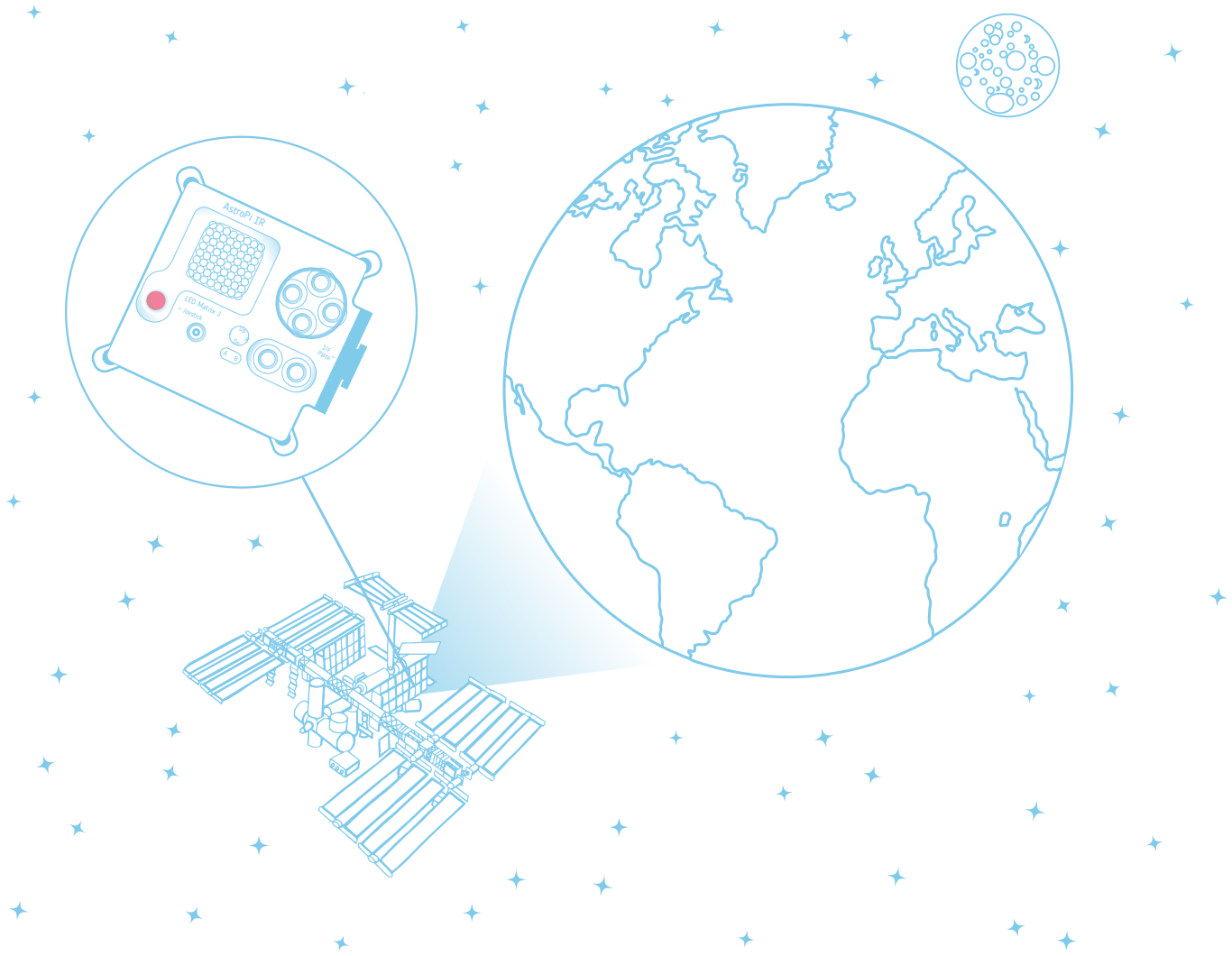


teach with space

→ MEET THE ASTRO PI CAMERAS

Using a Raspberry Pi to take pictures and see in near-infrared





Activity 0: Assembling the Pi Camera page 4

Activity 1: Using the Pi Camera page 5

Activity 2: Still pictures & recording videos page 10

Activity 3: Seeing in near-infrared page 13

Extension - Using the NoIR camera to monitor plant health page 15

Meet the Astro Pi Cameras

Two special Astro Pis, nicknamed Ed and Izzy, have been qualified for spaceflight and are on-board the International Space Station (ISS). Ed and Izzy are identical except for one detail: Astro Pi Ed has a standard visible spectrum camera and Astro Pi Izzy has an infrared (or NoIR) camera. Both cameras cover the visible spectrum; however, the NoIR camera extends into the near-infrared, providing many extra options for experimentation. With the cameras, you can take still images and record videos and time-lapses.

