

Machine Learning & Computer Vision Models applied to Control Algorithms for Cubesat Exploration Missions

Cabrera Muñoz L.A., Padilla Pérez J.A., Riojas Pesqueira J.P.

*Representative Robotics Team Fox Robotics, CETYS University.
Campus Mexicali, Calz. Cetys s/n, Rivera, 21259 Mexicali, B.C.*

*Creative Labs STEAM Education.
Blvd. Benito Juárez 3637, Residencias, 21280 Mexicali, B.C.*

Abstract— This project shows the development of a microsatellite whose mission encompasses astronomical observation. Below are listed different software solutions to perform this task, which is also accompanied by an instrumentation system developed by the team. Its objective is to monitor different performance variables of the satellite (Temperature, mechanical structure deformation, global positioning, acceleration, and rotation). All of the above, transmitted wirelessly using radio frequency communication protocols. Lastly, the project involves the design and analysis of a mechanical structure prepared to enhance an optimal performance of previously mentioned systems and withstand the space environment.

Keywords— Cubesat, Image Processing, Machine Learning, Control, Computer, monitoring, structure

I. INTRODUCTION

Microsatellites began at California Polytechnic State University (Cal-Poly) as a design initiative that allowed students and university faculty to develop prototypes of small and economical satellites, driving the development of this branch of aerospace engineering in academia. Additionally, microsatellites have long represented an option for companies and research centers focused on the development of satellite technology, serving as technology testers. That is, an economical device on which new materials, new sensors, new communication methods, and a myriad of other possibilities can be tested. This comes with an uncertain guarantee of success, in exchange for enabling field testing at a minimal cost. The objective for which a satellite is developed and launched is called a *Flight Mission* and is subject to the particular developer of the project.

Visualization of celestial bodies is a fundamental task in astronomical research projects, astrophysics, and many other sciences rooted in space exploration. Development of specialized equipment that facilitates the collection of this type of

information is a task of great interest for the astronomical scientific community.

Cubert-01 project involves structuring a microsatellite that allows precise observation and photography of the moon. This means to be achieved through the implementation of various computer vision and machine learning techniques in synergy with a closed-loop control system that keeps the position of a viewfinder fixed on the target of interest.

This control system is presented as the Flight Mission of a Cubesat prototype with a mechanical structure and control and monitoring system developed by the team.

As a student group, the design and manufacturing of this technology present an interesting opportunity to reinforce the technical skills gained in class and to develop innovative real-world solutions for the aerospace industry. The primary goal of these projects is to acquire, apply, and disseminate knowledge across various engineering fields, thereby contributing to technological advances in our university, industries and community. This aligns with the United Nations' 9th Sustainable Development Goal: Industry, Innovation, and Infrastructure, which aims to foster economic growth, social development, and climate action.

Microsatellites have gained relevance in the last years for their compact structure and economic development and deployment. Its functionalities are utilized by companies and academia to reinforce innovative solutions for space exploration, environmental macro monitoring, among other fields. This technology aims to become a paradigm shifter in aerospace, environmental sciences, telecommunications, and astronomy fields.

A. Team Overview

Software:

- Luis Adrian Cabrera Muñoz (Team Leader)
- Jesse Banda Chaidez (CS)

Hardware:

- José Alejandro Padilla Pérez (Team Leader)
- Erick Blanco Nakashima (EE)
- Guillermo Manuel Villegas (EE)
- Juan Adrian Astorga (EE)
- Julio Castañeda (EE)

Structure:

- Juan Pablo Riojas (Team Leader)
- Juan Pablo Aboytes (ME)
- Máximo Millán Cabrera (IE)

II. LITERATURE REVIEW

Cubertc-01 consists of a Cubesat prototype capable of performing all the basic tasks of a Microsatellite, as well as fulfilling a flight mission proposed by the development team. The main objectives are:

- Remote data acquisition system for GPS, IMU, temperature sensors, and strain gauges in a composite payload.
- Power the entire system using an integrated energy management solution.
- Develop an optimal system for detecting and tracking celestial bodies. This functions as the flight mission of the Cubert-01 project.
- Build a structure capable of withstanding the harsh conditions of space while enabling the functionality of the optical system

A. Microsatellites

A small satellite is generally considered to be any satellite that weighs less than 300 kg. A CubeSat, however, must conform to specific criteria that control factors such as its shape, size, and weight [1]. These design guidelines are specified in a document provided by Cubesat initiative founder university called “Cubesat Design Specification”.

The CubeSat Project began as a collaborative effort between Prof. Jordi Puig-Suari at California Polytechnic State University (Cal Poly), San Luis Obispo, and Prof. Bob Twiggs at Stanford University’s Space Systems Development Laboratory (SSDL) in 1999. The intent of the CubeSat Project was to reduce cost and

development time, increase accessibility to space, and sustain frequent launches [2].

A CubeSat is a class of satellite that adopts a standard size and form factor, whose unit is defined as ‘U’. A 1U CubeSat is a 10 cm cube with a mass of up to 2 kg [2]. larger sizes have become popular, such as the 1.5U, 2U, 3U, 6U, and 12U [1].



Fig. 1. 1U CubeSat CP1 (left). 3U CubeSat CP10 (right) [Cal Poly]

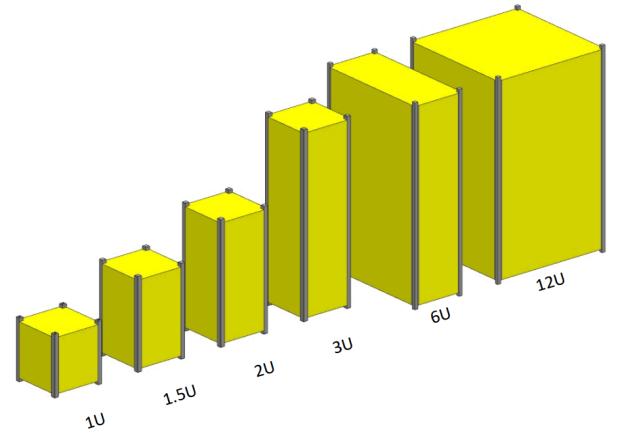


Fig. 2. The current Cubesat Family [CDS REV14]

U Configuration	Mass [kg]
1U	2.00
1.5U	3.00
2U	4.00
3U	6.00
6U	12.00
12U	24.00

Table 1. Cubesat mass specifications [CDS REV14]

B. Image Processing

Image processing is a field of computer science dedicated to the analysis and manipulation of digital images using a variety of algorithms and computational techniques. Its primary objective is

to enhance the quality of images, whether by correcting defects, removing noise, or improving parameters such as contrast and sharpness. However, it goes beyond that: it is also used to extract valuable information from images, such as identifying objects, recognizing patterns, measuring distances, calculating areas, and much more.

The applications of image processing are vast and encompass a wide range of fields and disciplines. In medicine, for example, it is used for the analysis of medical images, such as X-rays, CT scans, and MRIs, to diagnose diseases, guide surgeries, and monitor treatment progress. In astronomy, image processing is used to improve the quality of images captured by telescopes and to detect celestial objects that are difficult to see with the naked eye.

The applications of image processing are vast and encompass a wide range of fields and disciplines. In medicine, for example, it is used for the analysis of medical images, such as X-rays, CT scans, and MRIs, to diagnose diseases, guide surgeries, and monitor treatment progress. In astronomy, image processing is used to improve the quality of images captured by telescopes and to detect celestial objects that are difficult to see with the naked eye.



Fig. 3. In a B/W image, a value of 0 represents a black pixel, and 255 represents a white pixel.

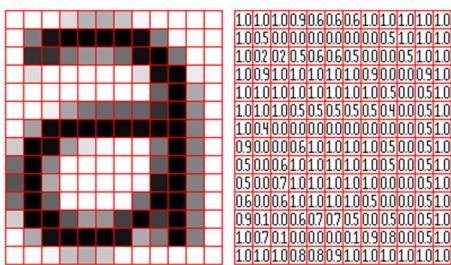


Fig. 4. How Images are processed in a pixel matrix. Pixel values are normalized to 0-1 values for this example.

Once the acquisition and preprocessing of the image are completed, a *segmentation* process is carried out. Segmentation involves dividing the image into parts or regions of interest, and there are multiple techniques for this.

Thresholding is a widely used image segmentation technique to separate objects from the background in grayscale images. The basic idea is

to choose a threshold value (typically around 150) and convert all pixels in the image above this value to one value and all pixels below this value to another value.

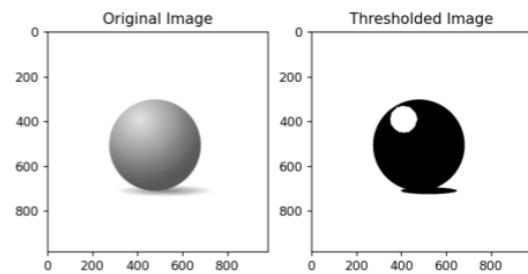


Fig. 5. Thresholding applied to a B/W ball image..

Other operations involve Convolution Filters. Through these, transformations can be applied to an image. They consist of a matrix (also known as a "kernel" or "mask") that slides over the original image and performs a mathematical operation on each pixel, typically multiplying the values of the neighboring pixels by the corresponding values in the matrix and then performing an operation decided by the developer with these data and the central value of the matrix.

These filters are widely used for various purposes in image processing, such as enhancing sharpness, edge detection, smoothing, noise removal, and feature extraction.

0	0	0	0	0
0	0	-1	0	0
0	-1	5	-1	0
0	0	-1	0	0
0	0	0	0	0



0	0	0	0	0
0	1	1	1	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0



Fig. 6. Two examples of applied kernels. These achieve the effects of sharpening and blurring respectively (the blur convolution matrix requires division by 9 to normalize the result).

Sobel Filters are an image processing technique based on calculating the derivative of the pixel intensity in an image to highlight areas of high contrast, which generally correspond to edges.

The Sobel filter uses two 3x3 convolution masks to calculate the approximations of the image

derivatives in the horizontal and vertical directions. These masks are:

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Fig. 7. Sobel Convolution Matrices for edge detection in the x-axis and y-axis.

C. Neural Networks and Object Detection Models

A neural network is a method of artificial intelligence that computers use to process data in a way inspired by the processing employed by the human brain. It structures interconnected nodes or neurons arranged in layers. This creates an adaptable system that computers use to learn from mistakes and continually improve. [3].

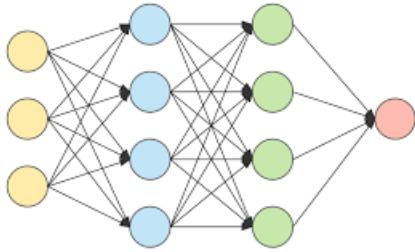


Fig. 8. Neural network basic structure

A basic neural network consists of interconnected artificial neurons organized into three layers:

The information from the outside world enters the artificial neural network through the **input layer**. Input nodes process, analyze, or classify the data and pass it to the next layer. **Hidden layers** analyze the output from the previous layer, further process it, and pass it to the next layer. They take their input from the input layer or other hidden layers. Finally, the **output layer** provides the final result of all the data processing performed by the artificial neural network. It can have one or multiple nodes. A binary classification problem will have one output node that results in either 1 or 0. A multiclass classification problem may have an output layer consisting of more than one output node. [3].

