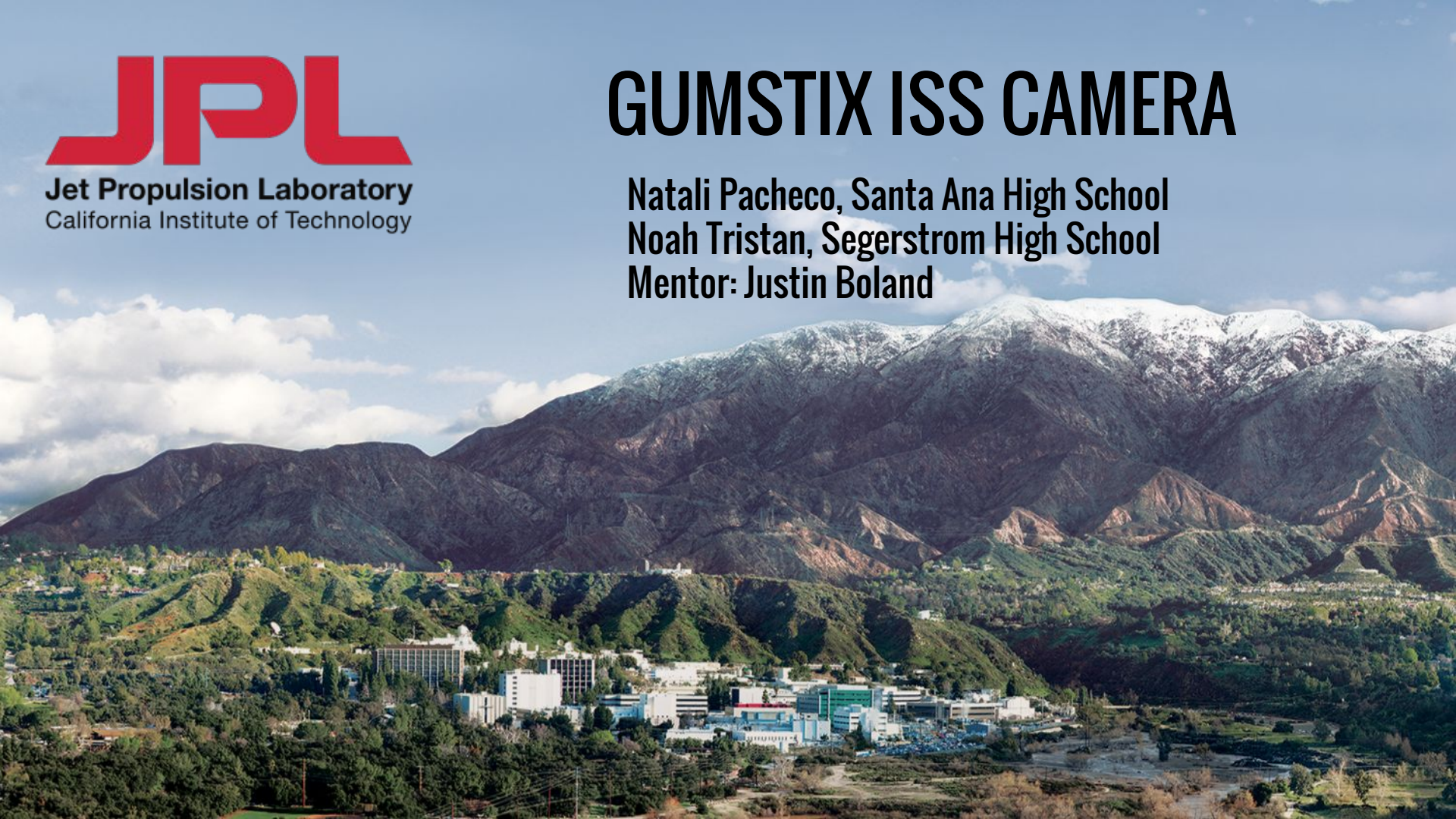
Jet Propulsion Laboratory
California Institute of Technology

