Mission Space Lab Phase 4 report

Team name: Atlantes

Chosen theme: Life on Earth

Organisation name: Niubit

Country: Spain

1. Introduction

Our research aimed to gamify the analysis of the data collected on the space station by Astro Pi IR by procedurally building worlds within the Minecraft video game.

The aim was to use motivational tools to develop aspects such as:

- Building bridges between the real and virtual worlds.
- Improve our Python coding skills.
- To know the NDVI index as an analysis tool.

At the same time, we also wanted this attractive project to motivate us to learn new techniques and/or improve old ones, as well as to work on the management of a long-term project in a team.

2. Method

Our Astro Pi IR program, during phase 3, collected every 15 seconds all possible sensor data added the position of the station and recorded it in a CSV file.

One of the first ideas of the project was to build graphics inside Minecraft worlds from the data. We wanted to code a kind of alternative to the matplotlib library or an adaptation of it to generate in-game graphics. We found that such a project already existed (Matplotlib-Minecraft-Backend) which made us consider alternatives. Figure 1 shows one of the tests we carried out with the library we found.

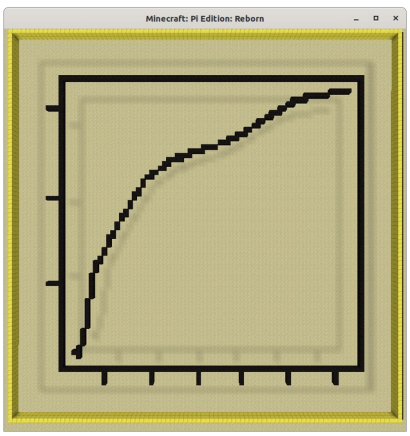

Figure 1: Temp. in ISS vs datetime

Our code also recorded pictures of the Earth with the IR camera that we converted to NDVI and coloured with the *fastiecm* scale. Another of our ideas was to translate these pictures into Minecraft to build worlds that could later be explored while debating their meaning within the simulation.

We made this the main objective, so we will focus on it in the results section. All the work was done in Jupyter notebooks, with the *mcpi* library as Python API for Minecraft and Minecraft Pi Reborn as client.

3. Experiment results

Having focused on the use of Earth pictures, the first result was a selection of those pictures. There was not much luck with either the landscapes or the NDVI graphics, but we selected a couple of scenes here:

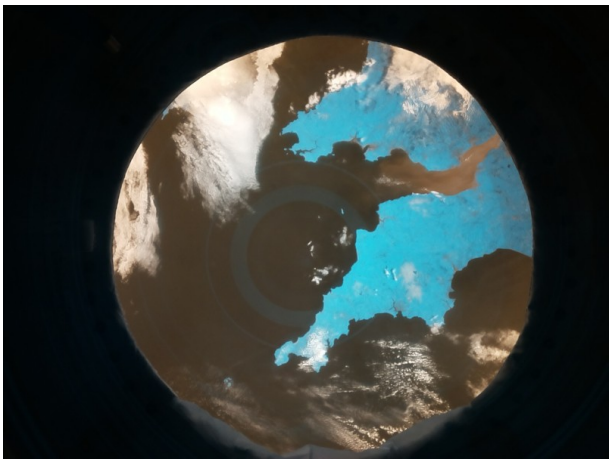

Figure 2: Bristol Channel

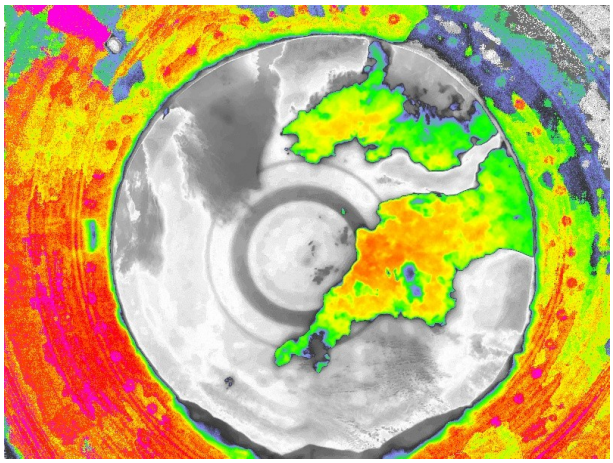

Figure 3: Bristol Channel - NDVI/fastiecm

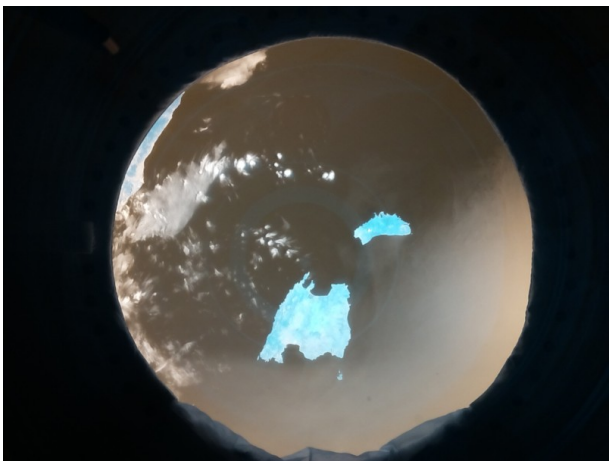

Figure 4: Balearic Islands

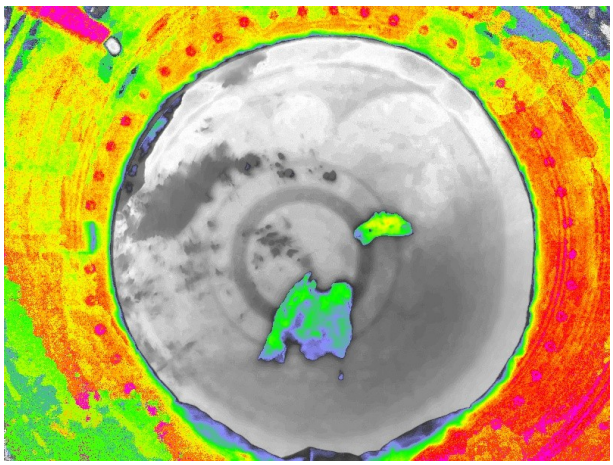

Figure 5: Balearic Islands - NDVI/fastiecm

In order to be able to represent the pictures in NDVI/fastiecm scale, we first had to convert the 256 colour table to the 16 colours allowed by the wool block in Minecraft. Using a simple algorithm and making some minor manual corrections, we arrived at the following:

Figure 6: Normal fastiecm scale vs adapted to Minecraft

With this table reduced to 16 colours, the same photo of the Bristol Channel coloured according to NDVI would look as in Figure 7.

The next step is to integrate the techniques seen above and draw the picture of the Bristol Channel with the wool blocks according to the correspondence found between them and the different NDVI values. The result shown in Figure 8 is a flat, colourful world, which, when viewed from a suitable height, looks very similar to Figure 7.

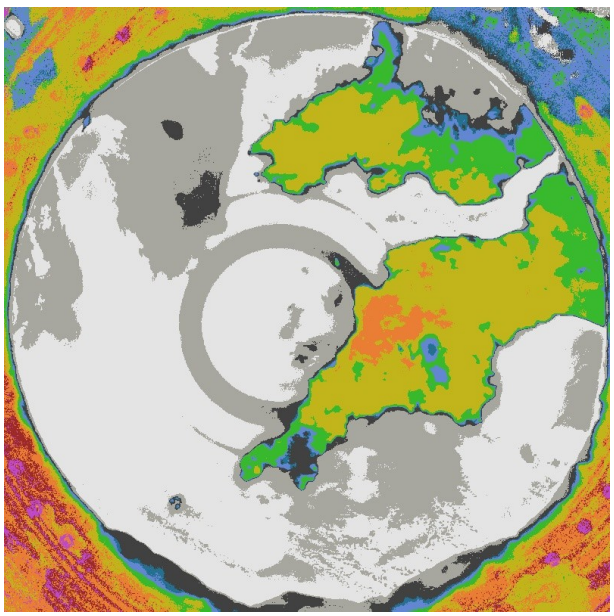

Figure 7: Bristol Channel - NDVI/fastiecm_wool

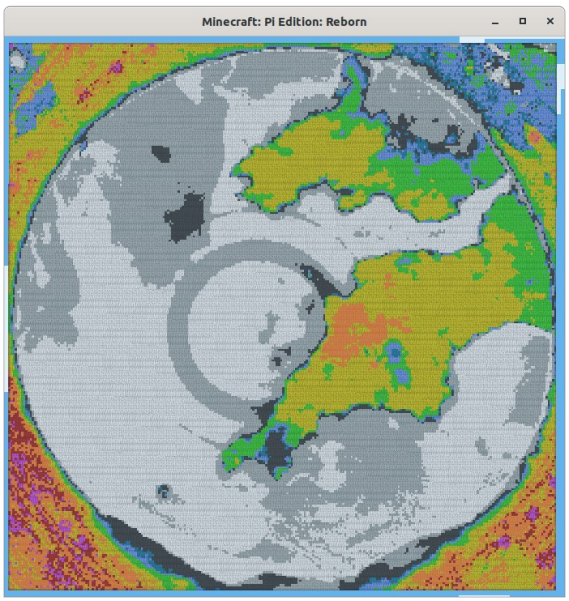

Figure 8: Bristol Channel - NDVI/Minecraft

Since Minecraft is a three-dimensional environment, we thought about adding depth to the data. The first step was to convert the different NDVI values to different block heights. All negative NDVI values are represented with the same height 0 so that the relief reflects only areas with life traces (Figure 9).