Neural networks can be classified based on how data flows from the input node to the output node. In the context of image processing and computer vision, the prominent use of **convolutional neural networks (CNN)** stands out [3].

The hidden layers of a convolutional neural network perform specific mathematical functions, such as synthesis or filtering. [3]. Just as each neuron responds to stimuli only in the restricted region of the visual field called the receptive field in the biological vision system, each neuron in a CNN processes data only in its receptive field as well. The layers are arranged in such a way so that they detect simpler patterns first (lines, curves, etc.) and more complex patterns (faces, objects, etc.) further along [4].

A CNN typically has three layers: a convolutional layer, a pooling layer, and a fully connected layer [4]. Earlier layers focus on simple features, such as colors and edges. As the image data progresses through the layers of the CNN, it starts to recognize larger elements or shapes of the object until it finally identifies the intended object [5].

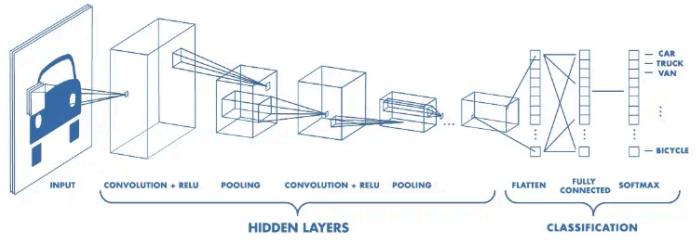


Fig. 9. CNN Architecture

Convolution layer requires a few components, which are input data, a filter, and a feature map. Assuming the input as a color image, this will have three dimensions: height, width, and depth. The feature detector, also known as a kernel or a filter, will move across the receptive fields of the image, checking if the feature is present. This process is known as a convolution [5].

The feature detector is a two-dimensional array of weights, which represents part of the image. The filter is applied to an area of the image, and a dot product is calculated between the input pixels and the filter. This dot product is then fed into an output array. Afterwards, the filter shifts by a stride, repeating the process until the kernel has swept across the entire image. The final output from the series of dot products from the input and the filter is known as a feature map, activation map, or a convolved feature [5].

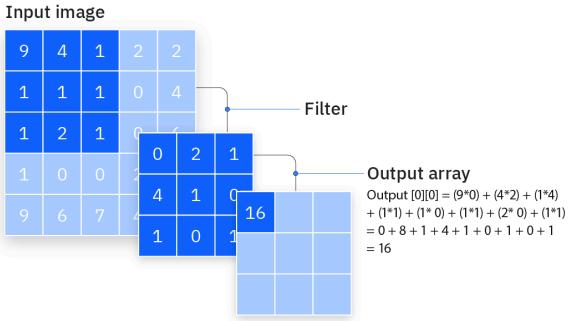


Fig. 10. Convolution Process using a 3x3 Kernel

There are three parameters which affect the volume size of the output that need to be set before the training of the neural network begins. These include:

- The *number of filters* affects the depth of the output. As the filters increase, the depth of the feature map increases too.
- *Stride* is the distance, represented in the number of pixels, that the kernel moves over the input matrix with each filter application to the input.
- *Zero-padding* is usually used when the filters do not fit the input image. This sets all elements that fall outside of the input matrix to zero, producing a larger or equally sized output. There are three types of padding: *Valid padding* drops the last convolution if dimensions do not align. *Same padding* ensures that the output layer has the same size as the input layer by modifying the output convolution formula. Finally, *full padding* increases the size of the output by adding zeros to the border of the input [5].

An input of size $W \times W \times D$ and D_{out} number of kernels with a spatial size of F with stride S and amount of padding P , then the size of output volume can be determined by the following formula [4]:

$$W_{out} = \frac{W - F + 2P}{S} + 1$$

Fig. 11. Activation map size formula

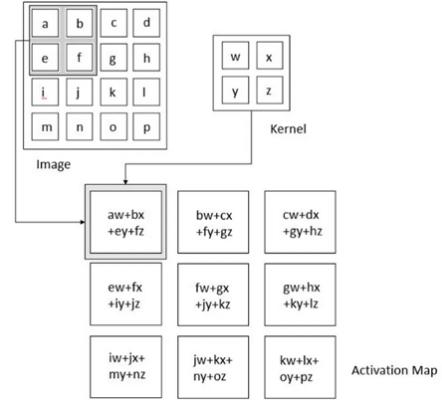


Fig. 12. Convolution Operation (Source: Deep Learning by Ian Goodfellow, Yoshua Bengio, and Aaron Courville)

Pooling layer, also known as downsampling, conducts dimensionality reduction, reducing the number of parameters in the input by deriving in certain locations, a summary statistic of the nearby outputs. This helps in reducing the spatial size of the representation, which decreases the required amount of computation and weights. [4], [5]. There are two main types of pooling: *Max pooling* selects the pixel with the maximum value to send to the output array. *Average pooling* calculates the average value within the receptive field to send to the output array [5].

An activation map of size $W \times W \times D$, followed by a pooling kernel of spatial size F , and stride S , will have a size of output volume determined by the following formula:

$$W_{out} = \frac{W - F}{S} + 1$$

Fig. 13. Pooling Kernel Output size formula

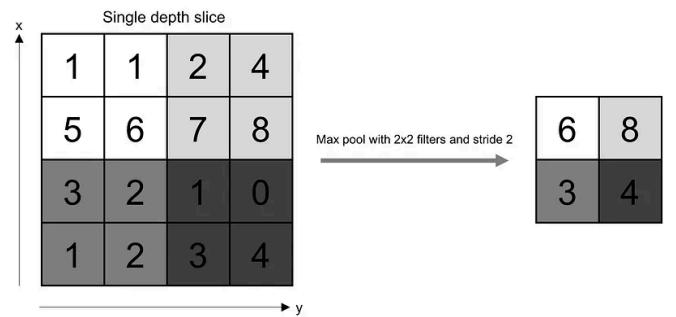


Fig. 14. Downsampling applied to 4x4 activation map

Fully connected layer connects each node in the output layer directly to a node in the previous layer. This layer performs the task of classification based on the features extracted through the previous layers and their different filters [5].

D. Closed-loop control systems

Closed-loop control systems are those in which the control action adjusts automatically based on feedback received from the controlled process. In these systems, the controller compares the actual output of the system with the reference or desired value, calculates the error, and adjusts the system inputs to minimize this error, thereby keeping the output as close as possible to the target value. This continuous feedback allows closed-loop control systems to be more precise and stable compared to open-loop systems, especially in the presence of disturbances or changes in environmental conditions. Closed-loop control systems are used to regulate and maintain process variables in a wide range of industrial, commercial, and domestic applications.

Intelligent control systems combine the principles of closed-loop control systems with artificial intelligence techniques such as machine learning, neural networks, and optimization algorithms. AI is used to enhance the accuracy and adaptability of control by analyzing large volumes of data, predicting future behaviors, and adjusting control strategies in real time. This allows intelligent control systems to adapt to dynamic changes in the environment and optimize their performance without constant human intervention, resulting in higher efficiency, reliability, and the ability to handle complex and nonlinear situations effectively.

The Proportional-Integral-Derivative (PID) control is a controller in closed-loop systems that combines three control actions: proportional, integral, and derivative. The proportional action adjusts the control signal in proportion to the current error, the integral action accumulates past errors to eliminate steady-state error, and the derivative action anticipates future error behavior by adjusting the signal based on its rate of change.

E. Hardware & Instrumentation

For this project we are using various modules, sensors and a microcomputer to process images. We can start by talking about determining the CubeSat position which is done by the **NEO6MV2** (GPS6MV2), a compact and efficient **GPS Module**, designed to provide precise data of position, speed and time thanks to satellite signals detection. With an integrated ceramic antenna and a port to connect an external antenna, it offers high sensitivity and

precision determining its geographic position. Using an UART interface to communicate with microcontrollers and other devices, it works within a voltage between 3.3 and 5 volts using NMEA protocol to transmit standard GPS data.

Ideal for drones, robots, navigation systems and vehicle tracking projects, the GPS6MV2 is a versatile solution and easy to integrate for geolocation necessities, but apart of a position module we should also have a motion sensor combining three orthogonal directions (X, Y, Z) to measure the linear acceleration experienced by the device, a work done by the **MPU-6050 Inertial Measurement Unit**. This is useful for detecting tilts, movements, and free falls. A three-axis gyroscope on a single chip detects the angular velocity around the three axes (X, Y, Z), measuring the rate of rotation, which is useful for detecting the device's turns and rotations. This sensor is used in applications for motion control, stabilization, and orientation detection in electronic systems.

Another valuable variable to measure is temperature, which is in charge of the **LM35 Precision Temperature Sensor**. It is designed to give a linear voltage reading of temperature with a sensibility of $10\text{mV}^{\circ}\text{C}$ and a precision of $\pm 0.5^{\circ}\text{C}$. It operates in a range between -55°C to 150°C , being ideal for ambient conditions monitoring, processes control and any other application that needs proper temperature control. It is also thanks to its low current consumption and a functioning rate from 4V to 30V which makes it so versatile that it is useful in a variety of electronic applications. With all this covered what is left to know is deformation and impacts, for what we have the **Strain Gauges**.

Strain Gauges BF350-3AA are used to measure the deformation of materials. This precision sensor changes its electrical resistance due to mechanical tension, which lets us measure the deformation a surface is exposed to.

It is ideal for small deformation detections, like material testing, structures monitoring and strength sensors. They are normally mounted on the surface of the material with a special adhesive that gives it a nice transference of deformation. It is also worth mentioning that it requires a wheatstone bridge for it to be read properly, but since the voltage difference is too small, we need to use an Operational Amplifier, now being the **Dual Op Amp LM358**, specially chosen by this particularity of including two independent Op Amp so we can have more deformation sensors using less space,

multiplying the entry voltage by 442 so we can read it in a scale from 0 to 3V.

With all these sensors we now need a microcontroller to receive these signals and send them so that we can interpret what they mean, that is where the **Lora Feather 32u4 II** gets in action. This microcontroller is an electronic board fabricated to establish long range wireless communications thanks to a 2.4 GHz transmission wave. The brain of the module is the ATmega32u4 chip, which synchronizes at 8 MHz and operates at 3.3V. This chip has 32 KB of flash and 2 KB of RAM. This chip synchronizes with the Semtech SX1276 RF transceiver chip. It features advanced **LoRa™** spread spectrum communication technology to ensure a very long communication range along with very low current consumption.

Lastly, we have an On Board Computer: The **Raspberry PI 4B**, a motherboard with mini PC functions without all the surrounding hardware, capable of performing a multitude of tasks. This microcomputer is capable of encoding 4k resolution videos, benefiting from faster storage via USB 3.0 and faster network connections through gigabit Ethernet.

F. Battery Management System

In order for our Cubesat to be completely wireless it needs to have its own power supply. Thinking of this, we choose to power it with two 18650 Li-on batteries of 3.7-4.2 volts with 8800 mAh connected in series. but we need to regulate the voltage going through the Microcomputer and all the components and a way to continuously charge our batteries, which is why we have integrated solar panels, a BMS board, boost and step buck converters. First, we have 5 **Solar Panels** that output 5 to 6 volts connected in parallel with protection diodes.

To output the 8 volts our BMS needs we connected the solar panels to a **Mini Boost Step Up** to reach our desired voltage. The solar panels charge the batteries and can also provide electricity to the Raspberry Pi sending it to the **Mini 360 Step Down** to get it to the Microcomputer and all the system at 5.2 volts and 2.5 - 3 amps.