In this set of activities, you will explore the Astro Pi cameras and learn the essential coding language that you need to program them.

Equipment

- Raspberry Pi
- Power supply
- Pi camera (Visible and NoIR)
- Monitor
- USB keyboard and mouse

→ ACTIVITY 0: ASSEMBLING THE PI CAMERA

In this activity you will assemble the Pi camera, connect it and enable it on the Raspberry Pi.

Exercise:

1. Firstly, shut down and disconnect power from your Raspberry Pi. Then you can connect your camera. You can choose either camera, as the installation process for both cameras is the same. Locate the camera port, pull the lock up and put the flat ribbon cable into it. It should be located between the Ethernet and HDMI ports, with the silver connectors facing the HDMI port. The blue part of the cable should face the Ethernet port.

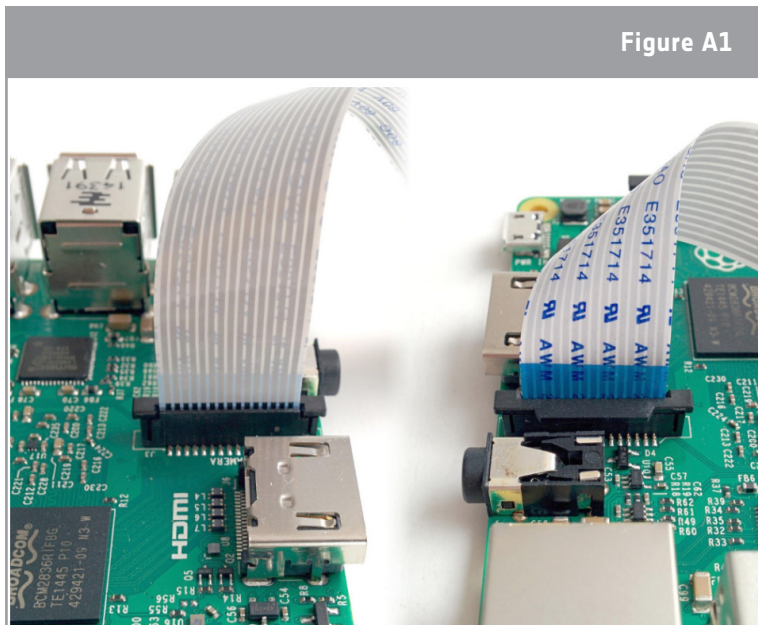


Figure A1

Note: To avoid technical issues please install your camera into the Astro Pi only when the computer is off.

2. Turn on the Raspberry Pi.
3. Choose main menu → Preferences → Raspberry Pi Configuration

↑ Showing the Raspberry Pi connected to the ribbon cable of the camera

4. Next, in the Interface section, make sure that the camera is enabled for your device. If it is not enabled, enable it and reboot the Raspberry Pi.

Note: To use the picamera library you must have an updated version of Rasbian. If you don't have it, you need to install it in a terminal window. The terminal window can be accessed by clicking the icon circled in blue in Figure A2. The command you will need to use is:
`sudo apt-get install python3-picamera`

Remember that you need to have an Internet connection to do this.