GUMSTIX ISS CAMERA

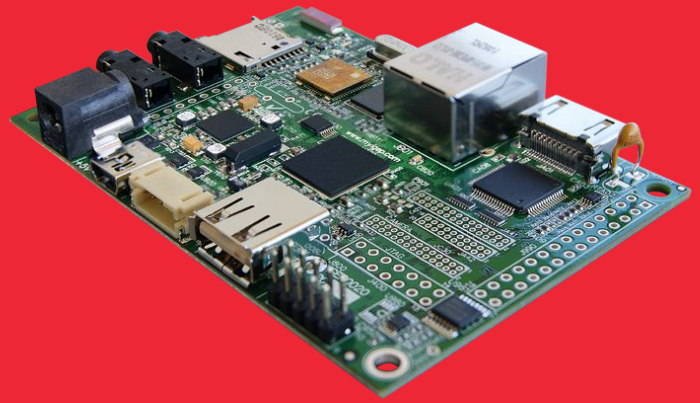
Natali Pacheco, Santa Ana High School
Noah Tristan, Segerstrom High School
Mentor: Justin Boland



ABSTRACT

With many jobs that are considered to be too meticulous for the astronauts on the International Space Station, our commercial camera aims to eliminate some of those tasks. Debris and other metal are often found falling away from the station, and this could signal a repair to be made. These cameras, programmed with OpenCV motion detection software, would be able to recognize such debris and relay a live feed to operators either on the station or back on Earth.

OVERVIEW



- Project split into two, 2- week phases
  - Optional 3rd phase for improvements

METHODS

Phase 1 **Duration: 4 weeks**

Objective

- Install Linux distribution Poky on SD card # 8
- Replace Angstrom distribution with Poky on SD card # 2
- Test out SD cards #1, 3, and 6
- Communicate with Gumstix cameras

Results

- Unable to download Poky onto SD card #8
- After attempting to download Poky on SD card #2, moved forward with Angstrom.
- Two working Gumstix cameras with communication; Boards #1 and 3
- Able to connect with SD cards #2 and 8; SD cards #1, 3, and 6 would not open.

Phase 2 **Duration: 3 Weeks**

Objective

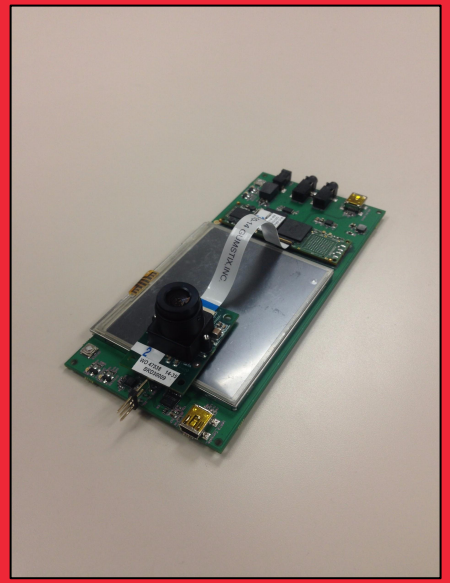
- Install OpenCV onto Gumstix cameras
- Program OpenCV to work with Gumstix boards
- Obtain image feed
- Program motion detection onto Gumstix

Results

- Installed OpenCV onto Board #1; Board #3 became unresponsive (error in terminal, no connection)
- Working solely on board #1, used tutorial code and created a build program to adapt OpenCV to Gumstix.
- Attempted to install various motion detection programs onto Gumstix

RESULTS

- Connection to Gumstix boards with two working micro-SDs
- Working OpenCV code that allows a user to access a specific camera/cameras
- Able to open camera on Mac with working code