G. Cubesat Structural design rules

The guidelines to follow are determined by a document called “Cubesat design rules”, or “CDS” for short. In this document, all requirements and recommendations considered by launch suppliers that determine if a satellite is a candidate for launch or not are stated. Some rules to highlight due to the concern they raised are as follows:

Materials: All aluminum alloys must be hard anodized. Hard anodizing refers to the practice of protecting our satellite with an outer layer that is achieved by exposing the metal into an electrical current in the presence of an electrolyte. This is done to avoid cold welding, which is a common incident in space that may jeopardize the flight mission.

Design: raceways must be added to ensure the proper deployment.

All satellites must have their center of gravity in the figure’s centroid, on which there is a 2 cm tolerance in any direction.

H. Cubesat work conditions

Cubesats are required to withstand both certain loads such as the one caused by their deployment mechanism and harsh weather conditions as presented in space. The conditions that were considered vary depending on the layer the satellite is positioned.

LEO: Having a distance from earth below 2000 km, LEO brings with it a plethora of both challenges and opportunities. Being the closest means that sharing data obtained via the satellite with earth is the easiest with a latency of 30ms, but the closer you are from earth, the faster all bodies are traveling, as they are competing to outrun the gravitational pull produced by the earth. positioning our satellite on LEO also adds two inconveniences that may shorten our mission’s life-span, which are atomic oxygen and temperature cycling. The first of which is a phenomenon caused by oxygen particles attaining an excited state, in which, due to the high speeds achieved in LEO, they may erode any exposed components if proper coating is not provided. Temperature cycling is a phenomenon that occurs in any orbit, but its effects worsen the quicker the body is travelling, as during orbit, heat cycles are determined by which side of the earth the body is in, if the sun is blocked by the earth the body cools down and vice-versa. In LEO, an object orbits around the earth every 128 minutes or less, meaning the cycle occurs 11.25 times a day. Finally,

another added challenge in LEO is air resistance, as this layer is not a perfect vacuum yet, as such, an ultimate failure condition is air resistance decreasing our speed enough to leave orbital speed after an extended period of time.

GEO: This orbit is characterized by its 24 hour orbit, as such, temperature cycling is not as much of a problem, but in return, we have to face a temperature range of -196 to +128 degrees celsius on a single day. Both extremes are a major limiting factor in both the material and electrical component selections. Another limiting factor is latency, as it is considered to be over 500 ms, this means that sharing the data attained by the satellite will be a painstaking process.

MEO: Is the orbit between LEO and GEO, as such, it lies between them in temperature, orbiting speed and latency. An important factor to consider is that there is no air resistance at MEO, as such, is an interesting in between as the only added challenge is the great quantity of satellites that are already orbiting at MEO.

I. Optical System

A beam splitter is a type of prism that allows two images to overlap in a single one by bouncing half of the light received into a perpendicular plane via a mirror.

The focal angle is the narrowness of the vision cone produced by a light-captivating object. The cone's origin is located at the centre of the captive element as a cone, the further the body, the easier it is to capture it.

See-through materials are required to properly capture light from the inside of the satellite while still protecting it. As any other material, they must offer sufficient mechanical properties, of which, stiffness, hardness and Young's modulus are our main concern. Stiffness is the resistance to elastic deformation, Young's modulus relates strain and stress finally, hardness refers to the resistance to punctual plastic deformation. In the case of hardness, it takes center stage for see through materials as any scratch will diminish our image quality permanently. Finally, an optical property to be considered is its optical quality, which is the percentage of light that it allows to go through unscathed, as going for an opaque material guided only by its mechanical properties may not allow the software to work.

J. Finite element analysis

Finite element analysis (FEA) is a software-driven numerical method that accurately predicts structural properties and how the entity will react to certain work conditions based on engineering data, geometry, meshing and model conditions. FEA was created with space in mind, as the impossibility of conducting trials and experiments under the actual work environment render complex prediction models necessary.

The software of choice to conduct the analysis is ANSYS, of which the multiphysics module was chosen for dynamic operations and ANSYS modal was selected to conduct modal analysis.

ANSYS Dynamic is used for modelling diverse dynamic conditions based on predicted values such as the speed of two different bodies and mechanical properties.

ANSYS modal is the base for further analysis such as harmonic and random vibration, as it determines the possible natural frequencies of a body based on the boundary conditions given, that are between free free or pre-stressed. Determining the natural frequency is key to determine the natural period of vibration which is needed to determine the time of application for impact loads and the effects of natural vibrations during deployments.

III. CUBERT-01

A. Software

Flight mission of the Cubesat involves experimenting and finding the best alternative for detecting and tracking predefined celestial bodies by implementing computer vision techniques together with neural network-based algorithms. Different alternatives were programmed for lunar detection. Below are presented two alternatives based on image processing and three solutions based on object detection models that work with convolutional neural network architectures.

a) Detection 01 | Thresholding

Pseudocode:

1. Use of Thresholding operation to achieve high contrast between the white color of the moon and the bluish-black color of the night sky.
2. Applying the HoughCircles function for detecting the moon as a circle in the input image.
3. Displaying the coordinates of the circle's center in the terminal.

Flowchart:

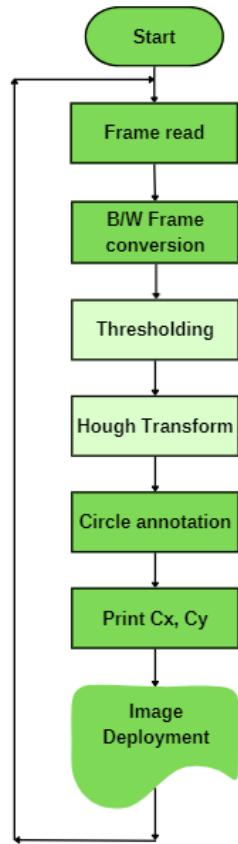


Fig. 15. Flowchart for cycled Thresholding implementation

Testing:

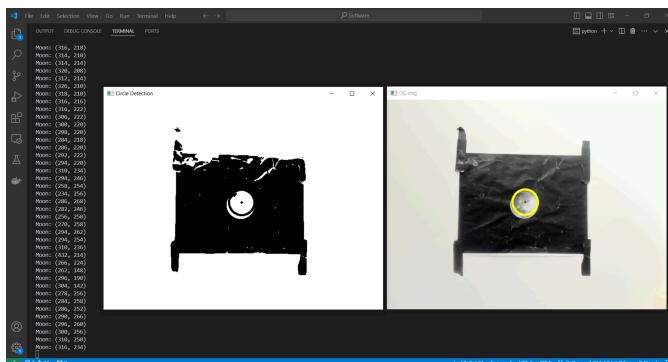


Fig. 16. Detection of the moon as a circle after applying the Threshold function followed by the Hough Transform

HoughCircles() function allows detection of small circles with considerable accuracy once the Threshold function is applied, thanks to the high contrast between the celestial body and the background generated by the Threshold function. The challenges encountered revolved around calibrating the HoughCircles() function, but they were not particularly difficult. The parametrization

of Threshold() function stands in the following line of code:

`Th, gray_trsh = cv2.threshold(gray, 100, 255, cv2.THRESH_BINARY)`

b) Detection_02 | Convolución Sobel

Pseudocode:

1. Use of Sobel convolution filters for the x and y axes to enable edge detection in the input image.
2. Apply the HoughCircles function for detecting the moon as a circle in the input image.
3. Display the coordinates of the circle's center in the terminal.

Flowchart:

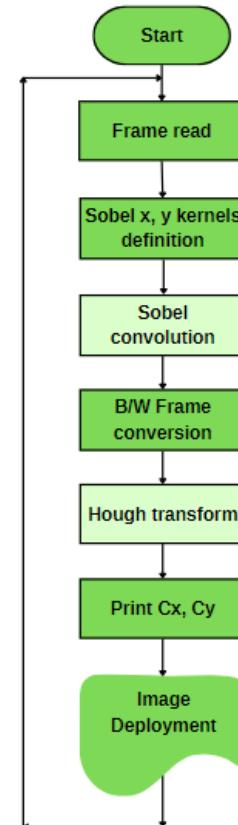


Fig. 17. Detection of the moon as a circle after applying the Threshold function followed by the Hough Transform

Testing:

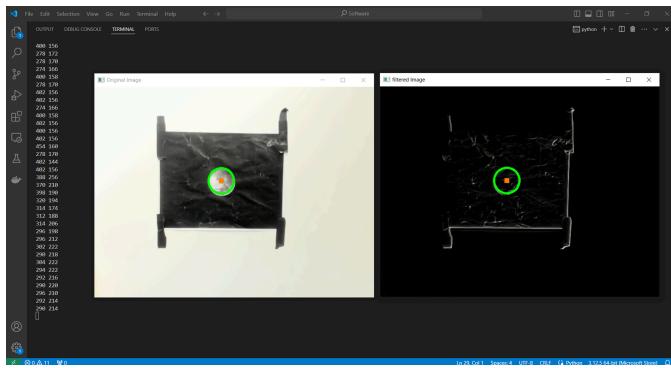


Fig. 18. Moon detection as a circle after applying the Sobel convolution in addition with Hough transform

The program works even better for detecting small circles by applying a calibration very similar to that used with Thresholding. Additionally, the Sobel convolution eliminates many false positives, a common problem when applying Thresholding.

c) Detection_03 | Yolov8 Object Detection Model

For this solution, we implemented a custom dataset with images of the moon.



Fig. 19. Custom Dataset of the Moon (Google Images). The upper group stands for a sample of the training group. The bottom group is a representation of the validation group

This dataset is used for training the Yolov8 Object Detection model. Yolov8 is a neural network architecture designed for object detection, representing a fast and accurate solution for computer vision applied to object detection.

Procedure:

1. Training the Yolov8 model (540 training images and 40 validation images).
2. Apply the trained model in real time and detect the moon in an input image.
3. Display the coordinates of the center of the bounding box in the terminal, as well as the distance of this center from the center of the frame.

Analysis & Testing:

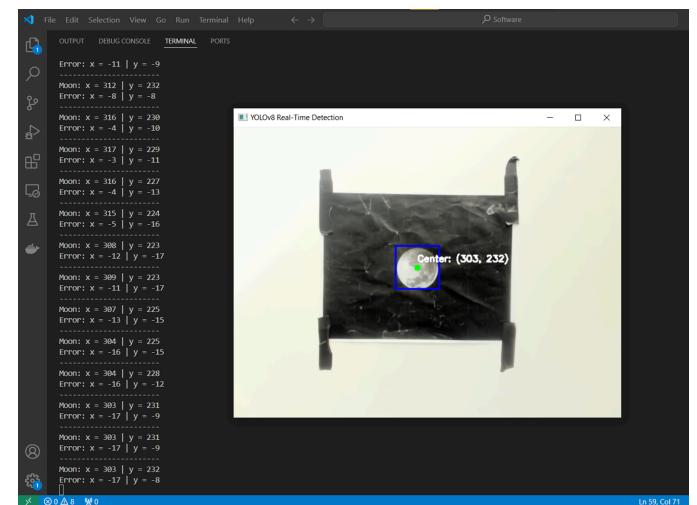


Fig. 20. Yolov8 mode used in real time for a simulated case.