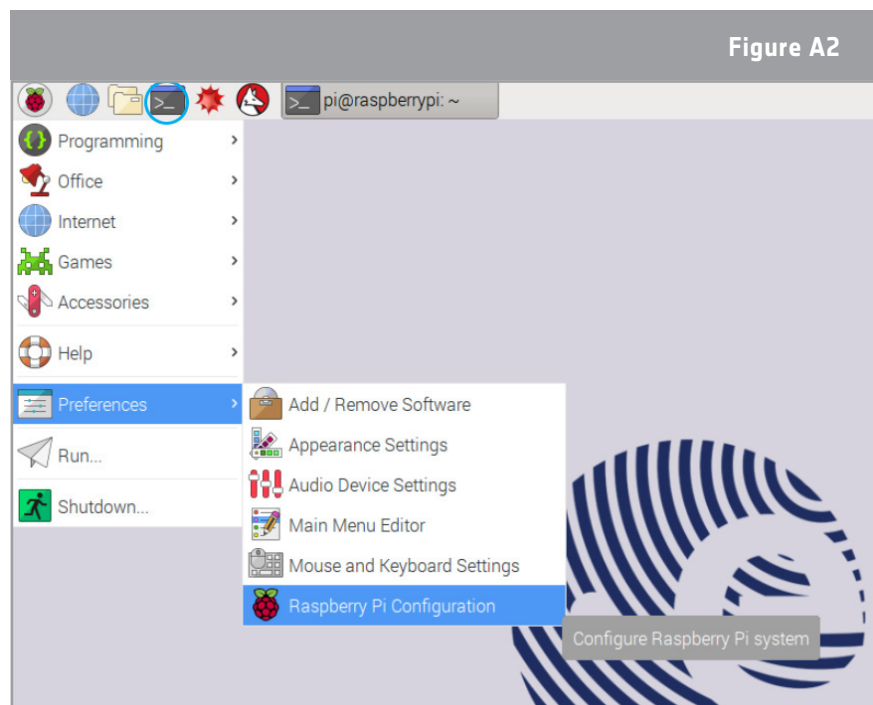


Figure A2

↑ How to access the Raspberry Pi Configuration Menu

→ ACTIVITY 1: USING THE PI CAMERA

The Astro Pi Cameras on the ISS are both five-megapixel cameras, giving high-quality still images and supporting a variety of video recording modes -just like those found in a normal mobile phone.

In this activity you will start exploring the basic features of the Astro Pi Cameras.

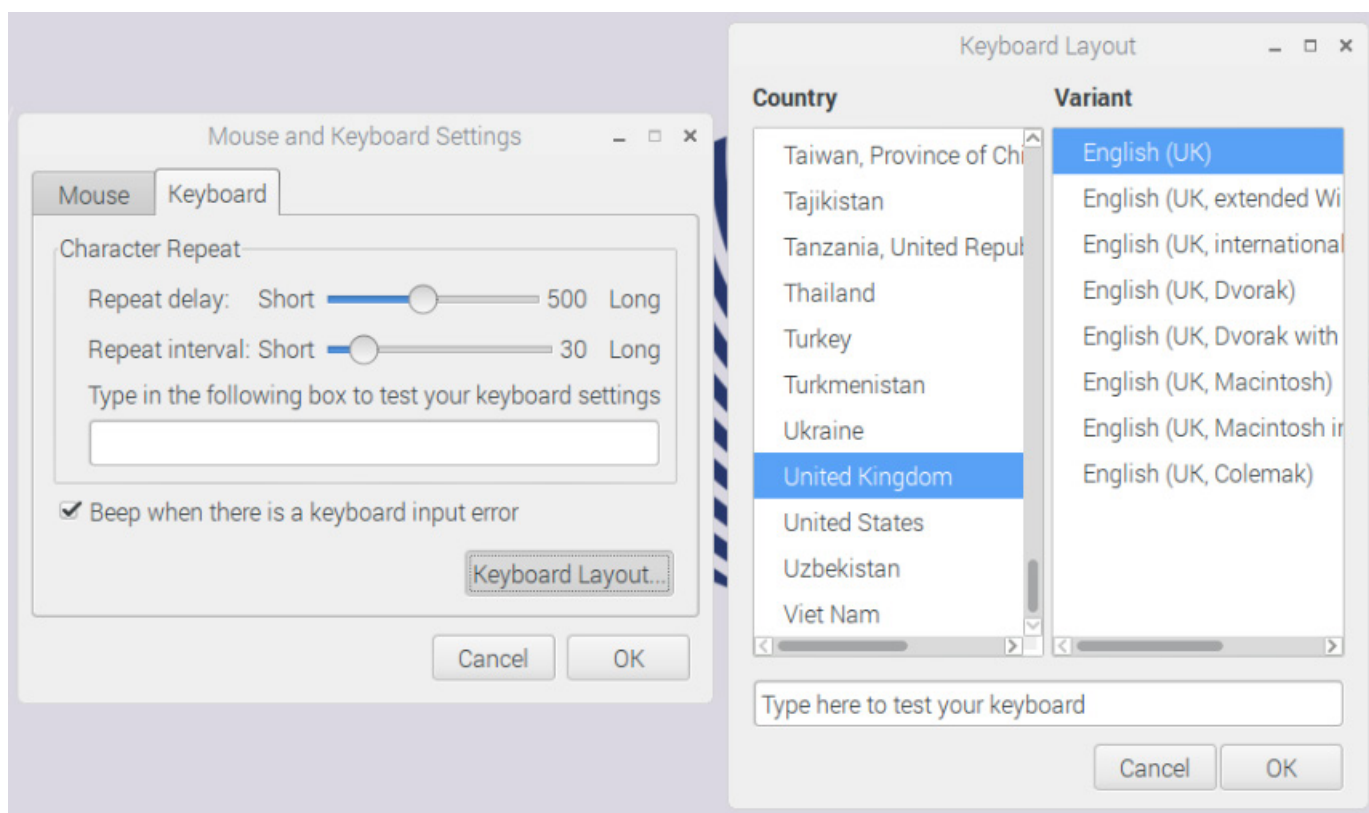
Activity 1.1 – Say cheese!