WORKING CODE

ntristan@ubuntu: ~/Desktop/camera

GNU nano 2.5.3 File: camera.cpp

```
#include <iostream>
#include <cv.h>
#include <highgui.h>

using namespace std;
char key;
int main()
{
    cvNamedWindow("Camera_Output", 1);    //Create window
    CvCapture* capture = cvCaptureFromCAM(CV_CAP_ANY);  //Capture using any camera connected to your system
    while(1){ //Create infinte loop for live streaming

        IplImage* frame = cvQueryFrame(capture); //Create image frames from capture
        cvShowImage("Camera_Output", frame);   //Show image frames on created window
        key = cvWaitKey(10);     //Capture Keyboard stroke
        if (char(key) == 27){
            break;      //If you hit ESC key loop will break.
        }
    }
    cvReleaseCapture(&capture); //Release capture.
    cvDestroyWindow("Camera_Output"); //Destroy Window
    return 0;
}
```

CONCLUSION

- If our project is to be continued, benchmarking has to be done
  - Motion detection programming
  - Refining of resolution and connection speeds
- If possible, one might consider to try using hardware other than Gumstix that is more user friendly

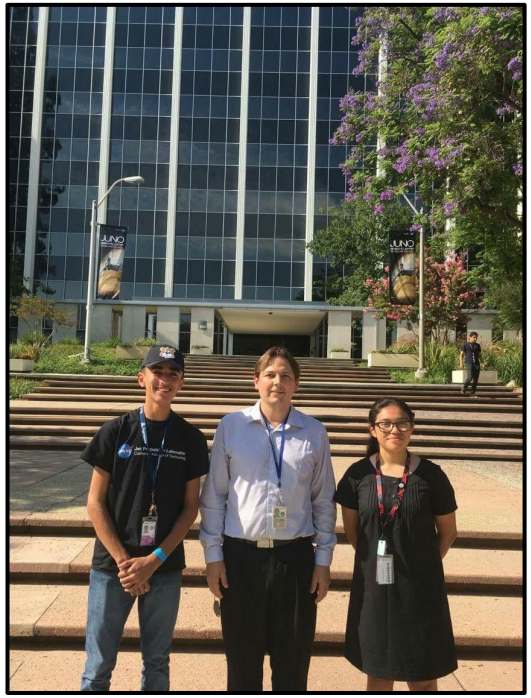
IDEAS LEARNED

- Better understanding of Linux Ubuntu, as well as other flavors of Linux (Angstrom, Poky)
- OpenCV software
- CMake building
- Arduino
- Basic knowledge of circuitry and electricity
- Further knowledge of Macintosh usage
- Exoplanets

EXPERIENCE AT JPL

ACKNOWLEDGEMENTS

- Our mentor, Justin Boland
- Gary Block and Kyle Wong
- Building 306 staff
- Santa Ana Unified School District
- And of course, Ota Lutz and David Seidel



REFERENCES

Q. (n.d.). Arduino Examples #1 Make An RGB Led Randomly Flash Different Colors. Retrieved July 26, 2016, from http://www.instructables.com/id/Arduino-Examples-1-Make-An-RGB-Led-Randomly-Flash/

Gumstix Developer Center. (n.d.). Retrieved June 29, 2016, from http://gumstix.org/access-source-code.html

Hauman, A. (n.d.). Installation in Linux¶. Retrieved June 27, 2016, from http://docs.opencv.org/2.4/doc/tutorials/introduction/linux_install/linux_install.html#linux-installation

P., R. (2014, July 14). Yocto Project Quick Start. Retrieved June 24, 2016, from http://www.yoctoproject.org/docs/1.8/yocto-project-qs/yocto-project-qs.html

Poky - Poky Build Tool and Metadata. (n.d.). Retrieved August 08, 2016, from http://git.yoctoproject.org/cgit/cgit.cgi/poky

Ubuntu Forums. (n.d.). Retrieved August 08, 2016, from https://ubuntuforums.org/

Jet Propulsion Laboratory
California Institute of Technology