The Yolov8 model performs precise detection of the moon and draws a bounding box around the object. The coordinates of this center, as well as the distances from the center of the frame, are printed to the console.

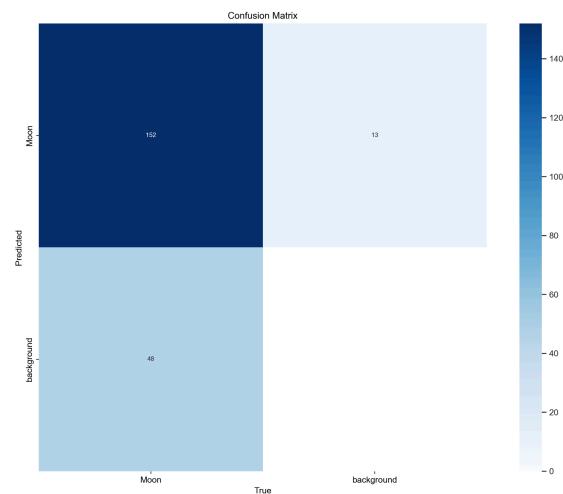


Fig. 21. Confusion matrix

Confusion matrix establishes a final run of the ML model in the validation set. This occurs after 15 training epochs. Each image is categorized in one of the next labels:

True positives (T.P.): 152 images

Images that contain a moon object and this correctly is detected by the ML model.

False positives (F.P.): 13 images

F.P. are images that do not contain a moon object. They are labeled as “Background Images” in most documentation of ML and object detection models. These images are incorrectly predicted as images that do contain a moon object, provoking a false positive detection.

False negatives (F.N.): 48

F.N. are images that do contain a moon object as T.P. but this is not detected by the ML model. Labeling these as background images.

True negatives (T.N.): 0

T.N. are images that, as equal as F.P., do not contain a moon object, and this is correctly detected by the ML model.

Once this process is executed by the ML model training process, an analytic process can be structured based on different proportions found in the results. The first one stands for the proportion of true positive detections among all the positive detections predicted by the model.

Precision is a proportion indicator of the accuracy of detected objects. It points out how many predictions were correct [7]. For this model a precision of 92.12% has been met after the training process. This indicates almost all moon detections are correctly landed.

$$\text{Precision} = \frac{T.P}{T.P + F.P} \cdot 100 = 92.12 \%$$

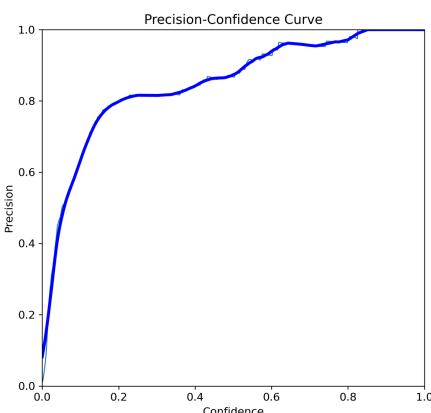


Fig. 22. Precision-Confidence curve

Due to the low quantity of false positives (13 images for this training). Increasing confidence threshold value results in elimination of these false positives to get a unitary Precision at 0.86 threshold value. This indicates that a relatively high threshold is necessary to maintain precision, avoiding false positive detections. While this approach ensures a high precision model, it may come at the cost of a poor recall result, potentially missing some true positives.

Recall, also called sensitivity, stands for the proportion of true positive detection among all the images that do contain moon objects. In this case, the recall finishes at 76%. Although it is a good result, it shows that the model requires more data to train, according to YOLO documentation [7].

$$\text{Recall} = \frac{T.P}{T.P + F.N} \cdot 100 = 76 \%$$

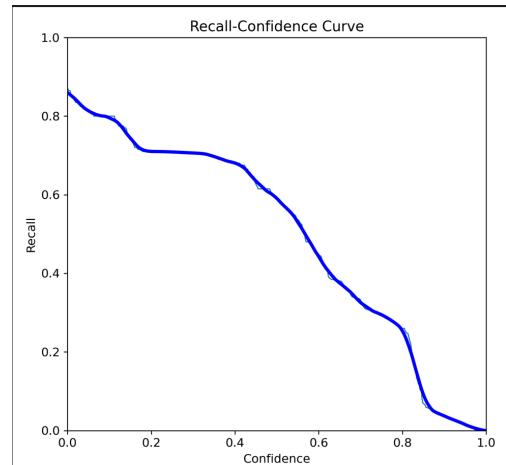


Fig. 23. Recall-Confidence curve

The model gets a 0.86 of instance detection at a close to zero threshold confidence value. As threshold value increases, the predicted moon in a given moon images dataset decreases. The recall retains around 70% until a threshold around 0.45, where it starts decreasing. Future training must seek to retain a high recall until a threshold value around 0.80 to consider a professionally viable implementation for this model.

F1 score is the harmonic mean of precision and recall, providing a balanced assessment of a model's performance while considering both false positives and false negatives [7].

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \cdot 100 = 83.28 \%$$

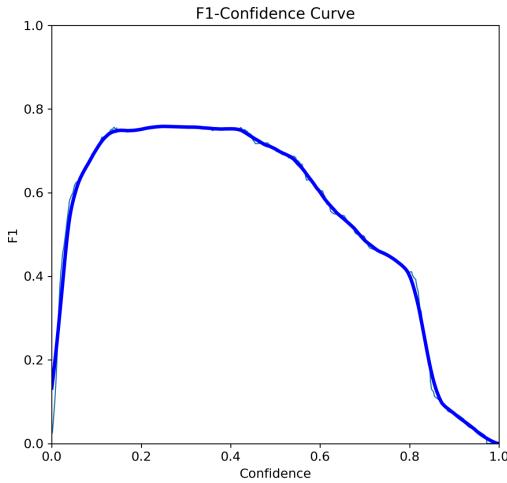


Fig. 24. F1 - Confidence curve

The F1 curve starts at a low value due to the inverse proportion between the product located in the numerator and precision value, which starts at a really low quantity. Then, it raises to around 75% F1 value, which lasts until 0.4 confidence value. This represents a stabilization of precision-recall relationship from 0.15 to approximately 0.45 confidence values. This being similar to the recall stabilization of 70% around the same confidence value. This stabilization also exists in the precision curve for 80% precision value. Lastly, this shows a F1 stable performance until the ML model meets a 0.50 confidence value. At this moment, F1 relationship starts decreasing, maintaining a decent performance until 0.75 confidence value, where model performance reaches an infeasible effectiveness.

Finally, the matrix results exhibit 71.36% of guesses as correctly identified and predicted by the model. Although it is a good accuracy, more data and data augmentation techniques are necessary to increase this percentage and, in general, ML model performance.

$$\text{Accuracy} = \frac{T.P + T.N}{T.P + T.N + F.P + F.N} \cdot 100 = 71.36 \%$$

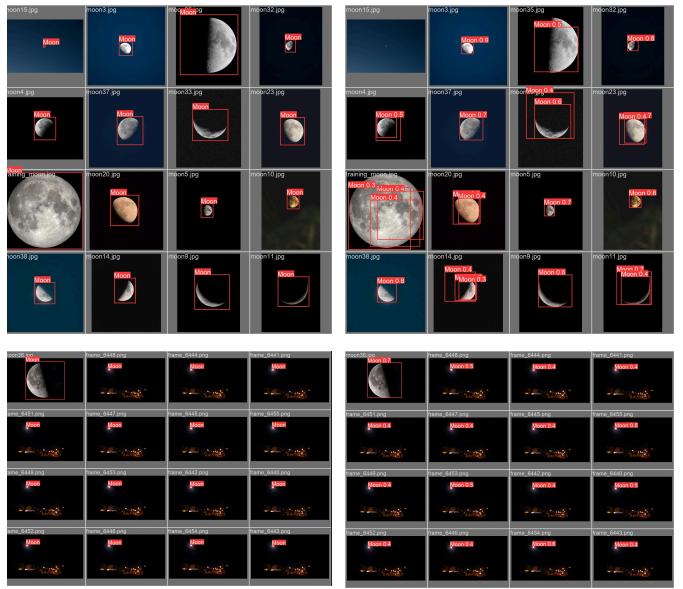


Fig. 25. Dataset annotation (left) Object Detection (Right)

Previous images represent the final run YOLO model executes over validation dataset. Google is detected with high confidence values. While local camera sample images (bottom images) are detected with more difficulty. It is hard to identify an image with more than 0.5 as a confidence value.

B. Moon Tracking | PID Control

Detection algorithm integrates its features in a closed loop control system (CLCS) that implements a physical tracking of the moon using a servo motor that rotates an additional component, which is the beam splitter referenced in “Optical system” inside Literature review. This movement occurs with the objective of provoking a frame adjustment in the camera and ensures an optimal capture of the moon.

Procedure:

- Determination of the X-axis coordinates of the identified object's center (Moon).
- Determination of the distance between the center of the input frame and the center of the identified object (Control system closed-loop error).
- Transmission of information to the MCU to perform control actions, ensuring that the steady-state error is minimized in terms of pixel distance between the two centers.

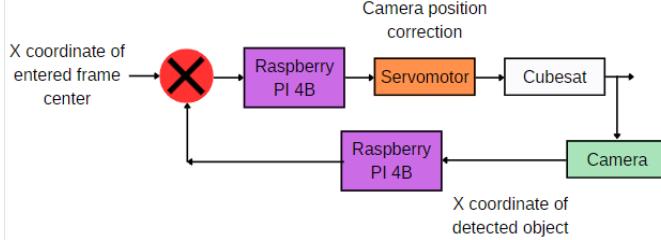


Fig. 27. Cubesat close vision system

To experiment with this control system, a base separate from the Cubesat body was used at first, which solely performed the task of object tracking. When the response was correctly performed by the system. A mechanical design was conceived for beam splitter integration. This design performs the tracking action.

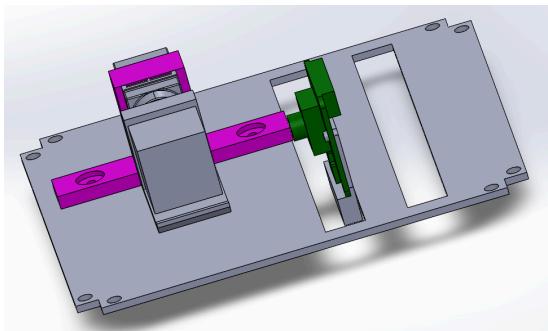


Fig. 27. Mechanical design for moon tracking CLCS integration

C. Hardware & Instrumentation

Flight mission will be accompanied by a control and monitoring system of mechanics, navigation and performance variables.

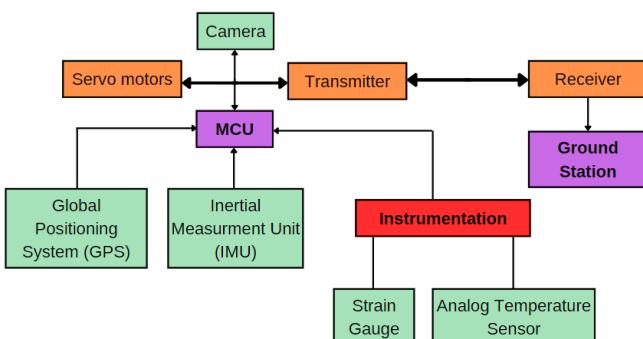


Fig. 28. Cubesat block diagram

Task	Component
Inertial Measurement Unit	MPU6050
Global Positioning System	NEO6MV2
Strain Gauges	BF350 3AA + Wheatstone Bridge & Operational Amplifier
Temperature Sensor	LM35 + Instrumentation

Table 2. Component/s selection for monitoring system tasks

The information provided by this module of the project will be processed by a Lora 32u4 II module. This microcontroller houses an AVR family microchip: ATmega32u4 integrated with an RF module: RFM95. This component will transmit the information via radio waves to a counterpart that will display the information on a second computer, enabling remote data acquisition.