Let's start this activity with a preview of yourself.

Exercise

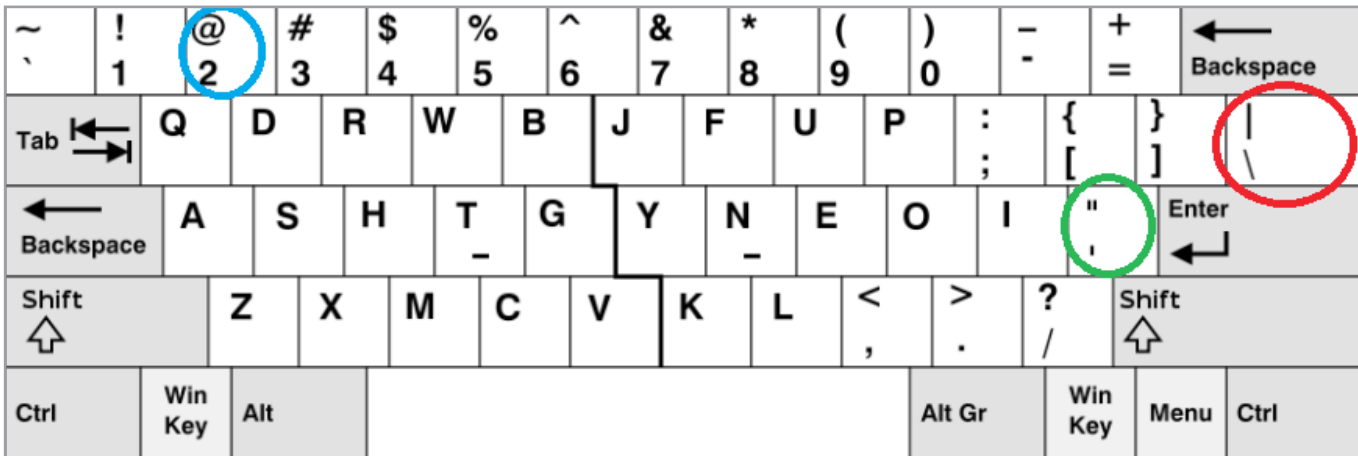
1. Open Python 3 by clicking on the Raspberry logo at the top of the screen. This will open the Menu. Select Programming > Python 3. This will open a Python Shell window.

Note: It may be necessary to change the keyboard settings in the Raspberry Pi, in order to locate important keys such as ‘, “ and #. The keyboard settings can be accessed in the same Preferences menu as used in activity o.



The location of the keys mentioned above will vary depending on the layout you select. In the UK-English layout the location of the important keys is as follows:

- # the backslash (\) key
- “ shift-pressing the 2 key
- , located next to Enter



Remember: In Python, # is used to mark comments, anything after the # on the line is executed as part of the code. It is a good idea to use comments to mark what each line does so that you can better understand it when you read back through.