We end with an integration of the relief display with the *fastiecm_wool* colour scale (Figure 10).

All the analysis was done with Python code on Jupyter with the notebook and libraries that can be found here: https://github.com/niubit/astropi_atlantes_2021-2022

This link to Google Colab allows you to view the notebook directly:
https://drive.google.com/file/d/1D78nh8ITbtj_tsL1k3zflt5wFiV8BV1k/view

Finally, a video presentation of the project and the results:
https://youtu.be/i7ny866KDjE

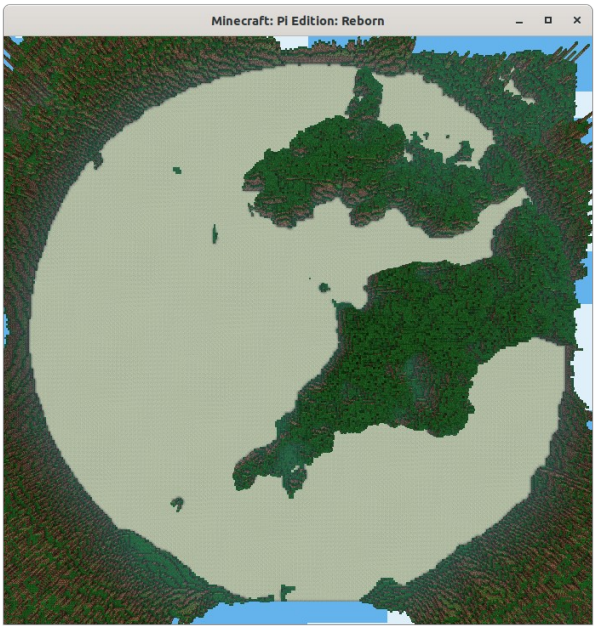

Figure 9: Bristol Channel – Minecraft Relief

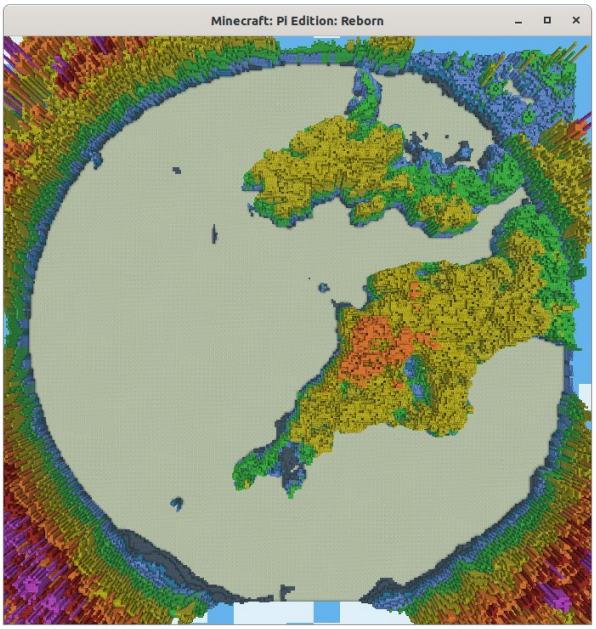

Figure 10: Bristol Channel - NDVI + Minecraft Relief

4. Learnings

The lessons learned from this project include encouraging us to use lateral thinking, since a priori a series of data generated by sensors and pictures have no direct relationship with a virtual environment such as Minecraft.

To achieve this, we had to hold several brainstorming sessions to gradually narrow down the options.

In addition to thinking, programming and playing, we decided to make a video to communicate the project, which had its own challenges, difficulties and learnings. Putting together a video takes a lot more effort than we thought it would!

5. Conclusion

The educational aspects of the video game Minecraft motivate us a lot and that was the main aspect that encouraged us to think that it would be a good idea to use it as an engine for a project like this.

NDVI analysis of pictures captured from the ISS seems like a powerful tool, but after the changes in Astro Pi hardware this year, we thought it would be better to start with a project where sample accuracy is not vital to the success of it. Gamifying the data helped us to ease the stress of understanding what NDVI means and how we can leverage it in future projects.

The Python/Minecraft pairing has been very interesting and powerful for us. If we add the excitement of the Astro Pi program, the result has been an engaging and stimulating project. We will surely continue to explore the possibilities of these tools in more projects.