Component	Interface
MPU6050	I2C Port
GPS6MV2	Serial Port
BF350 3AA	A0
LM35	A4 & A5

Table 3. Component/s connection with LoRa microcontroller

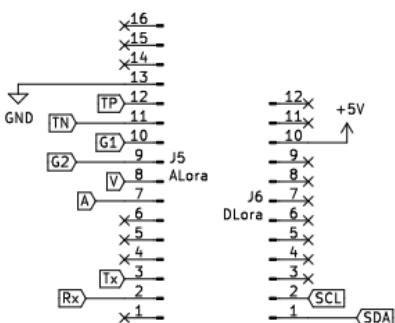


Fig. 29. Lora's pin connections

Strain gauges BF350-3AA must be properly instrumented so that their transduction can be correctly interpreted by the MCU. At this stage, a Wheatstone bridge is to be used, as well as a differential Op Amp LM358 with a gain of 62.66 to obtain a readable signal for the LoRa MCU. The calculations for the output voltage that will enter an analog pin of the microcontroller are detailed below:

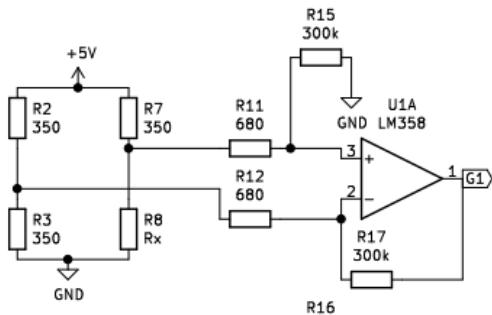


Fig. 30. Strain Gauges instrumentation schematic

Resistor	Resistance (Value)
R2	350 Ω
R7	350 Ω
R3	350 Ω
R8	Strain Gauge (350 Ω - 352 Ω)
R11	680 Ω
R12	680 Ω
R15	300 kΩ
R17	300 kΩ

Table 4. Resistance value for each resistor in Fig. 30. schematic

Wheatstone max voltage difference:

$$Vd = 5 \cdot \left(\frac{352}{702} - \frac{350}{700} \right)$$

$$Vd = 5 \cdot (0.501 - 0.5) = 0.0071v$$

Opamp voltage output:

$$Vout = Vd \cdot \left(\frac{300,000}{680} \right)$$

$$Vout = 0.0071 \cdot 441.176 = 3.13v$$

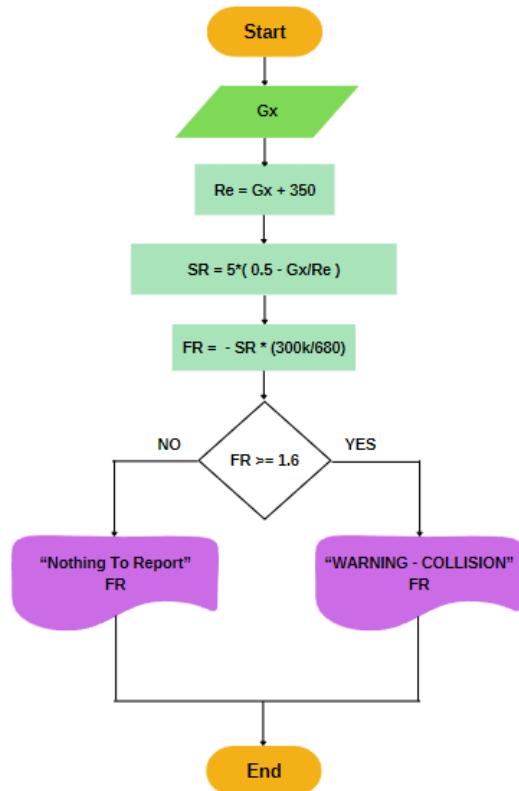


Fig. 31. Collision detection flow chart

The **operational amplifier** we are using to boost the Wheatstone bridge is the **LM358**, which belongs to a sort of low power dual operational amplifier, consisting of two independent, high gain, internally frequency compensated operational amplifiers, which were designed specifically to operate from a single power supply over a wide range of voltages.

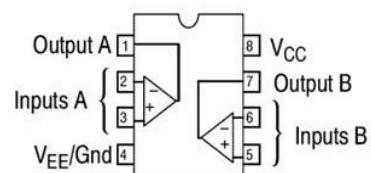


Fig. 32. Operational amplifier LM358 structure (Source: NetSonic)

In order to be able to read sub-zero temperatures without the need of a negative voltage supply, the **LM35 temperature sensor** needs instrumentation with a pair of resistance and small current diodes, which will get us the mentioned range of -55 °C to 150 °C after obtaining the difference between the readings of the output pin and the negative pin.

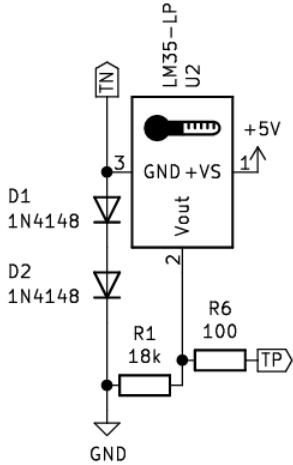


Fig. 33. Schematic for Lm35 instrumentation

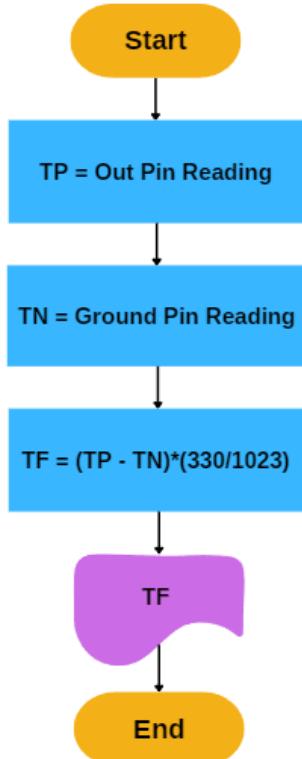


Fig. 34. Temperature sensor flow chart

Global Position System (NEO6MV2) and the **Inertial Measurement Unit (MPU6050)** modules mentioned before function in a pretty standard way, with the GPS printing numbers to decode in a maps software and know the precise location. For the IMU it prints several numbers that mean acceleration in X, Y and Z axes, along with the orientation in the same axes, but it also has an embedded temperature sensor which goes from -45 to 80 Celsius degrees, so we get some redundancy in case the other temperature sensor stopped working correctly and vice versa.

D. Structure and mechanics

In order to properly design a body **structure** for the satellite, we had to take into account weight limitations, work conditions, possible failure cases, weight distribution and electrical requirements.

Aluminum alloys were selected for the main frame and protective plates due to their low density and overall high scores in hardness trials in both Rockwell A and B format, considerable yield and ultimate tensile strength, respectable modulus of elasticity and overall amazing thermal management while having a relatively low density.[21]

Two alloys were chosen for further consideration, representing the 7xxx and 6xxx families. 2xxx alloys such as 2024 were not considered due to their poor resistance to corrosion. [22]

Property	7075	6061
Density	2.81 g/cc	2.7 g/cc
Hardness Rockwell	A: 53.5 B: 87	A: 40 B: 60
Ultimate tensile strength	527 MPa	310 MPa
Yield tensile strength	503 MPa	276 MPa
Modulus of elasticity	71.7 GPa	68.9 GPa
Machinability	70%	50%

Table 5. Mechanical properties comparison between aluminium 7075 and aluminium 6061 alloys.

In general, 7075 has a better density/mechanical property ratio, but it is harder to weld and more expensive. 6061 scores lower in all areas other than density. Selecting a final material requires further analysis but as of now, 6061 seems sufficient for our application but rigorous analysis via FEA and the final design must be done. A promising option is building all of the mainframe and all outer plates other than the ram face with 6061, and use 7075 for the ram face as it will experience most of the impacts.

For transparent materials polycarbonate and fused silica are both ends of a spectrum, in which, fused silica offers a near 100% light transmission percentage and amazing mechanical properties other than machinability,[23] but due to its expensive nature it is out of our scope.

Polycarbonate's melting point is way over the highest temperature expected in GEO, which is our worst case scenario temperature wise. Its mechanical properties are sufficient to withstand impacts and it is already used in space, but it will lower the image quality notoriously[24]. Other materials such as kevlar were considered, but were ultimately discarded due to its opacity.

Various **models** were designed with different objectives in mind, these models are our estimated final product, prototype for 3D printing and FEA model.

Estimated final product includes the satellite's payload and assembly method in its design. It is way too complex for printing or CAM but it is used for space optimization and simple evaluations such as weight.

Our printing prototype eliminates any electrical component and eliminates some unnecessary instances. It is also used to determine points of failure during metal manufacturing early on. Certain adjustments required for manufacturing such as adding slopes were done to ensure proper printing.

FEA model strips any unnecessary geometry from the base structure such as bores to ease both the load for the mesher and solver.

For manufacturing, all structural components were modeled in SIEMENS NX, as it offers a better CAD to CAM translation via its manufacturing modules and set standards for operations such as bores.

Gravitational gradient involved 3 mechanisms that were considered during the research phase: Reaction wheels allow a degree of free rotation per wheel added to the satellite; they work by storing torque generated by a motor spinning at thousands of RPM. Reaction wheels bring a set of problems that are intertwined with their design, as depending on constantly rotating components that are not available for maintenance usually leads to critical failure presented as fatigue damage on the shaft caused by the lack of lubrication. Reaction wheels also require another active gravitational gradient to unload, which means to reset the stored momentum by allowing another system to keep the satellite aligned.[25] A system that is usually used both as stand alone gravitational gradients and as a support system for reaction wheels are magnetorquers, they work by using the earth's magnetic field for stabilization, they are a viable alternative for our

mechanism of choice, but they are more expensive than our final pick. [26]

The mechanism we selected is a gradient boomstick, which is an extendable stick with a weighted mass. This mass must be enough to counteract the torque produced by the gravitational pull of earth, keeping the satellite under control. To design the gravitational gradient, the weight distribution of all other components must be found, as the satellite's gravitational matrix is needed.

Designing a way to adjust the camera based on the information provided by our MCU was one of the biggest bottle necks we faced, as all standard solutions were either insufficient or out of our reach. There was a need to devise an original **optical system**. Magnetorquers were too slow for any significant adjustments. Flywheels are expensive and unreliable as they are considered points of failure, and they require another active gravitational gradient mechanism to unload the generated torque.

The requirements that our system has are as follows: The captive element (camera) must be protected from direct sunlight as it would burn the system. Any circular motion actuator must be able to generate a back and forth motion to anulate any generated momentum. All internal mechanisms must not require the displacement of any wire or any other component that may cause a critical failure due to disconnection or unbounding.