2. Select File > New File, and type the code below into the new window.

```
File Edit Format Run Options Windows Help
from picamera import PiCamera
from time import sleep

camera = PiCamera()
camera.start_preview()
sleep(10)
camera.stop_preview()
```

3. Save your code and run it. What does it do?

Note: Do **not** name your file 'picamera.py'. If your camera preview does not end you can use the alt+F4 keyboard shortcut to close the window.

4. Try to move the camera and observe what is happening on the monitor. Can you explain what is happening in each line of code?

```
File Edit Format Run Options Windows Help
from picamera import PiCamera
from time import sleep #Imports the camera and time libraries

camera = PiCamera() #Defines the Pi Camera as a variable
camera.start_preview() #Starts the camera preview
sleep(10) #10 second delay
camera.stop_preview() #Stops the camera preview
```

↑ The comments (shown by a #) in this code explain what is happening at each stage, useful for humans only. They are ignored by the computer

5. Both cameras (Pi Camera and Pi NoIR Camera) can capture 2592 x 1944 static images and 1920 x 1080 video recordings, resulting in high definition images and videos. To change the resolution, we can use the camera.resolution command. For example, to change the resolution to 1024 x 768 the code would look like this:

```
File Edit Format Run Options Windows Help
from picamera import PiCamera
from time import sleep

camera = PiCamera()
camera.resolution = (1024,768)
camera.start_preview()
sleep(5)
camera.stop_preview()
```

Try to set the minimum resolution (64x64). Can you compare both images? Are important details lost in the lower resolution image?

6. Now try to set the resolution to the resolution of your smartphone. Does it work? Hint: You can use the internet to find the resolution of your smartphone.

Remember: The Astro Pi kit contains Visible Pi Camera V2 and NoIR Pi Camera V2, whilst Astro Pi Ed and Izzy are equipped with V1! Most importantly, this means that the maximum resolution that they can support is different. To explore the differences, visit the link here:

www.raspberrypi.org/documentation/hardware/camera

Activity 1.2 – Changing the camera orientation

The orientation of the Astro Pi camera once it is plugged in might not be what you expect. This is not a problem, as you can still save the images and videos and rotate them afterwards. However, it might be more useful to rotate the image before saving. Doing this means that the preview you see will be exactly the same as the final files you save.

Exercise

1. Let's try to change the orientation of the camera. Rotate the camera using this code:

```
File Edit Format Run Options Windows Help
from picamera import PiCamera
from time import sleep

camera = PiCamera()
camera.rotation = 180
camera.start_preview()
sleep(10)
camera.stop_preview()
```

2. Can you write a code that will change the rotation of camera by 90° every 3 seconds?

Write your code here and then try it on the Astro Pi:

There are many ways you could approach this problem; one solution is shown below:

```
File Edit Format Run Options Windows Help
from picamera import PiCamera
from time import sleep

x=0

camera = PiCamera()
camera.start_preview()
for x in [0,90,180,270]:
    camera.rotation = x
    sleep(3)
camera.stop_preview()
```


Activity 1.3 – Annotating images with text

Adding text is a good way to provide extra information with your images. This could be as simple as the date or time but could also include the coordinates of the Astro Pi when the photo or video was taken.

Exercise

1. Text can be added to the preview with the following line:

```
camera.annotate_text = 'Greetings from Earth'
```

```
File Edit Format Run Options Windows Help
from picamera import PiCamera
from time import sleep

camera = PiCamera()
camera.start_preview()
camera.annotate_text = 'Greetings from Earth'
sleep(5)
camera.stop_preview()
```

Note: To change size of the text use the command: `camera.annotate_text_size =`

→ ACTIVITY 2: STILL PICTURES & RECORDING VIDEOS

You already know how to use the camera to show a picture but for the Astro Pi mission you need to collect data. The collected data can be images, log files, or csv files containing a table of your sensor readings. In this activity you will learn how to capture a still image and video.

Activity 2.1 Take a selfie!

Let's start by taking a photo of yourself.

Exercise

1) Open a new Python 3 window and write the code below.

```
File Edit Format Run Options Window Help
from picamera import PiCamera
from time import sleep

camera = PiCamera()
camera.start_preview() #Starts the preview
sleep(5) #Runs for five seconds
camera.capture('/home/pi/Desktop/image.jpg') #Saves an image to the desktop named image.jpg
camera.stop_preview()
```

2) Select File > Save As and choose a file name for your program. Then select Run > Run Module. You have saved your first Astro Pi Picture for your Astro Pi Album.

3) Now let's create a time-lapse. To create a time-lapse still images are captured over a long period, with a delay in between each image capture. To try out the time-lapse feature, we will take a series of 5 pictures using a for loop instruction.

You will have to name your captures using an iterator. This means that every time we take a picture, it will be saved with a different name, and we won't lose the previous images! To do this we will need a line of code like this one:

```
camera.capture('/home/pi/Desktop/image%s.jpg' % i),
```

where *i* is an iterator. This means that each time around the loop the number stored as *i* increases by 1 and the number in the filename also increases by 1.

```
File Edit Format Run Options Windows Help
from picamera import PiCamera
from time import sleep

camera = PiCamera()
camera.start_preview()
for i in range(5): #Begins the iteration
    sleep(5)
    camera.capture('/home/pi/Desktop/image%s.jpg' % i) #Saves a picture every five seconds
camera.stop_preview() #as image0.jpg, image1.jpg... etc.
```

Now you can manually view your first time-lapse in the Astro Pi by viewing the first photo and pressing the right arrow key on your keyboard to view the other images as a slideshow.

Activity 2.2 Lights, camera, action!

Taking still images is a great way to use a camera to collect information. Another way is to capture a video. Now we are going to look at how we can use the camera to create a video.

Exercise

1) The line below is the command used to start the recording:

```
camera.start_recording('path')
```

where 'path' is the location you want to save the video.

Now, try the code below to create a simple video:

Don't be alarmed by '.h264'; this refers to the file type we will save our videos as, just like '.jpg' in our still image code!

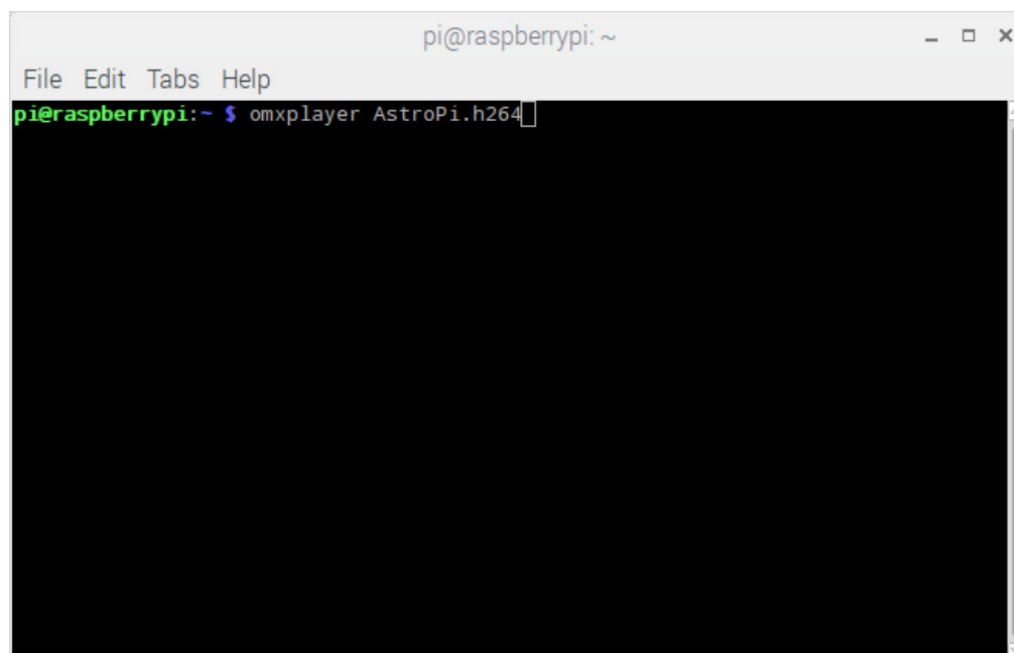
```
File Edit Format Run Options Windows Help
from picamera import PiCamera
from time import sleep

camera = PiCamera()
camera.start_preview()
camera.start_recording('/home/pi/Desktop/AstroPi.h264') #Begins recording a video named 'AstroPi' to the desktop
camera.wait_recording(10) #Records for 10 seconds
camera.stop_recording() #Stops recording the video
camera.stop_preview()
```

In order to play your video, you will need to open a terminal window and write the following command:

```
omxplayer AstroPi.h264
```

The terminal will look like this:



The ESA Astro Pi Team would love to see the videos you create! Perhaps you could create a greeting to an Astronaut and send it to us?