Based on these requirements, we decided that redirecting light into the camera is our best option. A beam splitter mounted to a servo motor was chosen as the beam splitter can capture light and redirect it in a perpendicular manner, allowing us to avoid a moving mount for our camera, complying with the last criteria we set. The servo motor allows us to precisely control each degree of movement while allowing back and forth moving to counteract any torque generated.

Finally, the base that connects the prism to the servo motor covers both the face that is directly in front to the one that is facing the camera and the face that is directly under the one capturing the images. This is done to avoid image mixture and to have an angle where the servo can not capture any light, protecting the camera from sun rays.

The finite element method for mechanical analysis was employed as a way to evaluate the toll some of our expected work conditions will have on our satellite.

Before running any analysis, we had to optimize our model based on the software requirements. First, any components that were deemed as non contributing for the analysis were removed, leaving only the aluminum main frame and the internal plates. Afterwards, the model was further simplified using SpaceClaim by converting the 3d outer and internal plates into shell elements, this decision was made as a way to circumvent the element cap imposed by ANSYS student version, as generating proper high quality hexahedral mesh with only 3D elements far exceeded the student version capabilities. Nonetheless, the model quality and precision were not compromised.

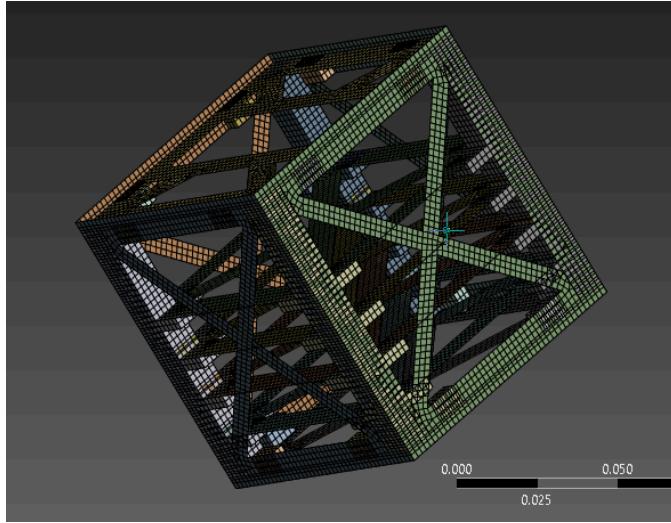


Fig. 35. ANSYS meshed model

The first module employed was ANSYS modal. This module is the cornerstone for every structural analysis, as it was employed to evaluate the model execution and mating.

A modal analysis will have 6 rigid modes for each free body in the system, as such, the first 6 modes must be 0 or close to it.

Mode	Frequency [Hz]	X Direction
1	0.	3.4717e-002
2	0.	-1.6413e-002
3	2.5211e-003	2.2117e-002
4	3.5822e-003	1.435e-002
5	4.4652e-003	2.9261e-002
6	5.2059e-003	1.8701e-002
7	1702.3	-1.6227e-012

Table 6. First seven modes

This abstract from the first 7 modes show that our model is treated as a single body, which validates the model for further analysis.

The next solution to run is a prestressed modal analysis. This is due to the fact that the distance

between our satellite and the earth's center is constant due to the orbital and escape velocity threshold. This means there is a movement constraint that is exerted. The load applied in our model was simulated as a static structural analysis, where the gravitational acceleration constant of MEO was exerted on all plates.

Mode	Frequency [Hz]	X Direction
1	1660.	8.5853e-006
2	1773.2	-1.1522e-004
3	1773.9	5.759e-004
4	1774.3	-0.11986
5	1774.5	0.0010000000000000002

Table 7. First four pre-stressed models

The first frequency or mode obtained is the one defining the application time of the impact load, which is necessary for further impact analysis.

Impact simulation for cubesats is out of reach as of now, because there is a variable called “damping coefficient” that measures how long it takes for a system to dissipate the energy it absorbs via mechanisms offered by the environment it is surrounded by. The conditions offered by space are impossible to simulate in conjunction, and as such the only way to obtain this value is by in-space experimentation.

Taking this into consideration, the strain gauges mentioned above play a key role in determining both the experimented deformation and the time it takes to recover its original state, which is the basis for obtaining the damping coefficient. The strain gauges will be placed in the main frame plates, as they mate with all surfaces that are exposed to impact and offer various points for instrumentation, which are key to determine a more precise damping coefficient. The reason we propose strain gauges to be used as the instrumentation method for this experiment comes from the way they manage voltage, as deformation can be measured precisely while generating a register of changes and even obtaining the permanent deformation in set plate coordinates, allowing us to estimate the point of impact. Having this into account, a transient structural model was designed for future evaluation of the obtained data.

The closer a satellite is from earth, the more collisions are expected, as such, we can select the average number of collisions in LEO as our model’s cycles, which is way over the actual expected limit with a value of 19 [27]. Another value that is constant is the velocity space debris is

travelling at, which is estimated by NASA to be at around 10 km/s [27].

Mass is not as straightforward, as there are more than 25,000 instances of space debris that are constantly being monitored as their diameter is larger than 10 cm. Other than those cases, there are around 500,000 particles in the ranges between 1 to 10 cm and more than a 100 million instances of particles larger than a mm [27].

The general idea of the analysis is to estimate when the structure will fail. There are various methods to estimate variable loads such as the Palmgren and Miner cumulative damage law and Goodman and Gerber and Soderberg mean fatigue stress diagrams. Both models are great data equalizers, but the mean fatigue stress diagrams offer a better relationship with our work conditions as it takes into account the probability of each magnitude instead of only the number of cycles it will take for failure to occur.

To determine the transient structural model we must dictate 4 variables which are the number of cycles, the impact force, impact duration and the damping coefficient. The number of cycles and the impact duration, of which, both the number of cycles, impact velocity and the impact duration were already determined. The impact force can be represented by the equation $F=MV/2T$ where: M equals mass, V equals velocity and T equals duration.[28]. To determine mass, the statistical approach was chosen:

size range (m)	f_i (absolute frequency)
0.001-0.01	130000000
0.01-0.1	1100000
0.1-1	40500

Table 8. Size categories and their absolute frequencies

With these values in mind, we can obtain the median using the formula $X=\Sigma(X_i)(f_i)/N$. where: X_i is the mark class, f_i the absolute frequency and N the total number of entries. After running this calculation, we get a median diameter of 0.0061 m. Obtaining this value means that we can calculate our mass, as we can assume in a conservative manner that the body has the volume of a sphere of the stated diameter and that the material that composes the debris is Aluminum, as most is the space debris comes from the crash of satellites [29].

As to calculate impact based on the formula previously stated, we get that mass equals the volume of a sphere times the density of aluminum which is taken as 2710 kg/ cubic meter, which is 0.000319. velocity was previously stated as 10000 m/s and time is the natural vibration period of the first mode divided by 3 which is 1/4980 or 0.0002008. Now we can calculate that the impact force for this model is 8016.5052 Newtons.

After obtaining these values, we can create our impact model. To achieve accurate data we must use impact physics. As for the time step, we consider that the auto step must be off as our impact force is constant and as such, we can determine out time step as a third of the load duration as we are not expecting major fluctuations. The step number has to be double the number of expected impacts, as there must be dedicated steps for unloading the structure, the duration of these steps can be selected arbitrarily as long as they comply with that criteria. As it was already mentioned, an impact load lasts a maximum of $\frac{1}{3}$ on the natural vibration period of the first non rigid mode, which in our case is 1/(1660x3). To set the boundary conditions a 0 DOF remote displacement was added to the plate directly in front of the one receiving the impacts as due to the speed the satellite is traveling at, a non catastrophic impact would not be able to alter the direction on that plate. Finally the impact load will only be applied in the x direction because both bodies are orbiting at the same radius.

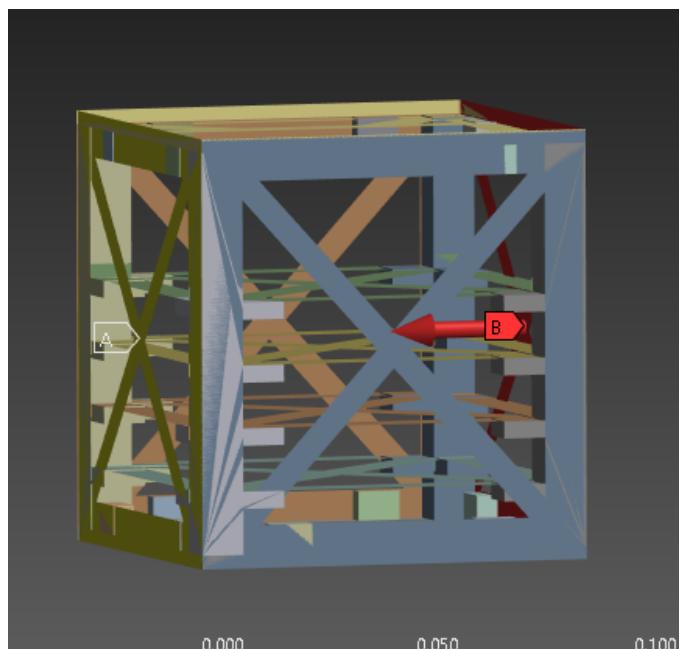


Fig. 36. Impact model constraints and loads

	Steps	Time [s]	<input checked="" type="checkbox"/> X [N]	<input checked="" type="checkbox"/> Y [N]	<input checked="" type="checkbox"/> Z [N]
1	1	0.	= -100.	= 0.	= 0.
2	1	5.8823e-003	-100.	0.	0.
3	2	1.	= -100.	= 0.	= 0.
4	3	1.0059	= -100.	= 0.	= 0.
5	4	2.	= -100.	= 0.	= 0.
6	5	2.0059	= -100.	= 0.	= 0.
7	6	3.	= -100.	= 0.	= 0.
8	7	3.0059	= -100.	= 0.	= 0.
9	8	4.	= -100.	= 0.	= 0.
10	9	4.0059	= -100.	= 0.	= 0.
11	10	5.	= -100.	= 0.	= 0.
12	11	5.0059	= -100.	= 0.	= 0.
13	12	6.	= -100.	= 0.	= 0.
14	13	6.0059	= -100.	= 0.	= 0.
15	14	7.	= -100.	= 0.	= 0.
16	15	7.0059	= -100.	= 0.	= 0.
17	16	8.	= -100.	= 0.	= 0.
18	17	8.0059	= -100.	= 0.	= 0.
19	18	9.	= -100.	= 0.	= 0.
20	19	9.0059	= -100.	= 0.	= 0.
21	20	10.	= -100.	= 0.	= 0.
*					

Table 9. Loading and unloading table

IV. RESULTS

The flight mission was successfully developed. After some tests, the ML model was deemed as the best alternative, being capable of detecting the moon during a field real time test.