Send your videos to astropi@esa.int !

2) Discuss the advantages and disadvantages of still pictures and video-recordings. If you pointed the camera at earth from the ISS, what would be the advantages of still or video images?

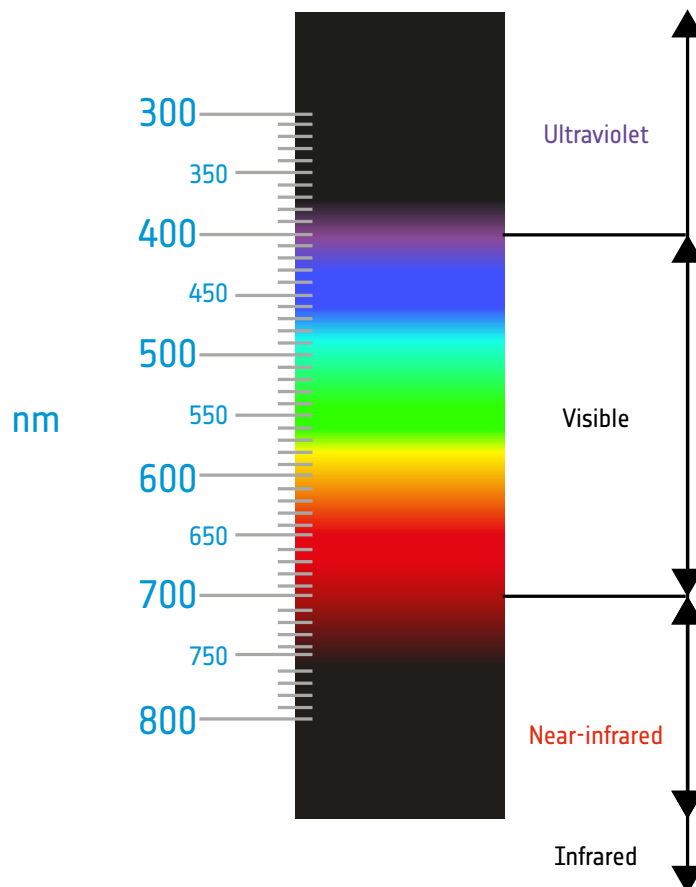
Visit the links below to take a look at some of the photos and time-lapses that ESA astronaut Paolo Nespoli took of Earth with his professional camera whilst on board the ISS!

[Photos from ESA astronaut Paolo Nespoli's VITA mission to the International Space Station](#)

[VITA mission 'Timelapse a Day' edition - from California to Mexico](#)

→ ACTIVITY 3: SEEING IN NEAR-INFRARED

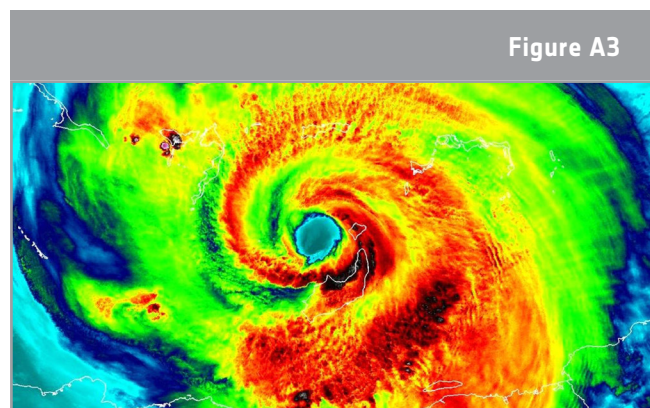
Near-infrared is part of the electromagnetic spectrum. The electromagnetic spectrum is what we use to classify the waves and phenomena around us in everyday life. The part of the electromagnetic spectrum you are most familiar with is the visible spectrum. This is the light and colours of the rainbow. Near-infrared is a different type of wave and invisible to humans, just like microwaves or X-rays are. A TV remote control flashes in near-infrared when a button is pressed, and we can't see it – but the NoIR camera can!