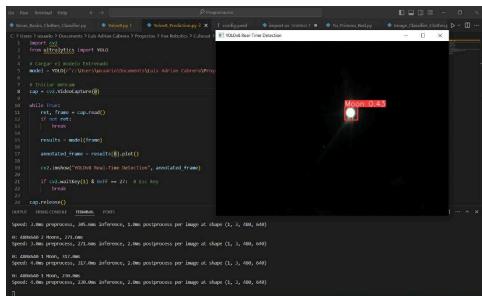


Fig. 37. Detection test with the real moon using the detection model Yolov8 with personalized Dataset

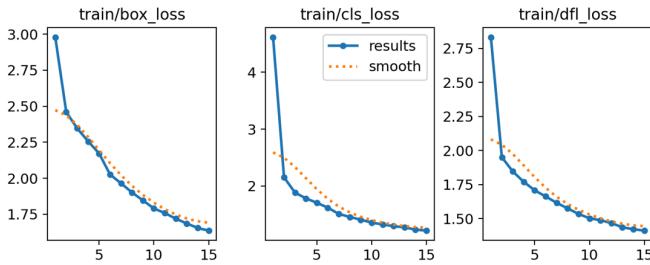


Fig. 38. loss graphs (training set)

Box loss is defined as an area magnitude that indicates how well the predicted bounding boxes

match the truth bounding boxes. train/box loss shows a likely-linear decreasing of box loss as the training epochs occur. Finishing at a Box loss value of 1.6365, according to the results file. Class loss and distribution focal loss are crucial when dealing with classification problems. Nevertheless, for a single class object detection model, these parameters are not really useful, they can also be found in the results files (A.2).

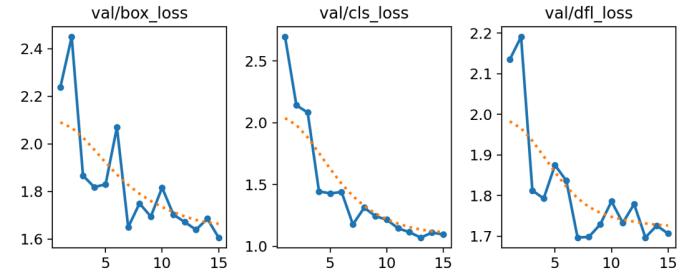


Fig. 39. loss graphs (validation set)

As the loss functions are displayed for the training dataset, YOLO also tracks the evolution of these loss functions across the validation dataset. Due to the limited quantity of validation data, the resulting curves exhibit significant variability, leading to abrupt adjustments in the model's weights and biases. These changes require delicate monitoring and more data application to ensure training stability in further model enhancement.

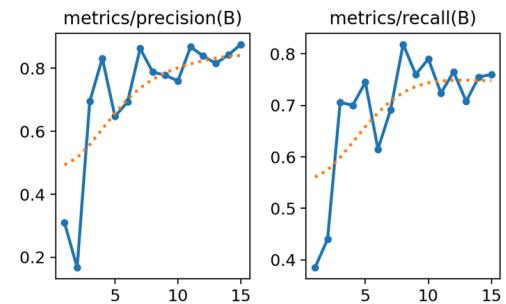


Fig. 40. Precision & Recall graphs

Precision is the ratio of true positive detections to the sum of all positive detections. Results file establishes 0.87621 as the final result for this metric. Although it results inconsistent with the proportion previously established by the confusion matrix, the final measure stands close to the 0.90 precision required to verify this metric as satisfactory for professionally viable ML model use. On the other hand, recall finishes at the predefined 0.76 value. Although this requires more data variation and data augmentation techniques as well as a larger training process. It is a good

performance to start complementing the ML model with control systems for moon tracking with satisfactory results.

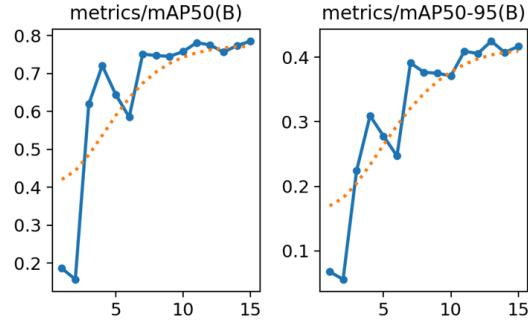


Fig. 41. Mean average precision metrics Graphs

Mean average precision (mAP50) measures model precision considering an IoU confidence threshold of 50%, in other words, precision over “easy detections” as mentioned in YOLO documentation [7]. The ML model exhibits a satisfactory mAP50 of 0.78544, a value that can be considered a viable moon detector for future academia development in the same field. mAP50-95 represents “hard” detections”. The precision measure starts with the same 0.50 precision threshold. As the evaluation continues, this threshold increases to 0.95, making the model only consider really low box loss detections to be a true positive result. Training process yields a 0.41686, which is expected due to the high difficulty detection task committed to the ML model. To improve these results, more data variability, data augmentation techniques and continuous training processes must be assigned to the present ML model.

Even though the results were good, it seemed to be hard for the Raspberry Pi to run the model. To fix this, a sampling process was added to the code. The moon is not an object that requires fast control adjustment. That is, the computer does not require constant tracking processes to maintain the object in focus. Consequently, a subprocess was introduced to count frames, and a condition was implemented for the program to perform a detection every 50 frames read. The results obtained with this configuration do not compromise the success of the flight mission. Steady state error was set around 20 pixels, and the tracking tests were successfully conducted.

Instrumentation circuit worked as expected with just some variations from theoretical calculus mostly due to resistance's usual variation, every

lecture was successfully processed and remote communication.

Sensors tests were quickly developed except for the GPS module that required a code correction to properly work. Although testing the components was a simple task, it was a challenge to be able to have them all in the same space and getting the hardware in the mechanical structure, but we were able to solve it on time. Here is the accommodation of the project.

Color	Label
Red	Vcc
Yellow	GND
Orange	Sensing Data
Others	Nodes bridges

Table 10. Color coding for first prototype cable management

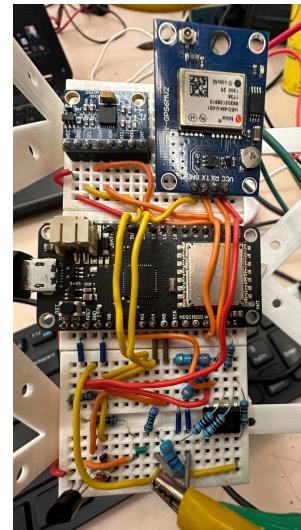


Fig. 42. Breadboard for payload implementation on first prototype

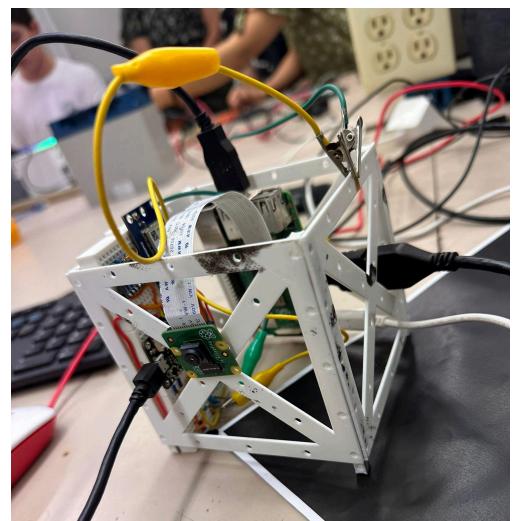


Fig. 43. Final assembly of the project (First prototype)

After the accommodation in the protoboard worked, we transported it to a PCB to get it properly assembled in the Cubesat, working the same way it did in the protoboard, but in a more compact and secure way, which is necessary for the prototype to fit in the designed dimensions. For the batteries we manufactured a small board in a perforated plate with the components mentioned before, working perfectly to supply power to the whole project with the batteries and the solar panels working independently. Then our little Cubert-01 was complete.

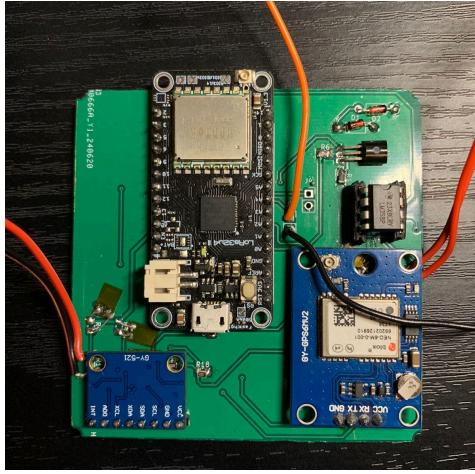


Fig. 44. Main PCB with embedded components

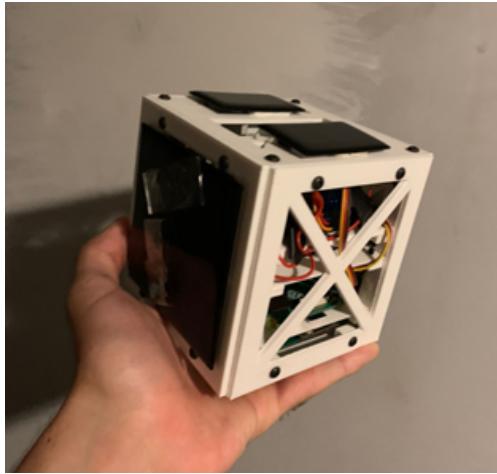


Fig. 45. Final assembly of the project (Actual prototype)

V. FUTURE VIEW

The next step for the flight mission is to expand detection and tracking CLCS to involve more celestial bodies as well as improve the metrics and performance of the present iteration. We are experimenting with platforms like roboflow, which simplifies the process of creating, training, and using a model. As a starting point, we prepared a

short dataset that now includes sun and moon to start a training process focused not only on object detection, but classification. It is important to mention that although it is not feasible to detect sun due to the optical systems implications with our capture instrument, it helps us to build a robust model and evaluate performance metrics with different variables to build high quality datasets for other celestial bodies in the future.

Orbit model-01

- Dataset: 867 images.
 - Training: 626
 - Validation: 165
 - Test set: 77
- Direct photos of the moon and the sun.
- Auto-orientation process.
- image size adjustment (640x640).
- 300 training epochs

To train the model, we used MS COCO (Common Objects in Context) v14 as the starting point, as it is currently the most accurate dataset, and YOLO-NAS as the model architecture, which is slightly slower but highly precise.

The purpose of including this model is to observe how it responds when adding more object classes, particularly in situations where there are similar objects in the camera's view, in this case, the sun.

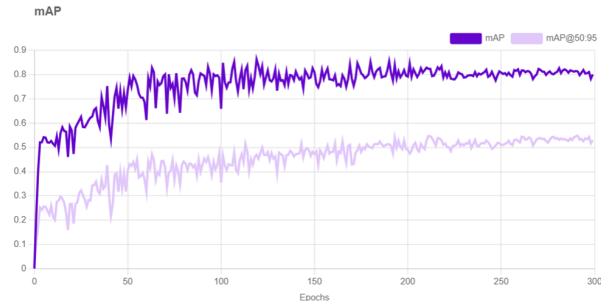


Fig. 46. mAP evolution through Orbit model-01 training process

- mAP: 86.7%
- Precision: 92.4%
- Recall: 82.5%

This model provides similar results to the main model in mAP and Precision with a slight improvement in recall metric. These are good results, and its implications in future training processes are accepted. Full metrics graphs are provided in appendices section (A.4).

For this project we used components that even though they worked as intended, aren't totally suitable for space applications, this is why we made

a parallel research of some Space-Certificate components we would like to implement in a more advanced prototype, starting with the on board computer. The ***NanoMind HP MK3*** proves to be a small and efficient microcomputer, with a similar size to the raspberry pi we are currently using, it offers great flexibility, performance and customizations options for all kinds of applications. Featuring a Xilinx Zynq 7030/7045 Programmable SoC, it can provide up to 800 Mhz for processing, and it has a functional operation between -40 and 85 degrees Celsius. The comparison of different On-Board computers to reach this candidate can be found in (A.3).