Many cameras are capable of seeing near-infrared light. However, to avoid color distortion, mobile phones and other small cameras usually have an infrared filter that only allows visible light to enter the camera. The NoIR has that filter removed, and so it is sensitive to the part of the IR spectrum near to visible red light. Let's test it out.

What a near-infrared camera can't see

It's important to distinguish between a near-infrared camera and a thermal infrared camera. You're probably used to colourful images showing differences in temperature when you think about an infrared camera. Unfortunately, a near-infrared camera doesn't see the thermal part of the spectrum, so you won't be able to produce images like this NASA image that was used to track the hot-spots of Hurricane Irma.



↑ Hurricane Irma

Exercise

1) Press the buttons on a TV remote, can you see anything coming from the emitter at the top of the remote?

2) Write a program in Python that shows a preview for 15 seconds. Turn the room light off and point the TV remote at the camera and press any button on the remote. What happens?

3) Is it possible to see it in daylight? What might be the reasons for the differences you see?

4) Can you identify the objects in the night sky that are sources of near-infrared light? What problems might there be in trying to view these objects with the NoIR camera?

Did you know....



Astronomers have to go to extreme lengths to collect the images they need for their research! Many observatories, such as this one at Mouna Kea are high on mountaintops. Near-infrared radiation is absorbed by water vapour, so near-infrared telescopes are placed above the clouds to get a clear image.

Extension - Using the NoIR camera to monitor plant health

Sensors carried by satellites can also 'see' other parts of the electromagnetic spectrum. Space scientists study near-infrared radiation to monitor Earth's vegetation from space. Plants strongly reflect near-infrared light, and healthy plants reflect more than stressed plants. Healthy plants reflect most of this light because they do not need it for photosynthesis. Just as we breathe in order to survive, plants use photosynthesis to stay alive. They process light into oxygen and carbohydrates. The chlorophyll is very important in this reaction. It absorbs long wavelengths (red light) and short wavelengths (blue light). Green light is not absorbed, and so this is why the leaves of a plant appear green to us.

Did you know?



Sentinel-3 is an ocean and land mission composed of three versatile satellites (Sentinel-3A, Sentinel-3B and Sentinel-3C). Sentinel-3A's ocean and land colour instrument can 'see' chlorophyll in vegetation. The amount of chlorophyll present in plants plays an important role in determining how healthy they are. Monitoring chlorophyll from space provides a valuable tool to support vegetation management, agriculture and forestry.

Your task is to investigate how you could use the NoIR camera to investigate the health of plants.

To do this, you may want to make use of one final piece of equipment in the ESA Astro Pi kit: the small blue square. It is a blue filter (we call it so because it appears blue to our eyes); it blocks green and red light and allows blue light and near-infrared light through. You can find the blue filter data sheet [here](#).

Use the questions below to guide your investigation!

1) Which part of the leaf is mainly responsible for the colour of the plants?

2) How is this different between healthy and unhealthy plants?

→ LINKS

Astro Pi training webinar:
astro-pi.org/updates/mission-space-lab-training-webinar

Astro Pi GitHub:
www.raspberrypi.org/learning/astro-pi-guide

Astro Pi resources produced by STEM UK:
www.stem.org.uk/elibrary/collection/4204

Information on the Astro Pi cameras:
www.raspberrypi.org/documentation/hardware/camera

Photographs taken by Paolo Nespoli:
www.flickr.com/photos/astro_paolo

Information on the blue filter:
www.raspberrypi.org/blog/whats-that-blue-thing-doing-here

Astro Pi camera resource produced by the Raspberry Pi Foundation
projects.raspberrypi.org/en/projects/getting-started-with-picamera

teach with space – meet the astro pi cameras | T05.4b
www.esa.int/education

The ESA Education Office welcomes feedback and comments
astropi@esa.int

An ESA Education production
in collaboration with ESERO Poland and ESERO Netherlands

Copyright 2018 © European Space Agency