Lodestone Module motion sensor features a 3 axis gyroscope (FRS from 15.625 up to 2000 degrees by second) and magnetometer, which are capable of capturing data from three different directions allowing our device to determine its attitude with ease, this is a strong candidate to replace MPU6050 sensor used for this prototype.

A more advanced temperature sensor would be beneficial. The ***Temperature Sensor 3200 series*** would be a great candidate if it wasn't for its size, since it is specifically designed for satellites and rockets, reading from -51 to 162.8 Celsius degrees. Nevertheless, a device with a similar temperature range is desirable for a successful performance of monitoring system for future project iterations

A deeper research on these subjects, as well as in batteries and camera modules topics is needed so that we can get better images and more operating time, but that will also depend on the budget we get as a team.

During the prototype's manufacturing face, some opportunities for design improvement were found in the cable management department as due to the limited space there were many connections that were harder to make than expected, which is a possible point of failure in manufacturing. As a solution we are considering designing in each plate an specific route for each cable, as adding gaps within the inner plates will greatly alter the satellite's resistance to impact.

There are 2 mechanisms pending for design: The gravitational gradient and the antenna deployment mechanism. The first was deemed as the last step as it requires the body's gravitational matrix. On the other hand, both of them require one time deployment mechanisms, which add a layer of complexity to them.

VI. CONCLUSIONS

During the development of this project we encountered various troubles that we managed to surpass as a team. We were enthusiastic to reach our goal, so we are still doing research about the topic and improving different systems of our prototype. We first started as it looked in Fig. 43, in a 3D printed simple structure, with different power supplies for the camera module the LORA and getting everything wired on a breadboard. Then preceding to make more accurate mechanical analysis, testing different programs and models for the camera system, as well as preceding to design a PCB for connections with the different sensors.

We have already mentioned the uses of a cubesat and taking in consideration different limitations that we as students have, our priority has been to get a working prototype with the resources and time at hand, that could someday conduct to a better version capable of traveling without a problem through the Earth's orbit.

What we have learned about different electrical systems, programming and mechanical abilities has teached us that we still have a vast area of opportunity to continue learning and improving, while also giving us new tools to help us during our major studies and working opportunities.

Cubesats still have a lot of potential and growing opportunities which we would like to exploit. For now we can confidently say that we satisfactorily reached our established goals, and look forward to keep developing this project with more research and funding.

VII. ACKNOWLEDGEMENTS

As a representative robotics and STEM team (Fox Robotics), we would like to recognize the invaluable management and technical contributions of our advisors Silvia Arámbula Zazueta and Daniel Millán Coronado. They provided a strong organization for us to develop the project we are presenting today. Also, it is important to mention the knowledge and intellectual material provided by our university at México: CETYS University. In addition, CREATIVE LABS STEAM provided an action path and a generous quantity of technical revisions to ensure that our project comes to compete on its best form, we don't have enough words to describe how grateful we are.

As Luis Adrian Cabrera Muñoz, I specially thank Jessica Borja professor, a person that provided constant technical support at the beginning, and

during the development process of the Machine learning model training and analysis. Daniel Millan was a key professor to begin this project. As I started working with him one year ago, I learned about organization, strategy, and the implications of working and collaborating in a project, with a team, a competition, and a University. Finally, this project would not have been possible without the support my family gave me during the technical and large administrative process of this project. What comes out from this is something I share with them.

As Jose Alejandro Padilla, I shall say that I feel the same way about our teacher Daniel Millan and my own family, to whom I am completely grateful for supporting me and helping me to be able to be here. Lastly I would like to thank the professor Mario Ramos for his tips during batteries and BMS experimentation tips, and for being a professor who has always supported the Fox Robotics team.

As Juan Pablo Riojas I first and foremost extend my wholehearted appreciation for my family, to whom I owe everything due to the work they did as role models. I would also like to acknowledge Silvia and Javier Arámbula individually as they opened the doors that brought me here. As a student I want to recognize the help I received from Dr Yamel Ungson Almeida, Dr. Juan Andres Rivera Santana, Mr Salvador Baltazar Murrieta and Dr Miguel Ponce, as they were incredible professors that although they were not part of the project, they helped us in various ways. Finally, I want to wholeheartedly thank Ing Erick Siqueiros, who without being a professor, taught me the possibilities of engineering with his projects.

VIII. REFERENCES

- [1] “Basic Concepts and Processes for First-Time CubeSat Developers NASA CubeSat Launch Initiative.” Available: https://www.nasa.gov/wp-content/uploads/2017/03/nasa_cslcubesat_101_508.pdf?emrc=05d3e2
- [2] “CubeSat Design Specification Rev. 14.1 The CubeSat Program, Cal Poly SLO CubeSat Design Specification Cal Poly -San Luis Obispo, CA Document Classification X Public Domain.” Available: https://static1.squarespace.com/static/5418c831e4b0fa4ecac1bacd/162193b7fc9e72e0053f00910/1645820809779/CDS+REV14_1+2022-02-09.pdf
- [3] “¿Qué es una red neuronal? - Explicación de las redes neuronales artificiales - AWS,” Amazon Web Services, Inc., 2023. <https://aws.amazon.com/es/what-is/neural-network/#:~:text=A%20neural%20network%20is%20a,that%20resembles%20the%20human%20brain>
- [4] M. Mishra, “Convolutional Neural Networks, Explained - Towards Data Science,” Medium, Aug. 26, 2020. <https://towardsdatascience.com/convolutional-neural-networks-explained-9cc5188c4939>
- [5] “What are Convolutional Neural Networks? | IBM,” Ibm.com, Oct. 06, 2021. <https://www.ibm.com/topics/convolutional-neural-networks>
- [6] C. Education, “Microsatélites: El CUBESAT,” YouTube. Oct. 09, 2020. Accessed: Jun. 23, 2024. [YouTube Video]. Available: <https://www.youtube.com/watch?v=I7qjejIYv8w>
- [7] Ultralytics, “YOLO Performance Metrics,” docs.ultralytics.com. <https://docs.ultralytics.com/guides/yolo-performance-metrics/#class-wise-metrics>
- [8] Technical Training, “22 Example4 Reading Positive and Negative Temperature from LM35 with manual scan,” YouTube. Apr. 04, 2024. Accessed: May 31, 2024. [YouTube Video]. Available: <https://www.youtube.com/watch?v=TY5XXtVzkw>
- [9] CESAR Education in Astronomy. (2020, 9 octubre). *Microsatélites: el CUBESAT* [Vídeo]. YouTube. <https://www.youtube.com/watch?v=I7qjejIYv8w>
- [10] CUBESAT 101 BASIC CONCEPTS AND PROCESSES FOR FIRST-TIME CUBESAT DEVELOPERS - NASA CUBESAT LAUNCH INITIATIVE. (s. f.). <https://www.readkong.com/page/cubesat-101-basic-concepts-and-processes-for-first-time-3834125>
- [11] CubeSat Information — CubeSAT. (s. f.). CubeSat. <https://www.cubesat.org/cubesatinfo>
- [12] CDS+REV14_1+2022-02-09.pdf (squarespace.com)
- [13] “datasheet mpu6050”. Home | TDK InvenSense. Accessed 26 May 2024. [Online]. Source: <https://invensense.tdk.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf>
- [14] “NEO-6MV2 GPS Module”. Components101. Accedido el 25 de mayo de 2024. [En linea]. Disponible: <https://components101.com/modules/neo-6mv2-gps-module>
- [15] “LM35 datasheet(3/13 Pages) NSC”. ALLDATASHEET.COM - Electronic Parts Datasheet Search. Accessed 26 May 2024. [Online]. Source: <https://html.alldatasheet.com/html-pdf/8866/NSC/LM35/113/3/LM35.htm>
- [16] LM358 Datasheet and Pinout - Low Power Dual Operational Amplifier. Available: <https://netsonic.fi/en/lm358-datasheet-and-pinout-low-power-dual-operational-amplifier/>
- [17] Raspberry Pi Datasheets. Accessed 26 May 2024. [Online]. Source: <https://datasheets.raspberrypi.com/rpi4/raspberry-pi-4-datasheet.pdf>
- [18] High Reliability Aerospace Thermostats. SatNow. Available: <https://www.satnow.com/products/temperature-sensors/honeywell-aerospace/151-1202-3200-series>
- [19] NanoMind HP MK3. GomSpace. Available: <https://www.spacemanic.com/files/datasheet/datasheet-AD-Lodestone.pdf>
- [20] Lodestone Gyro & Mag module. Space Manic. Available: <https://www.spacemanic.com/files/datasheet/datasheet-AD-Lodestone.pdf>
- [21] Thyssenkrup. (2023). What Are The Properties Of Aluminium?. <https://www.thyssenkrupp-materials.co.uk/properties-of-aluminium>
- [22] 22 WayKen rapid manufacturing “Types of aluminum alloys: Understand different aluminum grades - WayKen,” Rapid Prototype Manufacturing in China - WayKen, Mar. 10, 2023. <https://waykenrm.com/blogs/types-of-aluminum-alloys/>
- [23] 23 K. B. M. Director, “Material focus: Fused Silica glass,” UQG Optics. <https://www.uqgoptics.com/material-focus-fused-silica-glass/>
- [24] 24 Admin, “The Plastics That Made it to Space,” Plastrac, Jun. 08, 2021. <https://plastrac.com/the-plastics-that-made-it-to-space/>
- [25] 25 Aegis, “In space, no one can hear your bearings fail,” Aug. 23, 2018. <https://blog.est-aegis.com/in-space-no-one-can-hear-your-bearings-fail>
- [26] Nanoavionics (2024) “CubeSat magnetorquer SatBus MTQ” <https://nanoavionics.com/cubesat-components/cubesat-magnetorquer-satbus-mtq/>
- [27] NASA, “environmental Testing Survey” <https://nepg.nasa.gov/docuploads/C5E0869C-0469-4D11-9FAA8012C8F52351/environmental%20Testing%20Survey.doc>
- [28] GeeksforGeeks, “How to calculate the Impact Force?,” GeeksforGeeks, m<https://www.geeksforgeeks.org/how-to-calculate-the-impact-force/>
- [29] The European Space Agency. ESA Space Environmental. https://www.esa.int/Space_Safety/Space_Debris/ESA_Space_Environment_Report_2024#:~:text=About%209%20100%20of%20these,damage%20%2E2%20%30%20is%20over%20one%20million
- [30] United Nation Office for Outer Space Affairs. (2023, June). Updated UNOOSA and ESA space debris infographics and podcasts. Available: <https://www.unoosa.org/oosa/en/informationfor/media/unoosa-and-esarelease-infographics-and-podcasts-about-space-debris.html>
- [31] “NanoMind HP MK3 Datasheet.” [Online]. Available: https://gomspace.com/UserFiles/Subsystems/datasheet/gs-ds-NanoMind_HP_MK3.pdf
- [32] “Lodestone Gyro&Mag module SM-AD-LODESTONE FUNCTIONAL CHARACTERISTICS.[Online]. Available: <https://www.spacemanic.com/files/datasheet/datasheet-AD-Lodestone.pdf>
- [33] “3200 Series - Honeywell Aerospace | Space Qualified Temperature Sensor,” Satnow.com, 2024. <https://www.satnow.com/products/temperature-sensors/honeywell-aerospace/151-1202-3200-series>.